High Availability Cluster
Multi-Processing for AIX

Planning Guide

Version 5.4.1

SC23-4861-10

Eleventh Edition (October 2007)

Before using the information in this book, read the general information in Notices for HACMP Planning Guide.

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

Contents

About This Guide

This guide provides information necessary to plan and install the High Availability Cluster Multi-Processing for AIX software, version 5.4.1.

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

HACMP Version	Book Name	Book Number
5.4.1	*Planning Guide*	SC23-4861-10
5.4	*Planning Guide*	SC23-4861-09
5.3 update 7/2006	*Planning and Installation Guide*	SC23-4861-08
5.3 update 8/2005	*Planning and Installation Guide*	SC23-4861-07
5.3	*Planning and Installation Guide*	SC23-4861-06
5.2 update 10/2005	*Planning and Installation Guide*	SC23-4861-05
5.2 update 10/2004	*Planning and Installation Guide*	SC23-4861-04
5.2	*Planning and Installation Guide*	SC23-4861-03
5.1 update 6/2004	*Planning and Installation Guide*	SC23-4861-02

Who Should Use This Guide

This guide is intended for system administrators and customer engineers responsible for:

- Planning hardware and software resources for an HACMP™ cluster
- Installing and configuring an HACMP cluster
- Maintaining and troubleshooting an HACMP cluster.

As a prerequisite to installing the HACMP software, you should be familiar with:

- IBM System p™ system components (including disk devices, cabling, and network adapters)
- The AIX operating system, including the Logical Volume Manager subsystem
- The System Management Interface Tool (SMIT)
- Communications, including the TCP/IP subsystem.

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 5L: 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
- eserver
- eServer Cluster 1600
- Enterprise Storage Server
- HACMP
- IBM
- NetView
- RS/6000
- Scalable POWERParallel Systems
- Series p
- Series x
- Shark
- SP
- 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: Overview of the Planning Process

This chapter provides an overview of the planning and installation process. This chapter contains the following sections:

- Overview of the Planning Process Goals
- Overview of the Planning Tools
- Overview of the Planning Process

Overview of the Planning Process Goals

Your major goal throughout the planning process is to eliminate single points of failure. 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 of providing that function, and the application or service dependent on that component becomes unavailable.

For example, if all the data for a critical application resides on a single disk, and that disk fails, that disk is a single point of failure for the entire cluster. Clients cannot access that application until the data on the disk is restored. Likewise, if dynamic application data is stored on internal disks rather than on external disks, it is *not* possible to recover an application by having another cluster node take over the disks. Therefore, identifying necessary logical components required by an application, such as filesystems and directories (which could contain application data and configuration variables), is an important prerequisite for planning a successful cluster.

Realize that, while your goal is to eliminate all single points of failure, you may have to make some compromises. There is usually a cost associated with eliminating a single point of failure. For example, purchasing an additional hardware device to serve as backup for the primary device increases cost. The cost of eliminating a single point of failure should be compared against the cost of losing services should that component fail. Again, the purpose of the HACMP is to provide a cost-effective, highly available computing platform that can grow to meet future processing demands.

Note: It is important that failures of cluster components be remedied as soon as possible. Depending on your configuration, it may *not* be possible for HACMP to handle a second failure, due to lack of resources.

Planning Guidelines

Designing the cluster that provides the best solution for your organization requires careful and thoughtful planning. In fact, adequate planning is the key to building a successful HACMP cluster. A well-planned cluster is easier to install, provides higher application availability, performs better, and requires less maintenance than a poorly planned cluster.

For a critical application to be highly available, none of the associated resources should be a single point of failure. As you design an HACMP cluster, your goal is to identify and address all potential single points of failure. Questions to ask include:

- What application services are required to be highly available? What is the priority of these services?
- What is the cost of a failure compared to the necessary hardware to eliminate the possibility of this failure?
- What is the maximum number of redundant hardware and software components that HACMP can support? (See Eliminating Single Points of Failure: Configuring Redundant Components Supported by HACMP).
- What is the required availability of these services? Do they need to be available 24 hours a day, seven days a week, or is eight hours a day, five days a week sufficient?
- What could happen to disrupt the availability of these services?
- What is the allotted time for replacing a failed resource? What is an acceptable degree of performance degradation while operating after a failure?
- Which failures will by automatically detected as cluster events? Which failures need to have custom code written to detect the failure and trigger a cluster event?
- What is the skill level of the group implementing the cluster? The group maintaining the cluster?

To plan, implement, and maintain a successful HACMP cluster requires continuing communication among many groups within your organization. Ideally, you should assemble the following representatives (as applicable) to aid in HACMP planning sessions:

- Network administrator
- System administrator
- Database administrator
- Application programming
- Support personnel
- End users

HACMP supports a variety of configurations, providing you with a great deal of flexibility. For information about designing for the highest level of availability for your cluster, see the IBM whitepaper *High Availability Cluster Multiprocessing Best Practices* at the following URL:

htttp://www.ibm.com/servers/eserver/clusters/whitepapers/hacmp_bestpractices.html

Eliminating Single Points of Failure: Configuring Redundant Components Supported by HACMP

The HACMP software provides numerous options to avoid single points of failure. The following table summarizes potential single points of failure and describes how to eliminate them by configuring redundant hardware and software cluster components:

Cluster Components	To Eliminate as Single Point of Failure	HACMP Supports
Nodes	Use multiple nodes	Up to 32.
Power sources	Use multiple circuits or uninterruptible power supplies (UPSs)	As many as needed.
Networks	Use multiple networks to connect nodes	Up to 48.
Network interfaces, devices, and labels	Use redundant network adapters	Up to 256.
TCP/IP subsystems	Use networks to connect adjoining nodes and clients	As many as needed.
Disk adapters	Use redundant disk adapters	As many as needed.
Controllers	Use redundant disk controllers	As many as needed.
Disks	Use redundant hardware and disk mirroring, striping, or both	As many as needed.
Applications	Assign a node for application takeover, configuring an application monitor, configuring clusters with nodes at more than one site.	As many as needed.
Sites	Use more than one site for disaster recovery.	2
Resource Groups	Use resource groups to specify how a set of entities should perform.	Up to 64 per cluster.
Cluster Resources	Use multiple cluster resources.	Up to 128 for Clinfo (more can exist in cluster).

Overview of the Planning Tools

This section describes the planning tools (worksheets) supplied by HACMP. Use these tools according to the processes described in Overview of the Planning Process. Each stage of the cluster planning process has worksheets that correspond with your planning tasks, to aid you in the planning process.

Paper Worksheets

The paper worksheets, which you fill out by hand and have physically nearby to refer to as you configure your cluster, are located in Appendix A: Planning Worksheets.

Online Planning Worksheets Application

The Online Planning Worksheets application is an online version of the paper worksheets. The application provides the following benefits:

- You may enter data into the worksheets as you plan the cluster and store the worksheets online
- At the end of the planning process, HACMP enables you to convert your planning data into an actual HACMP cluster configuration. For more information, see Chapter 9: Using Online Planning Worksheets.

Overview of the Planning Process

This section describes the steps for planning an HACMP cluster.

Step 1: Planning for Highly Available Applications

In this step, you plan the core of the cluster—the applications to be made highly available, the types of resources they require, the number of nodes, shared IP addresses, and a mode for sharing disks (non-concurrent or concurrent access). Your goal is to develop a high-level view of the system that serves as a starting point for the cluster design. After making these initial decisions, record them on the Application Worksheet and start to draw a diagram of the cluster. Chapter 2: Initial Cluster Planning describes this step of the planning process.

Step 2: Planning Cluster Topology

In this step, you decide on names for the cluster and the nodes. Optionally, you also decide on names for sites and decide which nodes belong to which site. Chapter 2: Initial Cluster Planning describes this step of the planning process.

Step 3: Planning Cluster Network Connectivity

In this step, you plan the networks that connect the nodes in your system. You first examine issues relating to TCP/IP and point-to-point networks in an HACMP environment. Next, you decide on the networks you will use, record your decision on the networks worksheet and add the networks to the cluster diagram. Chapter 3: Planning Cluster Network Connectivity describes this step of the planning process.

Step 4: Planning Shared Disk Devices

In this step, you plan the shared disk devices for the cluster. You decide which disk storage technologies you will use in your cluster, and examine issues relating to those technologies in the HACMP environment. Complete the disk worksheets and add the shared disk configuration to your diagram. Chapter 4: Planning Shared Disk and Tape Devices describes this step of the planning process.

You also decide whether you will be using enhanced concurrent mode volume groups in non-concurrent resource groups: For such volume groups, a faster disk takeover mechanism is used in HACMP. For more information, see Chapter 5: Planning Shared LVM Components.

Step 5: Planning Shared LVM Components

In this step, you plan the shared volume groups for the cluster. You first examine issues relating to LVM components in an HACMP environment, and then you fill out worksheets describing physical and logical storage. You may want to include planning for disaster recovery with cross-site LVM mirroring in this step. Chapter 5: Planning Shared LVM Components describes this step of the planning process.

Step 6: Planning Resource Groups

Planning resource groups incorporates all of the information you have generated in the previous steps. In addition, you should decide whether to use dependent resource groups or particular runtime policies for keeping certain related resource groups on the same node, on different nodes, or on the same site. Chapter 6: Planning Resource Groups describes this step of the planning process. Also see Appendix C: Resource Group Behavior During Cluster Events in the *Administration Guide* for further information that may help you plan your configuration.

Step 7: Planning Cluster Event Processing

In this step, you plan the event processing for your cluster. Chapter 7: Planning for Cluster Events describes this step of the planning process.

Step 8: Planning HACMP Clients

In this step, you examine issues relating to HACMP clients. Chapter 8: Planning for HACMP Clients describes this step of the planning process.

Chapter 2: Initial Cluster Planning

This chapter describes the initial steps you take to plan an HACMP cluster to make applications highly available, including completing the initial planning worksheets. This chapter contains the following sections:

- Prerequisites
- Overview
- Planning Cluster Security
- Planning Cluster Nodes
- Planning Cluster Sites
- Planning Cluster Security
- Planning Considerations for Multi-Tiered Applications
- Planning Applications and Application Servers
- Planning for AIX Fast Connect
- Planning for Highly Available Communication Links
- Drawing a Cluster Diagram
- Where You Go from Here.

Prerequisites

Before you start HACMP planning, make sure that you understand the concepts and terminology relevant to HACMP. In addition, thoroughly read and understand all concepts in the *Concepts and Facilities Guide* and the *Master Glossary*.

Overview

An HACMP cluster provides a highly available environment for mission-critical applications. In many organizations, these applications must remain available at all times. For example, an HACMP cluster could run a database server program that services client applications, keeping it highly available for clients that send queries to the server program.

To create effective clusters, fill out the planning worksheets provided in Appendix A: Planning Worksheets. These worksheets help you to ensure that you include the necessary components in a cluster, and provide documentation for future reference.

During the cluster-planning process, the Online Planning Worksheets application enables you to enter configuration data and save it to a cluster definition file. At the end of the planning process, you can use the cluster definition file to immediately configure your cluster. For information about Online Planning Worksheets, see Chapter 9: Using Online Planning Worksheets.

Planning Cluster Nodes

For each critical application, be mindful of the resources required by the application, including its processing and data storage requirements. For example, when you plan the size of your cluster, include enough nodes to handle the processing requirements of your application after a node fails.

You can create HACMP clusters that include up to 32 nodes. The Cluster Site Worksheet, provided in Appendix A: Planning Worksheets, is useful for this task.

Keep in mind the following considerations when determining the number of cluster nodes:

- An HACMP cluster can be made up of any combination of IBM System p™ workstations, LPARs, and SP nodes. Ensure that all cluster nodes do *not* share components that could be a single point of failure (for example, a power supply). Similarly, do *not* place nodes on a single rack.
- Create small clusters that consist of nodes that perform similar functions or share resources. Smaller, simple clusters are easier to design, implement, and maintain.
- For performance reasons, it may be desirable to use multiple nodes to support the same application. To provide mutual takeover services, the application must be designed in a manner that allows multiple instances of the application to run on the same node.

  For example, if an application requires that the dynamic data reside in a directory called ***/data***, chances are that the application can*not* support multiple instances on the same processor. For such an application (running in a non-concurrent environment), try to partition the data so that multiple instances of the application can run—each accessing a unique database.

  Furthermore, if the application supports configuration files that enable the administrator to specify that the dynamic data for *instance1* of the application resides in the *data1* directory, *instance2* resides in the *data2* directory, and so on, then multiple instances of the application are probably supported.
- In certain configurations, including additional nodes in the cluster design can increase the level of availability provided by the cluster; it also gives you more flexibility in planning node fallover and reintegration.

  The most reliable cluster node configuration is to have at least one standby node.
- Choose cluster nodes that have *enough* I/O slots to support redundant network interface cards and disk adapters.

  Remember, the cluster composed of multiple nodes is still more expensive than a single node, but without planning to support redundant hardware, (such as enough I/O slots for network and disk adapters), the cluster will have no better availability.
- Use nodes with similar processing speed.
- Use nodes with the sufficient CPU cycles and I/O bandwidth to allow the production application to run at peak load. Remember, nodes should have enough capacity to allow HACMP to operate.

  To plan for this, benchmark or model your production application, and list the parameters of the heaviest expected loads. Then choose nodes for an HACMP cluster that will *not* exceed 85% busy, when running your production application.

When you create a cluster, you assign a name to it. HACMP associates this name with the HACMP-assigned cluster ID.

Planning Cluster Sites

Cluster configurations typically use one site. If you have multiple sites, in addition to HACMP, use one of the following HACMP/XD features for disaster recovery:

- HACMP/XD for AIX for Metro Mirror
- HACMP/XD for AIX for GLVM
- HACMP/XD for AIX for HAGEO.

You also plan for sites if you intend to use cross-site LVM mirroring.

You configure sites as part of your cluster configuration process.

Planning Resources and Site Policy

HACMP tries to ensure that the primary instance of a resource group is maintained online at one site, and the secondary instance is maintained online at the other site. Plan which nodes to configure at which site, and where you want the active applications to run, so you can plan the resource group policies accordingly.

For more information on planning resource groups in a configuration with sites, see the section Planning Resource Groups in Clusters with Sites in Chapter 6: Planning Resource Groups.

For more information on migrating resource groups with replicated resources, see the *Administration Guide.*

All resources defined to HACMP must have unique names, as enforced by SMIT. The service IP labels, volume groups and resource group names must be both unique within the cluster and distinct from each other. The name of a resource should relate to the application it serves, as well as to any corresponding device, such as `websphere_service_address`.

HACMP/XD for GLVM Mirroring Overview

High Availability Cluster Multi-Processing Extended Distance (HACMP/XD) for Geographic Logical Volume Manager (GLVM) provides disaster recovery and data mirroring capability for the data at geographically separated sites. It protects the data against total site failure by remote mirroring, and supports unlimited distance between participating sites.

HACMP/XD for GLVM increases data availability by providing continuing service during hardware or software outages (or both), planned or unplanned, for a two-site cluster that serially accesses mirrored volume groups across an unlimited distance over an IP-based network.

HACMP/XD for GLVM provides two major facilities:

- *Remote data mirroring*. HACMP/XD for GLVM creates a remote mirror copy of the data that the application can access both at the local site and at the remote site.

  The software protects critical data by mirroring of the non-concurrent volume groups to which the application sends I/O requests. The application can access the same data regardless of whether the application is running on the local or remote site.

The data mirroring function utilizes the mirroring capabilities of the AIX Logical Volume Manager along with the mirroring function of the *Geographic Logical Volume Manager* that is provided by the HACMP/XD for GLVM software.

- *Integration with HACMP*. By integrating with HACMP, HACMP/XD for GLVM keeps mission-critical systems and applications operational in the event of disasters. It manages the fallover and fallback of the resource group that contains the application.

  Upon a node, a network interface, or a site failure, HACMP/XD for GLVM moves the resource group to another node. The node may belong either to the same site or to a remote site. With this operation, a complete, up-to-date copy of the volume group's data remains available for the application to use on the node at the same site or at another site.

  When the application is moved to a remote site, HACMP/XD for GLVM continues to mirror the data.

  See the *HACMP/XD for GLVM Planning and Administration Guide* for details.

HAGEO for AIX Overview

High Availability Geographic Cluster for AIX (HAGEO) is a component of HACMP/XD. HAGEO extends an HACMP cluster to encompass two physically separate data centers. Data entered at one site is sent across a point-to-point TCP/IP network and mirrored at a second, geographically distant location.

For example, data resulting from a bank transaction in New York City can be mirrored in Washington, DC. 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. This capability of the HAGEO software is called *geographic mirroring* (geo-mirroring).

HAGEO is a logical extension to the HACMP software. HACMP ensures that the computing environment *within a site* remains highly available. The HAGEO software ensures that the critical data remains highly available *even if an entire site* fails or is destroyed by a disaster.

Data entered on nodes at one site is written locally and mirrored to nodes at the other site. If a site fails, HAGEO automatically notifies the system administrator and makes the geo-mirrored data available at the remote site.

When the failed site has recovered, starting HAGEO on a node at the reintegrating site automatically reintegrates the site into the cluster. The geo-mirrored data is synchronized between the sites during the reintegration.

The HAGEO cluster thus increases 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.

The HACMP software provides site event scripts to customize how HACMP responds when a site goes up or down. For information about site event scripts, see Chapter 7: Planning for Cluster Events.

For complete information about the HAGEO product and how it is integrated with HACMP, see the HAGEO documentation at the following URL:

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

HACMP/XD for Metro Mirror Overview

IBM 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.

HACMP/XD Metro Mirror for SVC provides a fully automated, highly available disaster recovery management solution that takes advantage of the SAN Volume Controller’s ability to provide virtual disks derived from varied disk subsystems. The HACMP interface is designed so that once the basic SVC environment is configured, PPRC relationships are created automatically; no additional access to the SVC interface is needed.

See the *HACMP/XD: PPRC Planning and Administration Guide* for complete information.

Cross-Site LVM Overview

Cross-site LVM mirroring replicates data between the disk subsystem at each site for disaster recovery. You can set up disks located at two different sites for remote mirroring.

A storage area network (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.

These remote 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 a 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 from the surviving mirrors to stale mirrors after a disk or site failure and subsequent reintegration.

Completing the Cluster Site Worksheet

You use HACMP sites if you use cross-site LVM mirroring or any of the HACMP/XD components. Use the Cluster Site Worksheet to plan your HACMP sites. Note that related site information appears on the Resource Group Worksheet.

To complete the Cluster Site Worksheet:

1. Record the **Site Name**. Use no more than 32 alphanumeric characters and underscores.
2. Record the names of the cluster nodes that belong to the site in the **Cluster Nodes in Site** list. The nodes must have the same names as those you define to the HACMP cluster. A site can have 1–7 nodes. A node can belong to only one site.

   With sites configured for HAGEO, you can have up to eight nodes configured.

With sites configured for HACMP/XD for GLVM or HACMP/XD for Metro Mirror, you can have as many nodes in a cluster as HACMP supports.

3. Decide on site dominance and record it in the **Site Dominance** field. (Consult the documentation for the HACMP/XD component you plan to use.)
4. Select the site backup communication method and record it in the **Site Backup Communication Method** field. (Consult the documentation for the HACMP/XD component you plan to use.)
5. Record the **Inter-site Management Policy** on the Resource Group Worksheet. The default is Ignore. You can also select Online on Either Site, Online on Both Sites or Prefer Primary Site. For complete information on inter-site management policies and how they work with other resource group attributes and runtime policies, see Chapter 6: Planning Resource Groups.

Planning Cluster Security

HACMP provides cluster security by:

- Controlling user access to HACMP
- Providing security for inter-node communications.

Managing User Account Security

For information about configuring user accounts, see Chapter 16: Managing Users and Groups in the *Administration Guide*.

Managing Cluster Security

This section provides an overview of cluster security. For detailed information about configuring HACMP cluster security, see Chapter 17: Managing Cluster Security in the *Administration Guide*.

To secure inter-node communications, you can configure:

- Connection Authentication
- Message Authentication and Encryption.

Connection Authentication

HACMP provides connection authentication to protect HACMP communications between cluster nodes. There are two types of connection authentication:

- Standard authentication (default). Standard authentication includes verified connections by IP address and limits the commands that can be run with root privilege. This mode uses the principle of least-privilege for remote command execution, ensuring that no arbitrary command can run on a remote node with root privilege. A select set of HACMP commands is considered trusted and allowed to run as root; all other commands run as user `nobody`. Starting with HACMP 5.1, the ~**/.rhosts** dependency for inter-node communication was eliminated.
- Kerberos authentication. Kerberos authentication can be used in addition to standard security mode on SP systems.

You can also configure a virtual private network (VPN) for inter-node communications. If you use a VPN, use persistent labels for VPN tunnels.

Message Authentication and Encryption

HACMP provides security for HACMP messages sent between cluster nodes as follows:

- Message authentication ensures the origination and integrity of a message.
- Message encryption changes the appearance of the data as it is transmitted and returns it to its original form when received by a node that authenticates the message.

HACMP supports the following types of encryption keys for message authentication and encryption:

- Message Digest 5 (MD5) with Data Encryption Standard (DES)
- MD5 with Triple DES
- MD5 with Advanced Encryption Standard (AES).

Select an encryption algorithm that is compatible with the security methodology used by your organization.

Application Planning

Before you start planning for an application, be sure you understand the data resources for your application and the location of these resources within the cluster in order to provide a solution that enables them to be handled correctly if a node fails. To prevent a failure, you must thoroughly understand how the application behaves in a single-node and multi-node environment. Do *not* make assumptions about the application's performance under adverse conditions.

Use nodes with the sufficient CPU cycles and I/O bandwidth to allow the production application to run at peak load. Remember, nodes should have enough capacity to allow HACMP to operate.

To plan for this, benchmark or model your production application, and list the parameters of the heaviest expected loads. Then choose nodes for an HACMP cluster that will *not* exceed 85% busy, when running your production application.

We recommend that you configure multiple application monitors for an application and direct HACMP to both:

- Monitor the termination of a process or more subtle problems affecting an application
- Automatically attempt to restart the application and take appropriate action (notification or fallover) if restart attempts fail.

This section explains how to record all the key information about your application or applications on an Application Worksheet, provided in Appendix A: Planning Worksheets,and begin drawing your cluster diagram.

Keep in mind the following guidelines to ensure that your applications are serviced correctly within an HACMP cluster environment:

- Lay out the application and its data so that only the data resides on shared external disks. This arrangement *not* only prevents software license violations, but it also simplifies failure recovery.
- If you are planning to include multi-tiered applications in parent/child dependent resource groups in your cluster, see the section Planning Considerations for Multi-Tiered Applications. If you are planning to use location dependencies to keep certain applications on the same node, or on different nodes, see the section Resource Group Dependencies in Chapter 6: Planning Resource Groups.
- Write robust scripts to both start and stop the application on the cluster nodes. The startup script especially must be able to recover the application from an abnormal termination, such as a power failure. Ensure that it runs properly in a single-node environment before including the HACMP software. Be sure to include the start and stop resources on both the Application Worksheet and the Application Server Worksheet in Appendix A: Planning Worksheets.
- Confirm application licensing requirements. Some vendors require a unique license for each processor that runs an application, which means that you must license-protect the application by incorporating processor-specific information into the application when it is installed. As a result, even though the HACMP software processes a node failure correctly, it may be unable to restart the application on the fallover node because of a restriction on the number of licenses for that application available within the cluster. To avoid this problem, be sure that you have a license for each system unit in the cluster that may potentially run an application.
- Ensure that the application runs successfully in a single-node environment. Debugging an application in a cluster is more difficult than debugging it on a single processor.
- Verify that the application uses a proprietary locking mechanism if you need concurrent access.

For more information about what types of applications work best under HACMP and for strategies that can help keep your applications highly available, see Appendix B: Applications and HACMP.

Planning for Capacity Upgrade on Demand

Capacity Upgrade on Demand (CUoD) is one of the facilities of DLPAR (Dynamic Logical Partitioning) on some of the System p™ IBM servers that lets you activate preinstalled but yet inactive (and unpaid for) 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 (DLPAR) 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.

When it is necessary to run the application on the standby node, HACMP ensures that the node has the sufficient resources to successfully run the application and allocates the necessary resources. The resources can be allocated from two sources:

- The free pool. The DLPAR function provides the resources to the standby node, by allocating the resources available in the free pool on the frame.

- CUoD provisioned resources. If there are *not* enough available resources in the free pool that can be allocated through DLPAR, the CUoD function provides additional resources to the standby node, should the application require more memory or CPU.

  When you configure HACMP to use resources through DLPAR and CUoD, the LPAR nodes in the cluster do *not* use any additional resources until the resources are required by the application.

  HACMP ensures that each node can support the application with reasonable performance at a minimum cost. This way, you can upgrade the capacity of the logical partition in cases when your application requires more resources, without having to pay for idle capacity until you actually need it.

  For information on how to plan and configure CUoD in HACMP, see the *Administration Guide.*

Application Servers

To put the application under HACMP control, you create an *application server* resource that associates a user-defined name with the names of specially 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. This protects your application so that it does *not* become a *single point of failure*. An application server can also be monitored using the application monitoring facility or the Application Availability Analysis tool.

After you define the application server, you can add it to a *resource group.* A resource group is a set of resources that you define so that the HACMP software can treat them as a single unit. For information about planning to add your application servers and other resources to resource groups, see Chapter 6: Planning Resource Groups.

Applications Integrated with HACMP

Certain applications, including Fast Connect Services and Workload Manager, can be configured directly as highly available resources, without application servers or additional scripts. In addition, HACMP cluster verification ensures the correctness and consistency of certain aspects of your Fast Connect Services, or Workload Manager configuration.

Later sections in this chapter describe how these applications work in HACMP and how to plan for configuring them.

HACMP Smart Assist Programs

HACMP 5.4.1 offers three HACMP Smart Assist applications 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.

For more information about the HACMP Smart Assists, see the section Accessing Publications in About This Guide.

Application Monitoring

HACMP can monitor applications that are defined to application servers, in one of two ways:

- *Process monitoring* detects the termination of a process, using RSCT Resource Monitoring and Control (RMC) capability.
- *Custom monitoring* monitors the health of an application, using a monitor method that you define.

You can configure multiple application monitors and associate them with one or more application servers. You can assign each monitor a unique name in SMIT. 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. Otherwise, you can configure a custom monitor to check the health of the database along with a process termination monitor to instantly detect termination of the database process.

For information about how to define application monitors using the SMIT interface, see Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

You can use the Application Availability Analysis tool to measure the exact amount of time that any of your HACMP-defined applications is available. The HACMP software collects, time stamps, and logs the following information:

- An application monitor is defined, changed, or removed
- An application starts, stops, or fails
- A node fails or is shut down, or comes up
- A resource group is taken offline or moved
- Application monitoring via multiple monitors is suspended or resumed.

For more information about this tool, see the section Measuring Application Availability in Chapter 10: Monitoring an HACMP Cluster in the *Administration Guide*.

Planning Considerations for Multi-Tiered Applications

Business configurations that use multi-tiered applications can utilize parent/child dependent resource groups. For example, the 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 then started again) whenever a database fails. There are many application services provided by an environment like SAP, and the individual application components often need to be controlled in a specific order.

Establishing interdependencies between resource groups is also useful when system services are required to support application environments. Services such as **cron** jobs for pruning log files or for initiating backups need to move from one node to another along with an application, but typically are *not* initiated until the application is established. These services can be built into application server start and stop scripts, or they can be controlled through pre- and post- event processing. However, dependent resource groups simplify the way you configure system services to be dependent upon applications they serve.

Note: To minimize the chance of data loss during the application stop and restart process, customize your application server scripts to ensure that any uncommitted data is stored to a shared disk temporarily during the application stop process and read back to the application during the application restart process. It is important to use a shared disk as the application may be restarted on a node other than the one on which it was stopped.

You can also configure resource groups with location dependencies so that certain resource groups are kept ONLINE on the same node, at the same site, or on different nodes at startup, fallover, and fallback. For information on location dependencies between resource groups, see Chapter 6: Planning Resource Groups.

Planning Applications and Application Servers

This section contains the following topics:

- Completing the Application Worksheet
- Completing the Application Server Worksheet
- Completing the Application Monitoring Worksheet (Process Monitor or Custom Monitor).

Completing the Application Worksheet

Print the Application Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Print one copy for each application you want to keep highly available in the cluster.

To complete the Application Worksheet:

1. Assign a name to the application and record it in the **Application Name** field.
2. Enter information describing the application’s executable and configuration files under the **Directory/Path**, **File System**, **Location**, and **Sharing** columns. Enter the full path name of each file. You can store the file system for either the executable or configuration files on either an internal or external disk device. Different situations may require you to do it one way or the other. Be aware, if you store the file system on the internal device, the device will *not* be accessible to other nodes during a resource takeover.
3. Enter information describing the application’s data and log files under the appropriate columns listed in Step 2. Data and log files can be stored in a file system (or on a logical device) and must be stored externally if a resource takeover is to occur successfully.
4. Enter in the **Normal Start Command/Procedures** field the names of the start scripts you created to start the application after a resource takeover.
5. Enter in the **Verification Commands/Procedures** field the names of commands or procedures to use to ensure that the normal start scripts ran successfully.

Completing the Application Server Worksheet

Print the Application Server Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Print one copy for each application in the cluster.

To complete the Application Server Worksheet:

1. Enter the cluster name in the **Cluster Name** field.

   You determined this value while completing the worksheet.

   For each application server you define, fill in the following fields:

   - Assign and record a name for the application in the **Application** field.
   - Assign a symbolic name that identifies the server and record it in the **Server Name** field. For example, you could name the application server for the customer database application *custdata*. The name can include up to 64 characters and can contain only alphanumeric and underscore (_) characters.
   - Record the full pathname of the user-defined script that starts the application server in the **Start Script** field. (Maximum 256 characters.) This information was recorded in the Application Worksheet. Be sure to include the script's arguments, if necessary. The script is called by the cluster event scripts. For example, you could name the start script and specify its arguments for starting the *custdata* application server as follows:

     ```
     /usr/start_custdata -d mydir -a jim_svc
     ```

     where the **-d** option specifies the name of the directory for storing images, and the **-a** option specifies the service IP address (label) for the server running the demo.
   - Record the full pathname of the user-defined script that stops the application in the **Stop Script** field. (Maximum 256 characters.) This script is called by the cluster event scripts. For example, you could name the stop script for the *custdata* application server **/usr/stop_custdata**.

Completing the Application Monitoring Worksheet

This section describes how to complete the following worksheets:

- Completing the Application Monitor (Process) Worksheet
- Completing the Application Monitor (Custom) Worksheet.

When you use an application monitor with HACMP, HACMP restarts the application. The Systems Resource Controller (SRC), should *not* restart it.

If a monitored application is controlled by the SRC, ensure that **action:multi** is set as follows:

-O Specifies that the subsystem is *not* restarted if it stops abnormally.

-Q Specifies that multiple instances of the subsystem are *not* allowed to run at the same time.

To review these settings, use the following command:

```
lssrc -Ss Subsystem | cut -d : -f 10,11
```

If the values are *not* **-O** and **-Q**, use the **chssys** command to change them.

Completing the Application Monitor (Process) Worksheet

Print the Application Monitor Worksheet (Process Monitor) from Appendix A: Planning Worksheets, and fill it out using the information in this section. Print as many copies as you need. You can configure multiple application monitors and associate them with one or more application servers. You can assign each monitor a unique name in SMIT.

To complete the Application Monitor Worksheet (Process Monitor):

1. Enter the cluster name in the **Cluster Name** field.
2. Specify the **Application Server Name** for which you are configuring a process monitor.
3. Ensure whether this application can be monitored with a process monitor.

   For example, shell scripts can*not* be monitored. For more information about application monitoring, see the section Configuring Application Monitoring in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

   If the application can be monitored, proceed to step 4.

   If the application is *not* to be monitored, proceed to the instructions for the Application Monitor Worksheet (Custom Monitor).
4. Indicate the name(s) of one or more processes to be monitored. Be careful when listing process names. It is very important that the names are correct when you enter them in SMIT to configure the application monitor. For information about identifying process names, see Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.
5. Specify the user ID of the owner of the processes specified in step 4 (for example, *root*). Note that the process owner must own all processes to be monitored.
6. Specify how many instances of the application to monitor. The default is **1** (one) instance. This number *must* be **1** (one) if you have specified more than one process to monitor.
7. Specify the time (in seconds) to wait before beginning monitoring. For instance, with a database application, you may wish to delay monitoring until after the start script and initial database search have been completed. You may need to experiment with this value to balance performance with reliability. In most circumstances, this value should *not* be zero.

   Note: Allow at least 2–3 seconds before beginning application monitoring, to ensure HACMP and user applications quiesce.
8. Specify the Restart Count, denoting the number of times to attempt to restart the application before taking any other actions. The default is **3**. Make sure you enter a Restart Method (see step 13) if your Restart Count is any non-zero value.
9. Specify the interval (in seconds) that the application must remain stable before resetting the Restart Count. This interval becomes important if a number of failures occur over a period of time. Resetting the count to zero at the proper time prevents a later failure from being counted as the last failure from the previous problem, in cases when it should be counted as the first of a new problem.

   Do *not* set this interval to be shorter than (Restart Count) x (Stabilization Interval). The default is 10 percent longer than that value. If it is too short, the count will be reset to zero repeatedly, and the specified failure action will never occur.

10. Specify the action to be taken if the application can*not* be restarted within the Restart Count. The default choice is **notify**, which runs an event to inform the cluster of the failure. You can also specify **fallover**, in which case the resource group containing the failed application moves over to the cluster node with the next-highest priority for that resource group.

    Note: Keep in mind that if you choose the **fallover** option of application monitoring, which may cause resource groups to migrate from their original owner node, the possibility exists that while the highest priority node is up, the resource group remains down. This situation occurs when an **rg_move** event moves a resource group from its highest priority node to a lower priority node, and then you stop cluster services on the lower priority node and bring the resource groups offline.

    Unless you bring up the resource group manually, it will remain in an inactive state. For more information, see the section on Application Monitoring Prerequisites and Considerations in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

11. (*Optional*) Define a notify method that will run when the application fails. This user-defined method, typically a shell script, runs during the restart process and during notify activity.

12. (*Optional*) Specify an application cleanup script to be called when a failed application is detected, before calling the restart method. The default is the application server stop script you define when you set up the application server.

    Note: Since the application is already stopped when this script is called, the server stop script may fail. For more information on writing correct stop scripts, see Appendix B: Applications and HACMP.

13. (*Required if Restart Count is not zero.*) The default restart method is the application server start script you define when the application server is set up. Specify a different restart method if desired.

Completing the Application Monitor (Custom) Worksheet

Print the Application Monitor Worksheet (Custom Monitor) from Appendix A: Planning Worksheets, and fill it out using the information in this section. If you plan to set up a custom monitor method, complete this worksheet for each user-defined application monitor you plan to configure. You can configure multiple application monitors and associate them with one or more application servers. You can assign each monitor a unique name in SMIT.

To complete the Application Monitor Worksheet (Custom Monitor):

1. Enter the cluster name in the **Cluster Name** field.
2. Fill in the name of the application server.

3. Specify a script or executable for custom monitoring of the health of the specified application. Do *not* leave this field blank when you configure the monitor in SMIT. The monitor method must return a zero value if the application is healthy, and a non-zero value if a problem is detected. See the note in Step 6 regarding defining a monitor method.
4. Specify the polling interval (in seconds) for how often the monitor method is to be run. If the monitor does *not* respond within this interval, it is considered "hung."
5. Specify a signal to kill the user-defined monitor method if it does *not* return within the monitor interval. The default signal is **kill -9**.
6. Specify the time (in seconds) to wait before beginning monitoring. For instance, with a database application, you may wish to delay monitoring until after the start script and initial database search have been completed. You may need to experiment with this value to balance performance with reliability.

   Note: In most circumstances, this value is *not* be zero. Allow at least 2–3 seconds before beginning application monitoring, to ensure HACMP and user applications quiesce.

7. Specify the restart count, denoting the number of times to attempt to restart the application before taking any other actions. The default is **3**.
8. Specify the interval (in seconds) that the application must remain stable before resetting the restart count. This interval becomes important if a number of failures occur over a period of time. Resetting the count to zero at the proper time keeps a later failure from being counted as the last failure from the previous problem, when it should be counted as the first of a new problem.

   Do *not* set this to be shorter than (Restart Count) x (Stabilization Interval + Monitor Interval). The default is 10 percent longer than that value. If it is too short, the count will be reset to zero repeatedly, and the specified failure action will never occur.
9. Specify the action to be taken if the application can*not* be restarted within the restart count. You can keep the default choice **notify**, which runs an event to inform the cluster of the failure, or choose **fallover**, in which case the resource group containing the failed application moves over to the cluster node with the next-highest priority for that resource group.

   Note: Keep in mind that if you choose the **fallover** option of application monitoring, which may cause resource groups to migrate from their original owner node, the possibility exists that while the highest priority node is up, the resource group remains down. This situation occurs when an **rg_move** event moves a resource group from its highest priority node to a lower priority node, and then you stop cluster services on the lower priority node and bring resource groups offline.

   Unless you manually bring up the resource group, it will remain in an inactive state. For more information, see the section on Application Monitoring Prerequisites and Considerations in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

10. (*Optional*) Define a notify method that will run when the application fails. This custom method runs during the restart process and during a **server_down** event.
11. (*Optional*) Specify an application cleanup script to be called when a failed application is detected, before calling the restart method. The default is the application server stop script defined when the application server was set up.

    Note: The application may be already stopped when this script is called, and the server stop script may fail. For more information on writing correct stop scripts, see Appendix B: Applications and HACMP.

12. (*Required if Restart Count is not zero.*) The default restart method is the application server start script you define when the application server is set up. Specify a different restart method if desired.

Notes on Defining a Custom Monitoring Method

When defining your custom monitoring method, keep in mind the following points:

- You can configure multiple application monitors and associate them with one or more application servers. You can assign each monitor a unique name in SMIT.
- The monitor method must be an executable program (it can be a shell script) that tests the application and exits, returning an integer value that indicates the application's status. The return value must be zero if the application is healthy, and must be a non-zero value if the application has failed.
- HACMP does *not* pass arguments to the monitor method.
- The monitor method logs messages to the **/tmp/clappmond.application_monitor_name.monitor.log** file by printing messages to the standard output (**stdout**) file. Each time the application runs, the monitor log file is overwritten.
- Do *not* make the method overly complicated. The monitor method is killed if it does *not* return within the specified polling interval. Test your monitor method under different workloads to arrive at the best polling interval value.
- Ensure that the System Resource Controller (SRC) is configured to restart the application and take steps accordingly. For more information on which steps to take, see Planning Applications and Application Servers.

Planning for AIX Fast Connect

Some applications, such as AIX Fast Connect, do *not* require application servers because they are already integrated with HACMP. You do *not* need to write additional scripts or create an application server for these applications to be made highly available under HACMP.

AIX Fast Connect allows client PCs running Windows, DOS, and OS/2 operating systems to request files and print services from an AIX server. Fast Connect supports the transport protocol NetBIOS over TCP/IP. You can use SMIT to configure AIX Fast Connect resources.

The Fast Connect application is integrated with HACMP. You can use SMIT to configure Fast Connect services as highly available resources in resource groups. HACMP can then stop and start the Fast Connect resources when fallover, recovery, and dynamic resource group migrations occur. This application does *not* need to be associated with application servers or special scripts.

In addition, the HACMP cluster verification process ensures the accuracy and consistency of certain aspects of your AIX Fast Connect configuration.

Planning Considerations for Fast Connect

To plan for configuration of Fast Connect as a cluster resource in HACMP:

- Install the Fast Connect Server on all nodes in the cluster.
- If Fast Connect printshares are to be highly available, ensure that the AIX print queue names *match* for every node in the cluster.
- For non-concurrent groups, assign the *same* NetBIOS name to each node when the Fast Connect Server is installed.

  This action minimizes the steps needed for the client to connect to the server after fallover.

  Note: Only one instance of a NetBIOS name can be active at one time. For that reason, do *not* to activate Fast Connect servers that are under HACMP control.

- For concurrently configured resource groups, assign *different* NetBIOS names across nodes.
- In concurrent configurations, define a second, non-concurrent resource group to control any file system that must be available for the Fast Connect nodes.

  Having a second resource group configured in a concurrent cluster keeps the AIX file systems used by Fast Connect cross-mountable and highly available in the event of a node failure.

- Do *not* configure Fast Connect in a mutual takeover configuration.

  A node can*not* participate in two Fast Connect resource groups at the same time.

Fast Connect as a Highly Available Resource

When AIX Fast Connect resources are configured as part of a resource group, HACMP handles them as described in the following list:

- Fast Connect start and stop. When a Fast Connect server has resources configured in HACMP, HACMP starts and stops the server during fallover, recovery, and resource group reconfiguration or migration.

  Note: The Fast Connect server must be stopped on all nodes when bringing up the cluster. This ensures that HACMP will start the Fast Connect server and handle its resources properly.

- Node failure. When a node that owns Fast Connect resources fails, the resources become available on the takeover node. When the failed node rejoins the cluster, the resources are again available on the original node (as long as the resource policy is such that the failed node reacquires its resources).

  Clients *do not* need to reestablish a connection to access the Fast Connect Server after fallover, as long as IP and Hardware Address Takeover (HWAT) are configured and occur, and users have configured their Fast Connect server with the same NetBIOS name on all nodes (for non-concurrent resources groups).

  For switched networks and for clients *not* running Clinfo, you may need to take additional steps to ensure client connections after fallover. For more information about configuration considerations for clients *not* running Clinfo, see Chapter 8: Planning for HACMP Clients.
- Adapter failure. When a service adapter or network interface card running the transport protocol needed by the Fast Connect server fails, HACMP performs an adapter swap as usual, and Fast Connect establishes a connection with the new adapter. After an adapter failure, clients are temporarily unable to access shared resources such as files and printers. After the adapter swap is complete, clients can again access their resources.

Completing the Fast Connect Worksheet

Complete the Fast Connect Worksheet to identify the resources to configure as Fast Connect resources. The worksheets are located in Appendix A: Planning Worksheets.

To complete the Fast Connect Worksheet:

1. Enter the cluster name in the **Cluster Name** field.
2. Record the name of the resource group that will contain the Fast Connect Resources.
3. Record the nodes participating in the resource group.
4. Record the Fast Connect Resources to be made highly available. When you configure your resources from the SMIT, you will select these resources from a picklist.
5. Record the file systems that contain the files or directories that you want Fast Connect to share. Be sure to specify these in the File Systems SMIT field when you configure the resource group.

For more information about using AIX Fast Connect, see Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

Planning for Highly Available Communication Links

HACMP can provide high availability for the following types of communication links:

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

LAN interface cards include Ethernet, Token Ring, and FDDI. These cards are configured as part of the HACMP cluster topology.

X.25 cards are usually, although *not* always, used for WAN connections. They provide a mechanism to connect dissimilar machines, from mainframes to dumb terminals. The typical use of X.25 networks makes these cards a different class of devices that are *not* included in the cluster topology and *not* controlled by the standard HACMP topology management methods. This means that heartbeats are *not* used to monitor an X.25 card's status, and you do *not* define X.25-specific networks in HACMP.

Once defined in an HACMP resource group, communication links are protected in the same way other HACMP resources are. In the event of a LAN card or X.25 link failure, or general node or network failures, a highly available communication link falls over to another available card on the same node or on a takeover node.

For clusters where highly available communication links have been configured, verification checks whether the appropriate fileset has been installed and reports an error in the following situations:

- If SNA HA Communication Links is configured as a part of a resource group and the **sna.rte** fileset version 6.1.0.0 or higher is missing on some nodes that are part of this resource group
- If X.25 HA Communication Links is configured as a part of a resource group and the **sx25.rte** fileset version 2.0.0.0 or higher is missing on some nodes that are part of this resource group
- On SNA X.25 and up, if HA Communication Links is configured as a part of a resource group and both the **sx25.rte** filesets version 6.1.0.0 *and* 2.0.0.0 or higher is missing on some nodes that are part of this resource group.

SNA and X.25 Links Required Software and Hardware

SNA links and X.25 links require additional configuration, outside of HACMP, to configure highly available communications links in HACMP. For more information about configuring the required software and hardware, see the manuals at the following URLs:

www.ibm.com

www.redbooks.ibm.com

SNA links require the following:

- CS/AIX version 6.1 or higher is required.
- SNA-over-LAN links are supported over Ethernet, Token Ring, and FDDI adapters.

X.25 links require the following:

- AIXlink/X.25 version 2 or higher is required.
- X.25 links are supported on the following adapters:
  - IBM 2-Port Multiprotocol Adapter (DPMP)
  - IBM Artic960Hx PCI Adapter (Arctic)

Completing the Communication Links Worksheets

This section describes how to fill out the following worksheets:

- Completing the Communication Links (SNA-Over-LAN) Worksheet
- Completing the Communication Links (X.25) Worksheet
- Completing the Communication Links (SNA-Over-X.25) Worksheet

Completing the Communication Links (SNA-Over-LAN) Worksheet

Print the Communication Links (SNA-Over-LAN) Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Complete one worksheet for each SNA-over-LAN communication link in your cluster.

The SNA link must already be configured separately in AIX before the link can be defined to HACMP. Much of the information you fill in here will be drawn from the AIX configuration information.

To complete the Communication Links (SNA-Over-LAN) Worksheet:

1. Enter the cluster name in the **Cluster Name** field.
2. Enter the resource group in which the communication link will be defined in the **Resource Group** field.
3. Enter nodes participating in the resource group in the **Nodes** field.
4. Enter the link name in the **Name** field.
5. Enter the DLC name in the **DLC Name** field. This is the name of an existing DLC profile to be made highly available.
6. Enter the names of any ports to be started automatically in the **Port(s)** field.
7. Enter the names of the link stations in the **Link Station(s)** field.
8. Enter the name of the **Application Service File** that this link uses to perform customized operations when this link is started or stopped.

Completing the Communication Links (X.25) Worksheet

Print the Communication Links (X.25) Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Complete one worksheet for each X.25 communication link in your cluster.

The X.25 worksheet is for configuring the link in HACMP. The X.25 adapter and link must already be configured separately in AIX before the link can be defined for HACMP. Much of the information you fill in here will be drawn from the AIX configuration information.

To complete the Communication Links (SNA-Over-LAN) Worksheet:

1. Enter the cluster name in the **Cluster Name** field.
2. Enter the resource group in which the communication link will be defined in the **Resource Group** field.
3. Enter nodes participating in the resource group in the **Nodes** field.
4. Enter the link name in the **Name** field.
5. Enter the X.25 **Port** to be used for this link (for example, sx25a0). The port name must be unique across the cluster. This name must begin with "sx25a" but the final numeric character is your choice. The port name can be up to eight characters long; therefore the final numeric can contain up to three digits.
6. In the **Address/NUA** field, enter the X.25 address (local NUA) that will be used by this link.
7. For **Network ID**, the default value is 5. Enter a different number here if needed.
8. Enter the X.25 **Country Code**, or leave it blank and the system default will be used.
9. For **Adapter Name(s)**, identify the communication adapters you want this link to be able to use. Enter HACMP names, *not* device names. In SMIT, you select an entry for this field from a picklist.
10. Enter the name of the **Application Service File** that this link uses to perform customized operations when this link is started or stopped.

Completing the Communication Links (SNA-Over-X.25) Worksheet

Print the Communication Links (SNA-Over-X.25) Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Complete one worksheet for each SNA-over-X.25 communication link in your cluster. The SNA-Over-X.25 worksheet is for configuring the link in HACMP. The SNA link and the X.25 adapter and link must already be configured separately in AIX before the SNA-over-X.25 link can be defined for HACMP. Much of the information you fill in here will be drawn from the AIX configuration information.

To complete the Communication Links (SNA-Over-X.25) Worksheet:

1. Enter the cluster name in the **Cluster Name** field.
2. Enter the resource group in which the communication link will be defined in the **Resource Group** field.
3. Enter nodes participating in the resource group in the **Nodes** field.
4. Enter the link name in the **Name** field.
5. Enter the **X.25 Port** to be used for this link (for example, sx25a0). The port name must be unique across the cluster. This name must begin with "sx25a" but the final numeric character is your choice. The port name can be up to eight characters long; therefore the final numeric can contain up to three digits.
6. In the **X.25 Address/NUA** field, enter the X.25 address (local NUA) that will be used by this link.
7. For **X.25 Network ID**, the default value will be 5. Enter a different number here if needed.
8. Enter the **X.25 Country Code**, or leave it blank and the system default will be used.

9. For **X.25 Adapter Name(s)**, identify the communication adapters you want this link to be able to use. Enter HACMP names, *not* device names. In SMIT, you select an entry for this field from a picklist.
10. Enter the **SNA DLC**. This is the name of an existing DLC profile to be made highly available. In SMIT, you select an entry for this field from a picklist.
11. Enter the names of any SNA ports to be started automatically in the **SNA Port(s)** field.
12. Enter the names of the SNA link stations in the **SNA Link Station(s)** field.
13. Enter the name of the **Application Service File** that this link uses to perform customized operations when this link is started or stopped.

    For information on how to write an appropriate script for this file, see Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

Drawing a Cluster Diagram

The cluster diagram combines the information from each step in the planning process into one drawing that shows the cluster's function and structure.

The following illustration shows a mixed cluster that includes a rack-mounted system and standalone systems. The diagram uses rectangular boxes to represent the slots supported by the nodes. If your cluster uses thin nodes, darken the outline of the nodes and include two nodes to a drawer. For wide nodes, use the entire drawer. For high nodes, use the equivalent of two wide nodes. Keep in mind that each thin node contains an integrated Ethernet connection.

Begin drawing this diagram by identifying the cluster name and the applications that are being made highly available. Next, darken the outline of the nodes that will make up the cluster. Include the name of each node. While reading subsequent chapters, you will add information about networks and disk storage subsystems to the diagram.

Figure 1. Initial Diagram of Two Types of HACMP Clusters

Where You Go from Here

Next you will design the network connectivity of your cluster, described in Chapter 3: Planning Cluster Network Connectivity.

Chapter 3: Planning Cluster Network Connectivity

This chapter describes planning the network support for an HACMP cluster. This chapter contains the following sections:

- Prerequisites
- Overview
- General Network Considerations for HACMP
- Heartbeating in HACMP
- Heartbeating over IP Aliases
- Heartbeating over Disk
- Designing the Network Topology
- Planning for IP Address Takeover via IP Aliases
- Planning for IP Address Takeover via IP Replacement
- Planning for Other Network Conditions
- Planning for SP Networks
- Completing the Network Worksheets
- Defining Hardware Addresses
- Adding the Network Topology to the Cluster Diagram

Prerequisites

In Chapter 2: Initial Cluster Planning, you began planning your cluster, identifying the number of nodes and the key applications you want to make highly available. You started drawing the cluster diagram. This diagram is the starting point for the planning you will do in this chapter.

Also, by now you should have decided whether or *not* you will use IP address takeover (IPAT) to maintain specific service IP addresses.

Overview

This chapter explains how to plan networking support for the cluster. Your primary goal is to use redundancy to design a cluster topology that eliminates network components as potential single points of failure. The following table lists these network components with solutions:

Cluster Network Object	Eliminated as a Single Point of Failure by...
Network	Using multiple networks to connect nodes
TCP/IP subsystem	Using a non-IP network to back up TCP/IP
Network Interface Card (NIC)	Using redundant NICs on each network

In this chapter you complete the following planning tasks:

- Designing the cluster network topology, that is, the combination of IP and non-IP (point-to-point) networks that connect your cluster nodes and the number of connections each node has to each network

  Note: To avoid cluster partitioning, we highly recommend configuring redundant networks in the HACMP cluster and using both IP and non-IP networks. Out of these networks, some networks will be used for heartbeating purposes. For information on planning heartbeating networks, see Heartbeating in HACMP in this chapter.

- Determining whether service IP labels will be made highly available with IP address takeover (IPAT) via IP Aliases or IPAT via IP Replacement with or without Alternate Hardware Address Takeover
- Completing the network and network interface planning worksheets
- Adding networking to your cluster diagram.

This chapter also includes detailed information about setting up IPAT via IP Replacement with Hardware Address Takeover (HWAT) via alternate hardware addresses.

General Network Considerations for HACMP

This section provides general guidelines for the type of networking required for successful HACMP operation.

Note: If you are planning networks for a cluster with sites, see the documentation for the corresponding disaster recovery solution (HACMP/XD for Metro Mirror, HACMP/XD for GLVM or HACMP/XD for HAGEO).

The following topics are described in this section:

- Supported Network Types
- IP Aliases

- Network Connections
- ARP Cache Updating
- IP Labels
- Cluster Partitioning
- General Network Connection Example
- HACMP Configuration in Switched Networks
- HACMP and Virtual Ethernet (VLAN)
- Topology Services and Heartbeat Rings.

Supported Network Types

HACMP allows inter-node communication with the following TCP/IP-based networks:

- Ethernet
- Token Ring
- Fiber Distributed Data Interchange (FDDI)
- ATM and ATM LAN Emulation
- SP Switch1 and SP Switch2
- Etherchannel

IP Aliases

An IP alias is an IP label/address that is configured onto a NIC *in addition to* the normally-configured IP label/address on the NIC. The use of IP aliases is an AIX function that HACMP supports. AIX supports multiple IP aliases on a NIC. Each IP alias on a NIC can be on a separate subnet. AIX also allows IP aliases with different subnet masks to be configured for an interface; HACMP does *not* yet support this function.

IP aliases are used in HACMP both as service and non-service addresses for IP address takeover (IPAT), in addition to configuring networks for heartbeating.

For information about subnet requirements for service and non-service IP labels/addresses managed by HACMP, see Planning Service IP Labels in Resource Groups in Chapter 6: Planning Resource Groups in this chapter.

Network Connections

HACMP requires that each node in the cluster has at least one direct, non-routed network connection with every other node. The software uses these network connections to pass heartbeat messages among the cluster nodes to determine the state of all cluster nodes, networks and network interfaces.

HACMP requires all of the communication interfaces for a given cluster network be defined on the same physical network and route packets (including broadcast packets) to each other. They must also be able to receive responses from each other without interference by any network equipment.

Between cluster nodes, place only intelligent switches, routers, or other network equipment that transparently pass through UDP broadcasts and other packets to all cluster nodes. That is, do *not* place intelligent switches, routers, or other network equipment between cluster nodes if they do *not* transparently pass through UDP broadcasts and other packets to all cluster nodes. This prohibition includes equipment that optimizes protocols, such as:

- Proxy ARP and MAC address caching
- Transforming multicast and broadcast protocol requests into unicast requests
- ICMP optimizations
- Routing protocol manipulation
- Manipulating or optimizing any other Internet protocol

If such equipment is placed in the paths between cluster nodes and clients, use a $PING_CLIENT_LIST variable (in **clinfo.rc**) to help inform clients of IP address movements. The specific network topology may require other solutions.

Bridges, hubs, and other passive devices that do *not* modify the packet flow may be safely placed between cluster nodes, and between nodes and clients.

ARP Cache Updating

During manufacturing, every NIC is given a unique hardware address, the *Media Access Control* (MAC) address. The MAC address is the address used by the network drivers to send packets between NICs on the local network. The MAC address *is not* an IP address.

Most TCP/IP systems maintain a list of recently used IP addresses and the corresponding MAC addresses. This list is called an *Address Resolution Protocol (ARP) cache*.

HACMP may be configured such that, if a NIC fails, the IP label on that NIC moves to another NIC. IP labels may also be moved to a NIC on another node if the original node fails. When an IP label moves, its ARP cache entry is then inaccurate. After a cluster event, HACMP nodes and network devices that support *promiscuous listen* automatically update their ARP caches. Clients and network appliances that do *not* support promiscuous listen continue to have incorrect entries. You can manage these addressing updates in one of two ways:

- Use alternate hardware addresses. This function of HACMP assigns an alternate MAC address to NICs with IP labels that may move. When the IP label moves, HACMP moves the MAC address as well. In this case, the ARP cache remains correct.
- Update the ARP cache. Through use of PING_CLIENT_LIST entries in **clinfo.rc**, **clinfo** can update the ARP caches for clients and network devices such as routers.

For more information about these strategies, see the chapter on installing HACMP on client nodes in the *Installation Guide*.

IP Labels

In a non-HACMP environment, a hostname typically identifies a system, with the hostname also being the IP label of one of the network interfaces in the system. Thus, a system can be reached by using its hostname as the IP label for a connection.

Hostnames and Node Names

Typically, the hostname can be the same as the node name. If you use the Standard configuration path in HACMP, HACMP retrieves the hostname from a node when you enter an IP address, an associated IP label, or a fully-qualified domain name (FQDN) as a communication path when adding a node to the cluster, and uses the hostname as the node name, unless you specify otherwise. In the Extended configuration path, you always specify the node name.

In some cases, a node name should be different from the hostname (for example, for an application that requires the interface through which it connects to have an IP label that corresponds to the system hostname).

In a case where an application requires that the AIX "hostname attribute" move with an application to another node at fallover, ensure that HACMP changes the node hostname to correspond to the service IP label when the resource group that contains this application falls over to another node. Use pre- and post-event scripts to change the hostname. For information about event scripts, see Chapter 6: Configuring Cluster Events in the *Administration Guide*.

IP Labels in TCP/IP Networks

For TCP/IP networks, an IP label and its associated IP address must appear in the **/etc/hosts** file.

The name of the service IP label/address must be unique within the cluster and distinct from the volume group and resource group names; it should relate to the application it serves, as well as to any corresponding device, such as `websphere_service_address`.

When you assign a service IP label to an interface, use a naming convention that helps identify the interface's role in the cluster. The related entries in the **/etc/hosts** file would be similar to the following:

```
100.100.50.1   net1_en0
100.100.60.1   net2_en1
```

You configure the NIC by following the instructions in the relevant AIX documentation. AIX assigns an interface name to the NIC when it is configured. The interface name is made up of two or three characters that indicate the type of NIC, followed by a number that AIX assigns in sequence for each adapter of a certain type. For example, AIX assigns an interface name such as **en0** for the first Ethernet NIC it configures, **en1** for the second, and so on.

Cluster Partitioning

Partitioning, also called *node isolation*, occurs when a network or NIC failure isolates cluster nodes from each other. When an HACMP node stops receiving network traffic from another node, it assumes that the other node has failed. Depending on your HACMP configuration, it may begin acquiring disks from the "failed" node, and making that node's applications and IP labels available. If the "failed" node is actually still up, data corruption may occur when the disks are taken from it. If the network becomes available again, HACMP stops one of the nodes to prevent further disk contention and duplicate IP addresses on the network.

You can configure additional networks to help prevent cluster partitioning by:

- Adding point-to-point networks to your network topology.

  For information about using point-to-point networks in an HACMP cluster, see Planning Point-to-Point Networks in this chapter.

- Tuning parameters for the HACMP network module on each node.

  HACMP heartbeats that flow over IP links are sent as UDP datagrams. Therefore, if the network is congested, or a node is congested, the IP subsystem can silently discard the heartbeats.

  For information about setting tuning parameters, see the section Monitoring Clusters in this chapter.

General Network Connection Example

An example of correct HACMP networking consists of two separate Ethernet networks, each with two network interfaces on each node. Two routers connect the networks, and route packets between the cluster and clients, but *not* between the two networks. A **clinfo.rc** file is installed on each node in the cluster, containing the IP addresses of several client machines.

HACMP Configuration in Switched Networks

Unexpected network interface failure events can occur in HACMP configurations using switched networks, if the networks and the switches are incorrectly defined or configured.

Follow these guidelines when configuring switched networks:

- VLANs. If VLANs are used, all interfaces defined to HACMP on a given network must be configured on the same VLAN (one network per VLAN). These interfaces are really the only ones “known” to HACMP, the other interfaces are only known to RSCT Topology Services.

  Topology Services uses other interfaces to determine interface failure, if the failing interface is the last one. So, if multiple interfaces are defined for HACMP on each node, Topology Services can determine interface failure reliably, without requiring interfaces *not* configured for HACMP.
- **Autonegotiation** settings. Some Ethernet NICs are capable of automatically negotiating their speed and other characteristics, such as half or full duplex. Configure NIC *not to use* **autonegotiate**, but to run at the desired speed and duplex value. Set the switch port to which the NIC is connected to the same fixed speed and duplex value.
- ARP settings. Some network switches have settings that can affect ARP responses, either by delaying them, or by providing proxy ARP responses that may *not* be correct. Either of these behaviors can cause unexpected network events. To remedy this situation, change the settings causing the problem, if possible, to ensure that the switch provides a timely response to ARP requests.

  For many brands of switches, this means *turning off* the following: the **spanning tree** algorithm, **portfast**, **uplinkfast**, and **backbonefast**. If it is necessary to have **spanning tree** turned on, then **portfast** should also be turned on.

HACMP and Virtual Ethernet (VLAN)

HACMP 5.4.1 supports Virtual Ethernet (from now on referred to as VLAN, Virtual Local Area Network) with the applicable APARs installed. The following restrictions apply to using VLAN in a cluster configuration:

- In general, we recommend using IPAT via IP Aliasing for all HACMP networks that can support VLANs. IPAT via Replacement and Hardware Address Takeover is *not* supported on a VLAN network.
- If you are using a VLAN network, the PCI Hot Plug utility in HACMP is *not* applicable. This is because the PCI Hot Plug facility uses the VIO (Virtual I/O) Server and the I/O adapters used are virtual (and *not* physical network interface cards).

Note: If you are using VLANs, you may need to use distribution preferences for service IP labels aliases. For more information about the types of distribution preferences for service IP labels, see the section Types of Distribution for Service IP Label Aliases in this chapter.

The following list contains additional configuration requirements for HACMP with Virtual Ethernet:

- If the VIO server has multiple physical interfaces defined on the same network, or if there are two or more HACMP nodes using VIO servers in the same frame, HACMP will *not* be informed of (and therefore will *not* react to) single physical interface failures. This does *not* limit the availability of the entire cluster because VIO server itself routes traffic around the failure. The VIO server support is analogous to EtherChannel in this regard. Use methods that are *not* based on the VIO server to provide notification of individual physical interface failures.
- If the VIO server has only a single physical interface on a network, then HACMP detects a failure of that physical interface. However, the failure will isolate the node from the network. Although some of these considerations may be viewed as configuration restrictions, many are direct consequences of I/O Virtualization.

Troubleshooting VLANs

To troubleshoot Virtual LAN (from now on referred to as VLAN, Virtual Local Area Network) interfaces defined to HACMP and to detect an interface failure, consider these interfaces as interfaces defined on single adapter networks. For information on single adapter networks and the use of the **netmon.cf** file, see the section Identifying Service Adapter Failure for Two-Node Clusters in this chapter.

In particular, list the network interfaces that belong to a VLAN in the **etc/cluster/ping_client_list** or in the PING_CLIENT_LIST variable in the **/usr/es/sbin/cluster/etc/clinfo.rc** script and run **clinfo**. This way, whenever a cluster event occurs, **clinfo** monitors and detects a failure of the listed network interfaces. Due to the nature of VLAN, other mechanisms to detect the failure of network interfaces are *not* effective.

Heartbeating in HACMP

The primary task of HACMP is to recognize and respond to failures. HACMP uses *heartbeating* to monitor the activity of its network interfaces, devices and IP labels.

Heartbeating connections between cluster nodes are necessary because they enable HACMP to recognize the difference between a network failure and a node failure. For instance, if connectivity on the HACMP network (this network's IP labels are used in a resource group) is lost, and you have another TCP/IP based network and a non-IP network configured between the nodes, HACMP recognizes the failure of its cluster network and takes recovery actions that prevent the cluster from becoming partitioned.

To avoid cluster partitioning, we highly recommend configuring redundant networks in the HACMP cluster and using both IP and non-IP networks. Out of these networks, some networks will be used for heartbeating purposes.

In general, heartbeats in HACMP can be sent over:

- TCP/IP networks.
- Serial (non-IP) networks (RS232, TMSCSI, TMSSA and disk heartbeating).

The Topology Services component of RSCT carries out the heartbeating function in HACMP.

The following sections describe heartbeating mechanisms in HACMP:

- Topology Services and Heartbeat Rings
- Heartbeating over IP Aliases
- Heartbeating over Disk

Topology Services and Heartbeat Rings

HACMP uses the Topology Services component of RSCT for monitoring networks and network interfaces. Topology Services organizes all the interfaces in the topology into different *heartbeat rings*. The current version of RSCT Topology services has a limit of 48 heartbeat rings, which is usually sufficient to monitor networks and network interfaces.

Heartbeat rings are dynamically created and used internally by RSCT. They do *not* have a direct, one-to-one correlation to HACMP networks or number of network interfaces. The algorithm for allocating interfaces and networks to heartbeat rings is complex, but generally follows these rules:

- In an HACMP network, there is one heartbeat ring to monitor the service interfaces, and one for each set of non-service interfaces that are on the same subnet. The number of non-service heartbeat rings is determined by the number of non-service interfaces in the node with the largest number of interfaces.
- The number of heartbeat rings is approximately equal to the largest number of interfaces found on any one node in the cluster.

Note that during cluster verification, HACMP calls the RSCT verification API. This API performs a series of verifications, including a check for the heartbeat ring calculations, and issues an error if the limit is exceeded.

Heartbeating over IP Aliases

This section contains the following subsections:

- Overview
- Setting up Heartbeating over IP Aliases
- How HACMP Assigns Heartbeat Rings
- Examples of Heartbeat Rings
- Viewing IP Addresses Assigned by HACMP for Heartbeating over IP Aliases.

Overview

In general, HACMP subnetting requirements can be complicated to understand and may require that you reconfigure networks in AIX to avoid features such as multiple subnet routes, which can lead to a single point of failure for network traffic.

When planning your cluster networks, you may need to:

- Reconfigure IP addresses of HACMP interfaces that will be used at boot time

or

- Update **/etc/hosts** with the new boot time IP addresses.

Heartbeating over IP Aliases is useful because it:

- Uses automatically generated IP aliases for heartbeating

  Heartbeating over IP Aliasing provides an option where the addresses used for heartbeating can be automatically configured by HACMP in a subnet range that is outside of the range used for the base NIC or any service addresses.

  Note: Although Heartbeating over IP Aliasing automatically configures proper aliases for heartbeating, you must still be aware of the implications of subnet routing for all boot and service IP addresses. That is, failure to plan subnets properly can lead to application failures that are *not* detectable by HACMP. Reliable HACMP cluster communication still requires that the interfaces on a single network can communicate with the other nodes on that network.

- Enables you to avoid reconfiguration of boot time addresses and **/etc/hosts**.

  RSCT sets up the heartbeat rings to go over a separate range of IP aliases. This lets you use a specified subnet in a non-routable range for a heartbeat ring, preserving your other subnets for routable traffic. This also allows you to avoid reconfiguring boot time addresses and entries in **/etc/hosts**.

- Makes HACMP topology configuration easier to understand.
- Does *not* require that you obtain additional routable subnets from the network administrator.

For instance, you can use heartbeating over aliases in HACMP, if due to the network system administration restrictions, the IP addresses that your system can use at boot time must reside on the same subnet. (In general, if there are *no* system administration restrictions, the IP addresses that your system can use at boot time can reside on either the same or *different* subnets).

Setting up Heartbeating over IP Aliases

Although Heartbeating over IP Aliasing automatically configures proper aliases for heartbeating, you must still be aware of the implications of subnet routing for all boot and service IP addresses. That is, failure to properly plan subnets can lead to application failures that are *not* detectable by HACMP. Once you are confident that you understand the subnetting requirements and that Heartbeating over IP Aliasing is appropriate to use, use the instructions in this section and in the *Administration Guide*.

To set up heartbeating over IP aliases, use SMIT to set an **IP Address Offset for Heartbeating over IP Aliases** as part of the network configuration.

The IP Address Offset for Heartbeating over IP Aliases has the following characteristics:

- HACMP uses it as a *starting address* to set up the range of IP aliases on multiple subnets on all the interfaces in the cluster network. The subnet mask is the same as the one used for the network.
- The range of IP aliases should *not* overlap with any addresses that are already used on this network.

  In other words, the starting address that you select must reside on a subnet that is *not* used by any other network in your physical network setup, and you must have enough available subnets above the one you type in for N networks, where N is the number of interfaces that each node has on the network. HACMP verifies this during the cluster verification process.
- The IP alias addresses in the range are only used by HACMP for heartbeats. They do *not* need to be routed and should *not* be used for any other traffic.
- HACMP Configuration Database HACMP adds the IP aliases to the network interfaces at cluster startup and during the cluster verification and synchronization.
- HACMP also gives RSCT the addresses to use for heartbeating. At shutdown, HACMP removes the aliases from the interfaces.

To activate heartbeating over IP aliases in HACMP, you:

1. Add an HACMP network using the **Extended Configuration** path in SMIT *and* specify for this network an IP Address Offset for Heartbeating over IP Aliases.
2. Add an IP Address Offset to an existing network using the Extended HACMP Configuration path. The netmask used for the IP alias address will be the same as the netmask of the underlying interfaces.

To deactivate heartbeating over aliases, remove the IP Address Offset from the network.

How HACMP Assigns Heartbeat Rings

This section explains how HACMP manages IP aliases and service labels used for heartbeating over IP aliases. Heartbeating over aliases can be used with either type of IPAT: IPAT via IP aliasing or IPAT via IP replacement:

- With IPAT via replacement, during fallover, the service IP label is swapped with the boot time address, *not* with the heartbeating alias IP address.
- With IPAT via aliasing, during fallover, the service IP label is aliased onto the boot interface along with the heartbeat alias.

HACMP assigns heartbeating aliases based on the interface address:

- For each interface in the network on a node, HACMP increments the subnet address.
- For each node in the network, HACMP increments the host address.

Examples of Heartbeat Rings

For example, if you specify in SMIT the IP Address Offset 192.168.1.1, HACMP builds the following heartbeating rings:

192.168.1.1 <---------> 192.168.1.2

192.168.2.1 <---------> 192.168.2.2

The following diagram displays these heartbeat rings:

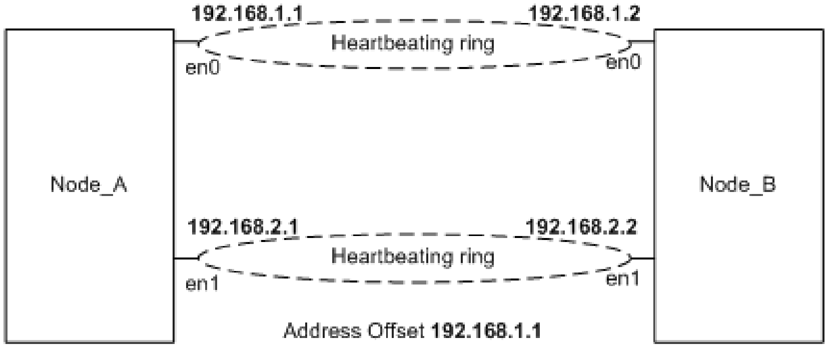


Figure 2. Heartbeating over IP Aliases: Heartbeating Rings

The following table displays an example of configured IP addresses:

IP Address Example	Role in HACMP	Action by HACMP
1.1.1.1	service IP alias	monitored by HACMP
192.168.1.1.	IP alias used for heartbeating	monitored by HACMP
2.2.2.1	IP address used at boot time	*not* monitored by HACMP

Note: In the previous table, the base addresses (such as 2.2.2.1) are *not* used for heartbeating traffic, but the health of the underlying NIC is still monitored using the heartbeating via IP alias address. Any failure of the NIC alerts HACMP to recover any service and persistent labels that are currently configured on that NIC.

As another example, you can use `10.10.10.1` as the IP Address Offset (starting address). If you have a network with two NICs on each node, and a subnet mask of `255.255.255.0`, HACMP assigns with the following heartbeat IP aliases:

Node A:

For network `en0, boot_label = nodeA_boot`

`hb_IP_alias_label1 = 10.10.10.1`

For network `en1, boot_label = nodeA_boot2`

`hb_IP_alias_label2 = 10.10.11.1`

Similarly, for Node B:

`hb_IP_alias_label1 = 10.10.10.2`

`hb_IP_alias_label2 = 10.10.11.2`

The HACMP cluster verification utility checks that you have a valid configuration for the address range. HACMP verifies that:

- All interfaces have the same netmask and the same type.
- The IP Offset Address allows for enough addresses and subnets.

Note: Verification does *not* check the interfaces or subnet requirements for a heartbeating over IP aliases configuration because they use a separate address range.

Viewing IP Addresses Assigned by HACMP for Heartbeating over IP Aliases

The IP aliases used for heartbeating show up when you run AIX commands such as **netstat**.

Heartbeating over Disk

You can configure a non-IP point-to-point heartbeating network, called a *disk heartbeating network*, over any shared disk in an enhanced concurrent mode volume group. Heartbeating over disk provides another type of non-IP point-to-point network for failure detection.

Disk heartbeating networks provide an alternative to other point-to-point networks such as RS232 that have cable length restrictions, or TMSSA which require special disk adapter hardware and cabling. Heartbeating over disk does *not* require additional or specialized hardware, cabling or microcode; it can use any disk that is also used for data and for which volume groups and file systems are included in an HACMP resource group.

In a disk heartbeating network, two nodes connected to the disk periodically write heartbeat messages and read heartbeat messages (written by the other node) on a small, non-data portion of the disk. A disk heartbeating network, like the other non-IP heartbeating networks, connects only two nodes. In clusters with more than two nodes, use multiple disks for heartbeating. Each node should have a non-IP heartbeat path to at least one other node. If the disk heartbeating path is severed, at least one node cannot access the shared disk.

You have two different ways for configuring a disk heartbeating network in a cluster:

- You can create an enhanced concurrent volume group shared by multiple nodes in your cluster. Then you use the HACMP Extended Configuration SMIT path to configure a point-to-point pair of discovered communication devices.

  or

- You can start by creating a cluster disk heartbeating network, and then add devices to it using the **Add Pre-Defined Communication Interfaces and Devices** panel in SMIT.

The HACMP cluster verification utility verifies that the disk heartbeating networks are properly configured.

For information on testing a disk heartbeating network for connectivity, see the section RSCT Command for Testing Disk Heartbeating in Appendix C in the *Administration Guide*.

For information on checking the disk heartbeating network, see the *Troubleshooting Guide.*

Heartbeating over Disk and Fast Method for Node Failure Detection

With HACMP 5.4.1, you can reduce the time it takes for node failure to be realized throughout the cluster. If you have a disk heartbeating network configured, and specify a parameter for a disk heartbeating NIM, then when a node fails, HACMP 5.4.1 uses a disk heartbeating network to place a departing message on the shared disk so neighboring nodes are aware of the node failure within one heartbeat period. For more information, see the section Decreasing Node Fallover Time in this chapter.

For information on enabling fast failure detection in HACMP, see the chapter on configuring NIMs in the *Administration Guide*.

Designing the Network Topology

The combination of IP and non-IP (point-to-point) networks that link cluster nodes and clients is called the cluster *network topology*. The HACMP software supports a large number of IP and point-to-point devices on each node to provide flexibility in designing a network configuration.

When designing your network topology, ensure that clients have highly available network access to their applications. This requires that *none* of the following is a single point of failure:

- The IP subsystem
- A single network
- A single NIC.

Eliminating the IP Subsystem as Single Points of Failure

If the IP software fails on a node, you lose all IP communications to and from the node. You can increase availability by configuring non-IP point-to-point connections that directly link cluster nodes. This provides an alternate heartbeat path for the cluster, and it prevents the IP software from being a single point of failure.

Non-IP network heartbeat rings protect the cluster from getting into an invalid state because of possible network contention issues. UDP packets that carry the heartbeat packet over IP networks will be dropped if there is a large amount of network contention on the wire where the heartbeating is occurring.

For information about using non-IP point-to-point networks, see the section Planning Point-to-Point Networks in this chapter.

Eliminating Networks as Single Points of Failure

In a single-network setup, each node in the cluster is connected to only one network and has a single service interface available to clients. In this setup, the network is a single point of failure for the entire cluster, and each service interface is a single point of failure. The following diagram shows a single-network configuration:

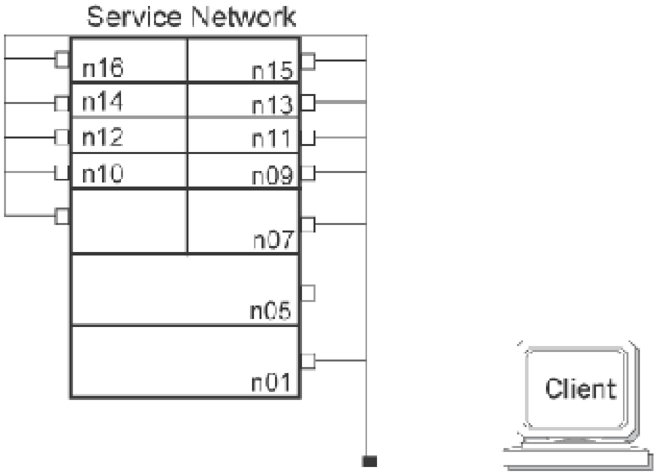


Figure 3. Single-Network, Single-NIC Setup

To eliminate the network as a single point of failure, configure multiple networks so that HACMP has multiple paths among cluster nodes. Keep in mind that if a client is connected to only one network, that network is a single point of failure for the client. In a multiple-network setup if one network fails, the remaining network(s) can still function to connect nodes and provide access for clients.

The more networks you can configure to carry heartbeats and other information among cluster nodes, the greater the degree of system availability. Non-IP point-to-point networks are often used as an alternative heartbeat path. For information about planning point-to-point networks, see the section Planning Point-to-Point Networks in this chapter.

The following diagram illustrates a dual-network setup with more than one path to each cluster node:

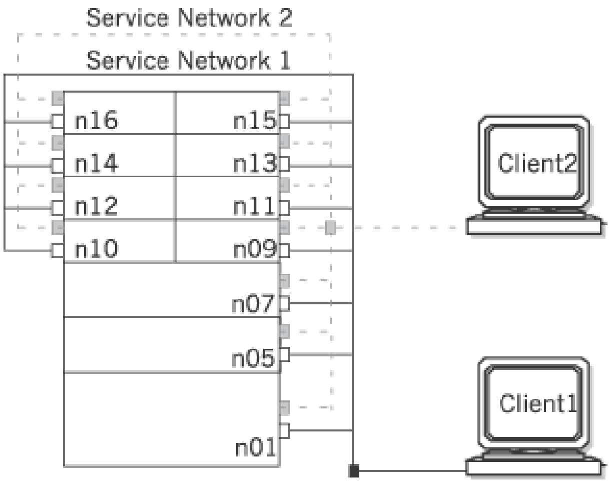


Figure 4. Dual-Network Setup

Note: Hot-replacement of the dual-port Ethernet adapter used to configure two interfaces for one HACMP IP network is currently *not* supported.

Planning Point-to-Point Networks

You can also increase availability by configuring non-IP point-to-point connections that directly link cluster nodes. These connections provide:

- An alternate heartbeat path for a cluster that uses a single TCP/IP-based network, and prevent the TCP/IP software from being a single point of failure
- Protection against cluster partitioning.

  For more information about cluster partitioning, see the section Cluster Partitioning in this chapter.

You can configure heartbeat paths over the following types of networks:

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

Selecting a Layout for Point-to-Point Networks

Because point-to-point networks play an important role in ensuring high availability of networks, select connections that create adequate paths in clusters that contain more than two nodes. The following table briefly lists the configuration types and describes the advantages or disadvantages of each:

Type	Description	Considerations to Keep in Mind
Mesh	Connects each node in the cluster to all other nodes in the cluster	Provides the most robust and reliable configuration
Star	Connects one node with all other nodes	May create partitioned clusters when multiple failures occur simultaneously
Ring or Loop	Connects each node to directly adjacent neighbors	

The type of point-to-point networks you include in your network topology depends on the hardware available and the requirements for your cluster.

Planning Serial Point-to-Point Networks

When planning a serial (RS232) network, keep in mind the following:

- If there are no serial ports available, and your planned HACMP configuration for that node uses an RS232 network, the configuration requires a serial NIC card.
- All RS232 networks defined to HACMP are brought up by RSCT with a default of 38400 bps. The tty ports should be defined to AIX as running at 38400 bps.

  RSCT supports baud rates of 38400, 19200, 9600.

Any serial port that meets the following requirements can be used for heartbeats:

- The hardware supports use of that serial port for modem attachment.
- The serial port is free for HACMP exclusive use.

Examples of processors with native serial ports that do *not* meet these conditions are S70, S7A, S80, and serial ports 1 and 2 in the F50, H50, and H70.

Certain RS/6000 systems do *not* support the use of native serial ports.

Note: HACMP supports serial ports that the hardware and the system software make available for application use. It is your responsibility to manage any modems or extenders between the ports.

Refer to the hardware documentation and HACMP support announcements to determine whether your serial ports meet the requirements.

The following figure shows four RS232 networks, in a ring configuration, connecting four nodes to provide full cluster non-IP connectivity:

Figure 5. Serial Network Configuration in an HACMP Cluster

Planning Disk Heartbeating Networks

Any shared disk in an enhanced concurrent mode volume group can support a point-to-point heartbeat connection. Each disk can support one connection between two nodes. The connection uses the shared disk hardware as the communication path.

A disk heartbeating network in a cluster contains:

- Two nodes

  A node may be a member of any number of one disk heartbeating networks. A cluster can include up to 256 communications devices.

- An enhanced concurrent mode disk that participates in only one heartbeat network.

  Keep in mind the following points when selecting a disk to use for disk heartbeating:

  A disk used for disk heartbeating must be a member of an enhanced concurrent mode volume group. However, the volume groups associated with the disks used for disk heartbeating do *not* have to be defined as resources within an HACMP resource group. In other words, an enhanced concurrent volume group associated with the disk that enables heartbeating *does not* have to belong to any resource group in HACMP.

  You can convert an existing volume group to enhanced concurrent mode. For information about converting a volume group, see Chapter 12: Managing Shared LVM Components in a Concurrent Access Environment in the *Administration Guide*.

- The disk should have fewer than 60 seeks per second at peak load. (Disk heartbeats rely on being written and read within certain intervals.)

Use the AIX **filemon** command to determine the seek activity, as well as the I/O load for a physical disk.

Typically, most disk drives that *do not* have write caches can perform about 100 seeks per second. Disk heartbeating uses 2 to 4 seeks.

Disks that are RAID arrays, or subsets of RAID arrays, may have lower limits. Check with the disk or disk subsystem manufacturer to determine the number of seeks per second that a disk or disk subsystem can support.

However, if you choose to use a disk that has significant I/O load, increase the value for the timeout parameter for the disk heartbeating network.

- When SDD is installed and the enhanced concurrent volume group is associated with an active vpath device, ensure that the disk heartbeating communication device is defined to use the **/dev/vpath** device (rather than the associated **/dev/hdisk** device).
- If a shared volume group is mirrored, at least one disk in each mirror should be used for disk heartbeating.

  Make sure to set up heartbeating in this way when you plan to set the forced varyon option for a resource group. For more information about using HACMP forced varyon, see the section Using Quorum and Varyon to Increase Data Availability in Chapter 5: Planning Shared LVM Components.

Planning Target Mode Networks

Target mode SCSI and target mode SSA are also supported for point-to-point heartbeat communications. Each of these types of networks includes two nodes, a shared disk, and SCSI or SSA communications (as appropriate to the disk type).

Extending the Distance of Point-to-Point Networks

HACMP point-to-point networks can be extended over longer distances through the use of extender technologies:

Network Type	Extension Mechanism
SCSI	SCSI Bus Extenders
SSA	SSA Optical Extenders
RS232	• Optical line drivers • Short-haul modems • PTT-supplied RS232 lines

When planning long-distance point-to-point network devices choose connections, such as fibre or copper cabling, within the operational tolerances at the distances required.

When extending RS232 links, ensure that the device:

- Supports the full RS232 protocol (all pins)
- Is capable of supporting the distance required at a minimum line speed of 9600 bps (the default is 38400 bps).

Some RS232 networks that are extended to longer distances require that the baud rate be lower than the default of 38400 bps. For details, see the section Changing an RS232 Network Module Baud Rate in Chapter 13: Managing the Cluster Topology in the *Administration Guide*.

The Failure Detection Rate may also need to be adjusted. For more information, see the section Setting Failure Detection Parameters in this chapter.

Configuring Global Networks

If you have multiple physical networks of the same type, each spanning only a portion of the cluster, you can group these networks into one *global HACMP network*. HACMP will route its heartbeat packets between the different networks, giving it another level of network redundancy, and thereby reducing the probability of cluster partitioning.

Note: Networks configured to use any form of IP address takeover can*not* be included in a global network.

Creating global networks is useful on large SP systems, as each SP frame of nodes is usually set up as a different subnet on the SP Administrative Ethernet. Each of these subnets is then defined as a different HACMP network. Defining a global network that includes all these SP administrative Ethernets will avoid the possibility of cluster partitioning. For more information, see the section Planning for SP Networks in this chapter.

Eliminating Network Interface Cards as a Single Point of Failure

A network interface card (NIC) physically connects a node to a network. When configured with a single NIC per network, the NIC becomes a potential single point of failure. To remedy this problem, configure a node with at least two NICs for each network to which it connects. In the following figure, each cluster node has two connections to each network:

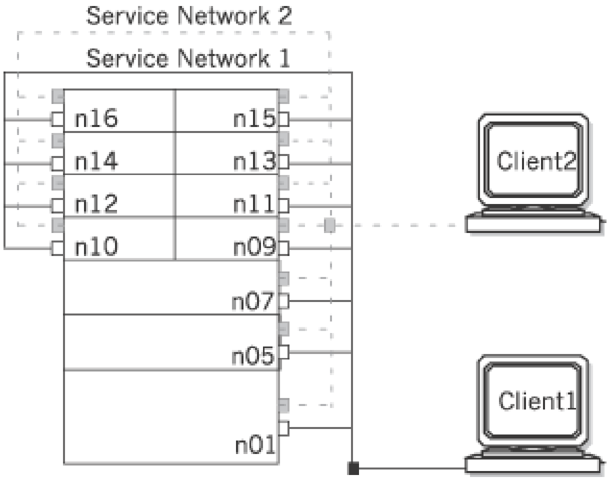


Figure 6. Dual-Network, Dual-NIC Configuration

For more information about the actions HACMP takes in response to NIC failures, see Appendix B: Resource Group Behavior during Cluster Events in the *Administration Guide*.

Note: Hot-replacement of the dual-port Ethernet adapter used to configure two interfaces for one HACMP IP network is currently *not* supported.

Network Interface Functions

When a node is configured with multiple connections to a single network, the network interfaces serve different functions in HACMP:

- The *service* interface
- The *non-service* interfaces.

You can also define a *persistent* node IP interface as an IP alias to a cluster network on a node.

Service Interface

A *service interface* is a communications interface configured with an HACMP service IP label. This interface serves as each node's primary HACMP connection to each network. The service IP label is used in the following ways:

- By clients to access application programs
- For HACMP heartbeat traffic.

Non-service Interface

A *non-service interface* is a communications interface with an HACMP non-service IP label that backs up a service interface. All client traffic is carried over the service interface. Non-service interfaces are hidden from client applications and carry only internal HACMP traffic. If a service interface fails, HACMP can move the service IP label onto a non-service interface. Using a non-service interface eliminates a network interface as a single point of failure.

In addition, if a node fails, the cluster can use a non-service interface on another cluster node as a location for its service IP label when performing a resource group fallover.

A node can have from zero to seven non-service interfaces for each network to which it connects. Your software configuration and hardware constraints determine the actual number of non-service interfaces that a node can support.

Persistent Node IP Label

A persistent node IP label is an IP alias that can be assigned to a specific node on a cluster network. A persistent node IP label:

- Always stays on the same node (is *node-bound*)
- Coexists on a NIC that already has a service or non-service IP label defined
- Does *not* require installing an additional physical NIC on that node
- Is *not* part of any resource group.

Assigning a persistent node IP label provides a node-bound address that you can use for administrative purposes, because a connection to a persistent node IP label always goes to a specific node in the cluster. You can have one persistent node IP label per network per node.

As one of HACMP's best practices, we recommend that you configure a persistent IP label for each cluster node. This is useful, for instance, if you must access a particular node in an HACMP cluster for purposes of running reports or for diagnostics. Having a persistent IP label configured has the advantage that HACMP can access the persistent IP label on the node despite individual NIC failures (provided there are spare NICs on the network).

After a persistent node IP label is configured on a specified network node, it becomes available at boot time and remains configured even if HACMP is shut down on that node.

You can create persistent node IP labels on the following types of IP-based networks:

- Ethernet
- Token Ring
- FDDI
- ATM LAN Emulator.

You cannot configure a persistent node IP label on the SP Switch, on ATM Classical IP, or on a serial cluster network.

The following list describes how HACMP responds to a failure when a persistent node IP label is configured:

- If a NIC that has a service IP label configured fails, and there is also a persistent label defined on this NIC, then the persistent label falls over to the same non-service interface to which the service IP label falls over.
- If all NICs on the cluster network on a specified node fail, then the persistent node IP label becomes unavailable. A persistent node IP label always remains on the same network, and on the same node; it *does not move* between the nodes in the cluster.

For more information about persistent node IP labels/addresses, see the section Configuring HACMP Persistent Node IP Labels/Addresses in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

IP Address Takeover via IP Aliases

HACMP IP networks have IPAT via IP Aliases enabled by default. When HACMP is started on the node, the service IP label is aliased onto one of the non-service interfaces defined to HACMP. If that interface fails, the service IP label is aliased onto another interface if one is available on the same network. Hardware Address Takeover (HWAT) is *not* possible in this configuration. In order to use IPAT via IP Aliases, the network must support gratuitous ARP.

For complete information on IPAT via IP Aliases, see Planning for IP Address Takeover via IP Aliases.

Subnet Considerations for Persistent Node IP Labels

When you configure a persistent node IP label on a cluster network configured with IP Address Takeover via IP Aliases, the IP address associated with a persistent IP label must be on a different subnet than any service address with which it may share an interface.

In some situations you may need to configure a persistent IP label on the same subnet as the service IP label. In this case, to avoid problems with network packets sent from either of the addresses, consider configuring the distribution preference for service IP aliases. This preference is available in HACMP 5.1 and up and, if configured, lets you configure the type of the distribution preference suitable for the VPN firewall external connectivity requirements.

For more information about the types of distribution preferences for service IP labels, see the section Types of Distribution for Service IP Label Aliases in this chapter.

Note: The subnet considerations are different if you are planning to configure NFS file systems. For information, see NFS Cross-Mounting and IP Labels in Chapter 5: Planning Shared LVM Components.

IP Address Takeover via IP Replacement

IP Address Takeover can be set up to use IP Replacement (by disabling the IPAT via IP Aliases option in SMIT). In this configuration, if a node fails, another node takes over the service IP label as part of a resource group takeover. This service IP label will be configured onto one of the takeover node's non-service interfaces (on the appropriate network) in place of the non-service IP label.

Hardware Address Takeover (HWAT) and IPAT via IP Replacement

This configuration supports Hardware Address Takeover (HWAT) through alternate hardware addresses. If HWAT is *not* used, ensure that clients and network equipment get their ARP caches updated after IPAT occurs, and after fallback, if necessary.

Subnet Considerations for Persistent Node IP Labels

When you configure a persistent node IP label on a cluster network configured with IP Address Takeover via IP Replacement, the subnet you use must either be the subnet used by the service IP label, or unique to that network on that node.

Planning for IP Address Takeover via IP Aliases

You can configure IP address takeover on certain types of networks using the IP aliases network capabilities supported in AIX. Assigning IP aliases to NICs allows you to create more than one IP label on the same network interface.

HACMP allows the use of IPAT via IP Aliases with the following network types that support gratuitous ARP in AIX:

- Ethernet
- Token Ring
- FDDI
- SP Switch1 and SP Switch2

Note: IPAT via IP Aliases is *not* supported on ATM networks.

During IP address takeover via IP aliases, when an IP label moves from one NIC to another, the target NIC receives the new IP label as an IP alias *and* keeps the original IP label and hardware address.

Configuring networks for IPAT via IP Aliases simplifies the network configuration in HACMP. You configure a service address and one or more non-service addresses for NICs.

This section contains the following topics:

- Assigning IP Labels for IPAT via IP Aliases
- Planning for Service IP Label Alias Placement
- Types of Distribution for Service IP Label Aliases

- Planning for Site-Specific Service IP Labels.

Assigning IP Labels for IPAT via IP Aliases

IP address takeover via IP aliases is an alternative method of taking over the IP address that allows one node to acquire the IP label and the IP address of another node in the cluster using IP aliases.

To enable IP Address Takeover via IP aliases, configure NICs to meet the following requirements:

- At least one boot-time IP label must be assigned to the service interface on each cluster node.
- Hardware Address Takeover (HWAT) *cannot be configured* for any interface that has an IP alias configured.
- Subnet requirements:
  - Multiple boot-time addresses configured on a node should be defined on different subnets.
  - Service addresses must be on a different subnet from *all* non-service addresses defined for that network on the cluster node. This requirement enables HACMP to comply with the IP route striping function of AIX, which allows multiple routes to the same subnet.
- Service address labels configured for IP address takeover via IP aliases can be included in all non-concurrent resource groups.
- Multiple service labels can coexist as aliases on a given interface.
- The netmask for all IP labels in an HACMP network must be the same.
- You can*not* mix aliased and non-aliased service IP labels in the same resource group.

HACMP non-service labels are defined on the nodes as the boot-time address, assigned by AIX after a system reboot and before the HACMP software is started. When the HACMP software is started on a node, the node’s service IP label is added *as an alias* onto one of the NICs that has a non-service label.

When you configure IPAT via IP Aliases, the node’s NIC meets the following conditions:

- The NIC has *both* the boot-time and service IP addresses configured, where the service IP label is an alias placed on the interface
- Unlike in IPAT via IP Replacement, the boot-time address is never removed from a NIC
- If the node fails, a takeover node acquires the failed node’s service address *as an alias* on one of its non-service interfaces on the same HACMP network.

Note: During a node fallover event, the service IP label that is moved is placed as an alias on the target node’s NIC *in addition* to any other service labels that may already be configured on that NIC.

For example, if Node A fails, Node B acquires Node A’s service IP label. This service IP label is placed as an alias onto the appropriate NIC on Node B, and any other existing labels remain intact on Node B’s NIC. Thus, a NIC on Node B now receives client traffic directed to Node A’s service address. Later, when Node A is restarted, it comes up on its boot-time address(es) and attempts to reintegrate into the cluster by requesting that Node B release Node A’s service IP label.

When Node B releases the requested service IP label, the alias for the service IP labels is deleted on Node B, and Node A once again puts it as an alias onto one of its non-service interfaces on the appropriate network.

When using IPAT via IP Aliases, service IP labels are acquired using all available non-service interfaces. If there are multiple interfaces available to host the service IP label, the interface is chosen according to the number of IP labels currently on that interface. If multiple service IP labels are acquired and there are multiple interfaces available, the service IP labels are distributed across all the available interfaces.

If you remove the boot-time address with the **ifconfig delete** command, even though the interface can have working aliased service IP labels on it, Topology Services detects a local NIC failure. This is done because the boot-time address that is being monitored for heartbeats is no longer available.

Note: If you plan to have an NFS file system and use IPAT via IP Aliases in the cluster, see the section NFS Cross-Mounting and IP Labels in Chapter 5: Planning Shared LVM Components.

For information about configuring networks to use IPAT via IP Aliases, see the section Configuring IP-Based Networks in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) of the *Administration Guide*.

Planning for Service IP Label Alias Placement

If you use IPAT via IP Aliases, you can configure a distribution preference for the placement of service IP labels that are configured in HACMP.

HACMP lets you specify the distribution preference for the service IP label aliases. These are the service IP labels that are part of HACMP resource groups and that belong to IPAT via IP Aliases networks.

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.

We recommend using the distribution preference for IP aliases to address the following cluster requirements:

- Firewall considerations
- Cluster configurations that use VLANs (when applications are expecting to receive packets from a specific network interface)
- Specific requirements for the placement of IP labels in the cluster.

Distribution Preference for Service IP Label Aliases: How It Works

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.
- Enables HACMP to redistribute the alias service IP labels according to the preference you specify.

- Allows you to configure the type of the 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 in the cluster as long as there are acceptable network interfaces available. HACMP always keeps service IP labels active, even if the preference can*not* be satisfied.

Types of Distribution for Service IP Label Aliases

You can specify in SMIT the following distribution preferences for the placement of service IP label aliases:

Type of distribution preference	Description
Anti-collocation	This is the default. HACMP distributes all service IP label aliases across all boot IP labels using a “least loaded” selection process.
Collocation	HACMP allocates all service IP label aliases on the same network interface card (NIC).
Anti-collocation with persistent	HACMP distributes all service IP label aliases across all active physical interfaces that are *not* hosting the persistent IP label. HACMP will place the service IP label alias on the interface that is hosting the persistent label only if no other network interface is available. If you did *not* configure persistent IP labels, HACMP lets you select the Anti-Collocation with Persistent distribution preference, but it issues a warning and uses the regular anti-collocation preference by default.
Collocation with persistent	All service IP label aliases are allocated on the same NIC that is hosting the persistent IP label. This option may be useful in VPN firewall configurations where only one interface is granted external connectivity and all IP labels (persistent and service) must be allocated on the same interface card. If you did *not* configure persistent IP labels, HACMP lets you select the Collocation with Persistent distribution preference, but it issues a warning and uses the regular collocation preference by default.

The following rules apply to the distribution preference:

- If there are insufficient interfaces available to satisfy the preference, HACMP allocates service IP label aliases and persistent IP labels to an existing active network interface card.
- If you did *not* configure persistent labels, HACMP lets you select the Collocation with Persistent and Anti-Collocation with Persistent distribution preferences, but it issues a warning and uses the regular collocation or anti-collocation preferences by default.
- You can change the IP labels distribution preference dynamically: the new selection becomes active during subsequent cluster events. HACMP does *not* interrupt the processing by relocating the currently active service IP labels at the time the preference is changed.

When a service IP label fails and another one is available on the same node, HACMP recovers the service IP label aliases by moving them to another network interface card on the same node. During this event, the distribution preference that you specified remains in effect.

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

Planning for Site-Specific Service IP Labels

You can have a service IP label configurable on multiple nodes, associated with a resource group than can move between nodes or sites. Since an IP address that is valid at one site may *not* be valid at the other site due to subnet issues, you can associate a service IP label that is configurable on multiple nodes with a specific site. *Site-specific service IP labels* are configured in HACMP and can be used with or without XD-type networks. This label is associated with a resource group and is active only when the resource group is in *online primary* state at the associated site.

Planning for IP Address Takeover via IP Replacement

If you disable the default option to configure the network with IP Address Takeover via IP Aliases, you can use IP Address Takeover via IP Replacement. This method uses a service IP label and one or more non-service IP labels per node for a given network. At boot, AIX comes up with a configured IP address on each NIC. When HACMP is started, it replaces non-service IP labels with service IP labels for the resource groups it brings online.

If the NIC that is hosting the service IP label fails, and there is another non-service interface for that network on the node, HACMP will replace that non-service IP label with the service IP label, in order to maintain the service IP label.

Each node on the network, including client nodes, maintains an ARP cache that maps each IP address to a particular hardware (MAC) address. After HACMP moves a service IP label and the address associated with it to a different NIC, these ARP cache entries must be updated so that the service address correctly matches the hardware address of the target NIC.

AIX does a gratuitous ARP when this reconfiguration occurs. This allows clients and network devices that support `promiscuous listen` mode to update their ARP caches. If your clients do *not* support `promiscuous listen` mode, you can either use **clinfo** to update their ARP caches, or use Hardware Address Takeover (HWAT).

All non-concurrent resource groups can be configured with IP labels on IPAT via IP Replacement networks.

If you are using an IPAT via IP Replacement network and plan on configuring a non-concurrent resource group, you may also consider using the node distribution policy for the resource group's startup. For more information, see Node Distribution Policy in Chapter 6: Planning Resource Groups.

Hardware Address Takeover with IPAT via IP Replacement

Certain types of networks cannot process ARP cache updates. Also, some clients and network devices do *not* support `promiscuous listen`. To eliminate these ARP cache difficulties, HACMP uses the Hardware Address Takeover (HWAT) facility of AIX. HWAT *not* only moves the service IP label, but also moves the underlying hardware (MAC) address of the target NIC so that it remains with the service IP label. This way, as both the IP address and the hardware address are moved, the ARP cache on the node remains correct. For more information, see the section Selecting an Alternate Hardware Address in this chapter.

Note: Turn off flow control on the gigabit Ethernet adapters. This can cause network problems after an HACMP fallover.

Planning for Other Network Conditions

The most typical network planning consideration involve nameserving in an HACMP environment, and setting up cluster monitoring and failure detection.

This section contains the following topics:

- Using HACMP with NIS and DNS
- Monitoring Clusters
- Planning for VPN Firewall Network Configurations in HACMP
- Setting Failure Detection Parameters
- Setting Values for the Network Grace Period
- Identifying Service Adapter Failure for Two-Node Clusters
- Setting RS232 TTY Baud Rates
- Planning Networks for Inter-Node Communication with Oracle.

Using HACMP with NIS and DNS

Some of the commands used to troubleshoot network and interface problems after a failure require IP lookup to determine the IP address associated with a specified IP label.

If NIS or DNS is in operation, IP lookup defaults to a nameserver system for name and address resolution. However, if the nameserver was accessed through an interface that has failed, the request does *not* complete, and eventually times out. This may significantly slow down HACMP event processing.

To ensure that cluster event completes successfully and quickly, HACMP *disables NIS or DNS hostname resolution* by setting the following AIX environment variable during service IP label swapping:

```
NSORDER = local
```

As a result, the **/etc/hosts** file of each cluster node must contain all HACMP defined IP labels for all cluster nodes.

DNS Requests Sent from Non-HACMP Processes

Disabling NIS or DNS hostname resolution is specific to the HACMP event script environment. HACMP sets the NSORDER variable to *local* when it attaches a service IP label and when it swaps IP labels on an interface.

Other processes continue to use the default system name resolution settings (for example, applications outside of HACMP that require DNS IP address resolution). If these processes request IP lookup, then during the network interface reconfiguration events in HACMP the processes may still *not* be able to contact an external name server. The request to the DNS will succeed *after* HACMP completes the network interface reconfiguration event.

Monitoring Clusters

Each supported cluster network has a corresponding cluster network module that monitors all I/O to its cluster network. The network modules maintain a connection to each other in a cluster. The Cluster Managers on cluster nodes send messages to each other through these connections.

Each network module sends periodic heartbeat messages to and receives periodic heartbeat messages from other network modules in the cluster to:

- Monitor interfaces
- Ensure connectivity to cluster peers
- Report when a connection fails.

Currently, HACMP network modules support communication over the following types of networks:

- Serial (RS232)
- disk heartbeating (over enhanced concurrent mode disks)
- Target-mode SCSI
- Target-mode SSA
- Ethernet
- Token Ring
- FDDI
- SP Switch
- ATM.

Planning for VPN Firewall Network Configurations in HACMP

Starting with HACMP 5.1, HACMP lets you specify the distribution preference for the service IP label aliases. These are the service IP labels that are part of HACMP resource groups and that belong to IPAT via IP Aliasing networks.

Certain VPN firewall configurations allow external connectivity to only one NIC at a time. If your firewall is configured this way, allocate all HACMP service and persistent IP labels on the same interface.

To have HACMP manage the IP labels to satisfy the requirements of such a VPN firewall:

- Specify the persistent IP label for each node in the cluster. The persistent IP label is mapped to an available interface on the selected network.
- Use IPAT via IP Aliases for the network that contains the service IP labels (this is the default). That is, when configuring service IP labels as resources in a resource group, ensure that the **Enable IP Address Takeover via IP Aliases** field is set to **Yes** under the **Configure HACMP Networks** SMIT panel.
- Specify the Collocation with Persistent distribution preference for the network containing the service IP labels. This ensures that all service IP label aliases are allocated on the same physical interface that is hosting the persistent IP label.

For an overview of types of distribution preferences you can specify in SMIT, see Types of Distribution for Service IP Label Aliases in this chapter. For configuration instructions, see the *Administration Guide.*

Setting Failure Detection Parameters

An important tuning parameter for network modules is the Failure Detection Rate, which determines how quickly a connection is considered to have failed.

The Failure Detection Rate consists of two components:

- *Cycles to fail (cycle)*. The number of heartbeats missed before detecting a failure
- *Heartbeat rate (hbrate)*. The number of seconds between heartbeats.

The time needed to detect a failure can be calculated using this formula:

```
(heartbeat rate) X (cycles to fail) X 2
```

The Failure Detection Rate can be changed for a network module in two ways:

- Select the preset rates of **slow**, **normal** or **fast**
- Change the actual components **cycle** or **hbrate**.

The preset values are calculated for each type of network to give reasonable results. Use the preset rate values (**Slow**, **Normal** or **Fast**) if you change your Failure Detection Rate. For information about customizing the Failure Detection Rate components, see the section Changing a Failure Detection Rate in this chapter.

You may want to consider changing the Failure Detection Rate to:

- Decrease fallover time
- Keep node CPU saturation from causing false takeovers.

Decreasing Fallover Time

The default setting for the failure detection rate is usually optimal. You can slightly speed up fallover by speeding up failure detection. However, the amount and type of customization you add to event processing has a much greater impact on the total fallover time. You should test the system for some time before deciding to change the failure detection speed of any network module.

Decreasing Node Fallover Time

HACMP 5.4.1 reduces the time it takes for a node failure to be realized throughout the cluster, while reliably detecting node failures.

HACMP 5.4.1 uses disk heartbeating to place a *departing* message on a shared disk so neighboring nodes are aware of the failed node within one heartbeat period (hbrate). 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.

The failure detection rates of **Fast**, **Normal** and **Slow** contain hbrates of 1, 2, or 3 seconds respectively. Therefore, if you are using disk heartbeating, the time for the neighbor nodes to determine if the node is down would be at most 1, 2 or 3 seconds, followed by the other cluster nodes immediately becoming aware of the failure.

For example, using the formula to calculate the time needed to detect a failure:

```
(heartbeat rate) X (cycles to fail) X 2
```

For a heartbeat rate of 2 seconds and cycles to fail of 12, the adapter failure detection time is 48 seconds compared to the fast method of node failure detection rate of 4 seconds.

Fast Node Failure Detection Prerequisites

Full support for the fast node failure detection method requires:

- HACMP 5.4.1
- AIX v.5.3 (the latest available maintenance level)
- All supported disks except for SSA disks. Fast method for node failure detection is not supported on SSA disks.

If you have an entire HACMP 5.4.1 cluster, but some nodes are *not* running AIX 5.3, you can still benefit from fast node failure detection method: If an AIX v.5.3 node fails, the neighboring node will react quickly to the failure (even if the neighboring node is running AIX v.5.2). If a node running AIX v.5.2 fails, the neighboring node will wait for the missing heartbeats before declaring its neighbor as failed.

Fast node failure detection is *not* supported in a mixed-version HACMP cluster.

For information about how to enable the fast method of failure detection in HACMP, see the section Changing the Failure Detection Rate of a Network Module, in the chapter on Managing the Cluster Topology in the *Administration Guide*.

Eliminating False Takeovers

If HACMP can*not* get enough CPU resources to send heartbeats on IP and point-to-point networks, other nodes in the cluster assume the node has failed and initiate takeover of the node resources. Once this takeover process begins, it is critical that the *failed* node does *not* become active and begin using these resources again.

To ensure a clean takeover, HACMP provides a Deadman Switch, which is configured to halt the unresponsive node one second before the other nodes begin processing a node failure event. The Deadman Switch uses the Failure Detection Parameters of the *slowest* network to determine at what point to halt the node. Thus, by increasing the amount of time before a failure is detected, you give a node more time in which to give HACMP CPU cycles. This can be critical if the node experiences saturation at times.

To help eliminate node saturation, modify AIX tuning parameters. For information about these tuning parameters, see the following sections:

- Configuring Cluster Performance Tuning in Chapter 1: Troubleshooting HACMP Clusters in the *Troubleshooting Guide*.
- Changing the Failure Detection Rate of a Network Module in Chapter 13: Managing the Cluster Topology in the *Administration Guide*.

Change Failure Detection Parameters only after these other measures have been implemented.

Changing a Failure Detection Rate

If you change the failure detection rate of a network module, keep in mind the following considerations:

- Failure detection is dependent on the *fastest* network linking two nodes.
- The failure rate of networks varies, depending on their characteristics.
- Before altering the network module, assess how much time you want to elapse before a real node failure is detected by the other nodes and the subsequent takeover is initiated.
- Faster heartbeat rates may lead to false failure detections, particularly on busy networks. For example, bursts of high network traffic may delay heartbeats and this may result in nodes being falsely ejected from the cluster. Faster heartbeat rates also place a greater load on networks. If your networks are very busy and you experience false failure detections, you can try slowing the failure detection speed on the network modules to avoid this problem.
- Before customizing the Failure Detection Rate, change the rate from normal to slow (or fast). To customize it, change the *hbrate* and adjust the *cycle* values to custom values using the SMIT panel for changing tuning parameters to custom values.
- The Failure Detection Rate for a particular network should be set equally for a given network across the cluster. The change must be synchronized across cluster nodes. The new values become active during a dynamic reconfiguration (DARE).
- Whenever a change is made to any of the values that affect the failure detection time (failure cycle (FC), heartbeat rate (HB) or failure detection rate), the new value of these parameters is sent as output to the screen in the following message:
- SUCCESS: Adapter Failure Detection time is now *FC* * *HB* * 2 or *SS* seconds

Setting Values for the Network Grace Period

During IP Address Takeover (IPAT) operations, the *network grace period* is the time period in which node reachability is *not* computed for the network. This is used so that a node is *not* considered to be down while its NICs are undergoing IPAT. The grace period value needs to account for the time it takes for the interface to be reconfigured with the new address, and the time it takes for it to rejoin its interface membership group. When IPAT is used with HWAT, it usually takes longer for the operation to complete, so larger values of the grace period may have to be used.

For more information about configuring these parameters, see the section Changing the Tuning Parameters to Predefined Values in Chapter 13: Managing the Cluster Topology in the *Administration Guide*.

Identifying Service Adapter Failure for Two-Node Clusters

In cluster configurations where there are networks that under certain conditions can become single adapter networks, it can be difficult for HACMP to accurately determine adapter failure. This is because RSCT Topology Services can*not* force packet traffic over the single adapter to confirm its proper operation. (This shortcoming is less of an exposure if the network adapter is under heavy use. In this case, the inbound packet count continues to increase over the service adapter without stimulation from RSCT Topology Services).

An enhancement to **netmon**, the network monitor portion of RSCT Topology Services allows for a more accurate determination of a service adapter failure. This function can be used in configurations that require a single service adapter per network.

You can create **netmon.cf** configuration file that specifies additional network addresses to which ICMP ECHO requests can be sent.

This file must exist at cluster startup—RSCT Topology Services scans the **netmon.cf** configuration file during initialization. When **netmon** needs to stimulate the network to ensure adapter function, it sends ICMP ECHO requests to each IP address. After sending the request to every address, **netmon** checks the inbound packet count before determining whether an adapter has failed.

Creating the netmon.cf File

The **netmon.cf** file must be placed in the **/usr/sbin/cluster** directory on all cluster nodes.

Requirements for creating the file:

- The **netmon.cf** file consists of one IP address or IP label per cable.
- Include each IP address and its corresponding label for the **netmon.cf** file in the **/etc/hosts** file.
- When selecting IP addresses (or hostnames) for the **netmon.cf** file, keep in mind ALL possible IP addresses that an interface might hold at any given time (boot IP addresses, service IP addresses used by HACMP, and other interfaces), and ensure that for each interface on the node, the **netmon.cf** file contains one or more targets that can be reached by those addresses.

  For example, an adapter on a node that serves as a boot adapter when it is *not* holding a service address should be able to ping some targets in the **netmon.cf** file using that boot address.

Note: Ensure that the names in the **netmon.cf** file are included in the **/etc/hosts** file. If this is *not* done, then when the NIM process (that is part of the RSCT Topology Services subsystem) attempts to determine the state of the local adapters, it may try to run the hostname resolution. The hostname resolution may in turn result in the process being blocked in cases when the network used for name resolution is unreachable. The blockage may result in longer adapter failure detection times, which will slow fallover operations.

- A maximum of 30 IP addresses/labels can be defined in **netmon.cf**.

The following example shows a **/usr/sbin/cluster/netmon.cf** configuration file:

```
180.146.181.119
steamer
chowder
180.146.181.121
mussel
```

Choosing IP Addresses for the netmon.cf File

Guidelines for choosing the IP addresses to include in the **netmon.cf** file depend on whether the local interface is a service or a non-service interface:

If the local interface is a service interface, it will verify that it is operational via point-to-point communication with the following interfaces:

- Existing remote service interface(s) on the same logical network
- One of the existing local non-service interfaces on the same logical network
- IP addresses/hostnames in the **netmon.cf** file.

If the local interface is a non-service interface, it verifies that it is operational via point-to-point communication with the following interfaces:

- Existing remote non-service(s) on the same logical network
- One of the existing local interfaces on the same logical network
- IP addresses/hostnames in the **netmon.cf** file.

The **netmon.cf** file should contain remote IP labels/addresses that are *not* in the cluster configuration that can be accessed from HACMP interfaces. Routers can also be used.

To ensure that the RSCT netmon is able to ping the address specified in **netmon.cf**, the following ping command issued from the command line must be answered:

```
# ping -S <Boot IP address> <IP addr in netmon.cf>
```

where <*Boot IP address*> is the IP address configured on the interface with **smit chinet**.

Setting RS232 TTY Baud Rates

The default baud rate is 38400 bps; however some modems or devices do *not* support a baud rate of 38400 bps. If this is the case for your situation, you can change the default by customizing the RS232 network module to read the desired baud rate (9600bps, 19200 bps, 38400 bps).

For more information about configuring these parameters, see the section Changing an RS232 Network Module Baud Rate in Chapter 13: Managing the Cluster Topology in the *Administration Guide*.

Planning Networks for Inter-Node Communication with Oracle

ORACLE uses the **private** network attribute setting to select networks for Oracle inter-node communications. This attribute is *not* used by HACMP and will *not* affect HACMP in any way. The default attribute is **public**.

Changing the network attribute to **private** makes the network Oracle-compatible by changing all interfaces to service (as well as changing the attribute in HACMPnetwork).

After creating your cluster networks (either manually or using discovery), you can change the network attribute by following this SMIT path:

Extended Configuration > Extended Topology Configuration > Configure HACMP Networks > Change/Show a Network in the HACMP Cluster.

Select the network to be changed, then change the Network Attribute setting to **private**. Synchronize the cluster after making this change. See the *Administration Guide* for complete instructions.

Rules for Configuring Private Networks

Follow these steps to configure private networks for use by Oracle:

1. Configure the network and add all interfaces. You can*not* change the attribute if the network has no interfaces.
2. Change the network attribute to **private**.
3. Private networks must have *either* all boot *or* all service interfaces. If the network has all boot interfaces (the default when using discovery) HACMP converts these interfaces to service (Oracle only looks at service interfaces).
4. Synchronize the cluster after changing the attribute.

Note: Once you define the network attribute as **private** you can*not* change it back to **public**. You have to delete the network and then redefine it to HACMP (it defaults to **public**).

Planning for SP Networks

An SP network requires special planning for HACMP networking. There are three SP-specific networks:

- *SP Ethernet*. Each SP frame includes an Ethernet network that connects all SP nodes to the Control Workstation. This network is used for software installation and other administrative purposes. In addition, it is available to HACMP and recommended for use as a heartbeat network.
- *SP Switch*. An SP frame can optionally contain a switch network that connects all SP nodes, providing a high-speed data connection. The SP Switch supports the TCP/IP protocol and is available to HACMP and client traffic.

Note: When using an SP Switch network, *configure an additional network* for HACMP. Otherwise, errors occur when synchronizing the cluster.

For IP Address Takeover via IP Aliases, (the default) the SP Switch `css0` base address should be configured as an HACMP base address. The Switch base address can*not* be modified. For more information about the SP Switch network, see the section Planning for IPAT with the SP Switch Network in this chapter.

- *SP serial network*. Every SP frame includes a serial network that connects all SP nodes to the Control Workstation. This network is used for administrative purposes and is *not* available for use by HACMP or external client traffic.

SP Planning Considerations

When planning to use multiple networks in your SP cluster, keep in mind the following:

- The SP Switch is a highly available network. Fault isolation and recovery is automatic.
- Do *not* route client traffic over the SP Ethernet. This network is used by the SP for administrative traffic, such as netinstalls, which could be affected by heavy client traffic.
- Applications that make use of the SP Switch network must be “tolerant” of brief network outages during switch initialization or switch faults. In general, most TCP/IP applications are *not* affected by switch initialization. There is no impact to HACMP other than it noticing this behavior.

Handling SP Network Failure

Rare failures may result in an SP Switch outage on one or more nodes in the cluster. HACMP distinguishes between two types of network failure:

- *Local.* A local network failure occurs when a node can no longer communicate over a particular network, but the network is still in use by other nodes. For example, this may happen if the cable between one node and the SP switch breaks.
- *Global.* A global network failure occurs when all nodes lose the ability to communicate over a network (for example, if the SP Switch itself were powered off). In either case, HACMP uses selective fallover to handle the situation.

For more information, see the section Selective Fallover Caused by Local Network Failures in Appendix B: Resource Group Behavior During Cluster Events in the *Administration Guide.*

Special SP Switch Failure Considerations

In the event that the SP Switch itself is powered off, the error notification subsystem interprets the failure as an `HPS_FAULT9_ERR` event followed by an `HPS_FAULT6_ERR` event (fault service daemon terminated).

Note that on some versions of the SP Switch, to recover from a major switch failure (power off, for example), you issue **Eclock** and **Estart** commands to bring the switch back online. The **Eclock** command runs **rc.switch,** which deletes the aliases HACMP needs for switch IP address takeover.

If you are configuring IPAT via IP Replacement on the SP Switch, create an event script for either the **network_up** or the **network_up_complete** event to add back the aliases for `css0` **.**

Planning for the SP Switch Network

The following sections discuss considerations specific to the SP Switch network, especially regarding IP Address Takeover configuration. Note that the planning depends on whether you are using the SP Switch 1 or the SP Switch 2. The considerations listed in this section apply to both types of switches unless otherwise noted.

Handling SP Switch Adapter Failure

You can*not* configure a non-service interface for the SP Switch Network if you only have css0. To eliminate this as a cluster single point of failure, use the AIX Error Notification facility to promote an SP Switch adapter failure to a node failure.

Planning for IPAT with the SP Switch Network

The SP Switch can make use of IP address aliases with HACMP to permit IP address takeover (IPAT).

You can configure IPAT on the SP Switch Network two ways. The following definitions are used in this manual to prevent confusion about which method of IPAT is discussed.

- *IPAT via IP Aliases on the SP Switch* is defined the same way as it is defined for other aliased networks in HACMP. The css0 base IP label of the SP Switch interface must be used as an HACMP non-service IP label in an IPAT via IP Aliases configuration.
- *IPAT via IP Replacement on the SP Switch* is supported without any changes to the interface definitions. The css0 base IP label of the SP Switch interface must *not* be used when you configure a resource group with IPAT via IP Replacement on the SP Switch.

The following figure is a general illustration of IPAT via IP Replacement on the SP Switch. In the figure, node 1 and node 2 are SP nodes. E1 and E2 are administrative ethernet IP addresses. The base1 and base2 labels reflect SP Switch base IP addresses. H1 and H2 are SP Switch alias HACMP service addresses. VG1 and VG2 are volume groups.

Figure 7. Sample Two-Node Cluster Configuration on an SP with Switch 1

Note: The HACMP base addresses are *not* included in the preceding figure. The HACMP base addresses are also aliases; they are *not* the same as SP switch base IP addresses.

Planning for IPAT via IP Aliases with the SP Switch

If you are planning to configure an SP Switch network manually, and want to use IPAT via IP Aliases, note the following considerations:

- Enable the Address Resolution Protocol (ARP) for the SP Switch network to use IPAT via IP Aliases on an SP Switch network.
- Configure the base adapter address on the SP Switch network as a non-service IP label in HACMP.
- Configure the service IP label on a *different* subnet than the non-service IP label to avoid errors during verification. If you have multiple service IP labels, they should all be on a different subnet than the non-service IP labels.

- Ensure that HWAT is disabled for all adapters on this network.
- Enable IP Address takeover via IP Aliases.
- The SP Switch alias addresses configured for IPAT via IP Aliases can be included in non-concurrent resource groups.

For more information about IPAT via IP Aliases, see the section Converting an SP Switch Network to an Aliased Network in Chapter 13: Managing the Cluster Topology in the *Administration Guide*.

Verifying Address Resolution Protocol (ARP) Settings for SP Switch

From HACMP 4.5 on, ARP is enabled for SP Switch networks by default. Once you have installed HACMP, ensure that the network is configured to support gratuitous ARP in HACMP. This procedure is explained in the section on SP Switch Address Resolution Protocol (ARP) in the chapter on Configuring AIX for HACMP in the *Installation Guide*.

Configuring the SP Switch2

SP Switch2 supports configurations with two switches, so that nodes can communicate over one or two switch planes. This provides improved communication performance and higher availability.

If you have two `css` interfaces on a node, they must be configured into two separate logical networks. You can*not* use one `css` interface as a service interface and another interface as a non-service on the same network, and fall over between them.

Note: PSSP allows you to configure an Aggregate IP. This is an `ml0` interface with its own IP address. This `ml0` interface is *not* supported by HACMP.

Avoiding SP Switch 2 Configuration Errors

If you have two switch adapters, but definitions for only one switch network, then the definition will be taken as applying to `css0`. A Warning Message reminds the user during the adapter definition process. If you intend to use css0, then no action is required. Otherwise, follow the process described in the preceding example. If you do *not* follow these guidelines, the interface name is css0 by default.

If you define multiple adapters without adding the subnet address of the non-service/service then synchronization will fail with the following message:

```
ERROR: Multiple switch adapters defined on %s without base subnets. Add
base subnets to switch networks.
```

Completing the Network Worksheets

In the following sections you fill out the following network planning worksheets:

- TCP/IP Networks Worksheet
- TCP/IP Network Interface Worksheet
- Point-to-point Network Worksheet
- Serial Interface Worksheet

Completing the TCP/IP Networks Worksheet

The TCP/IP Networks Worksheet helps you organize the networks for an HACMP cluster. Print the TCP/IP Networks Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Print as many copies as you need.

To complete the TCP/IP Networks Worksheet:

1. Enter a symbolic name for each network in the **Network Name** field.

   You use this value during the installation process when you configure the cluster. This name can be up to 32 characters long and can include alphanumeric characters and underscores. Do *not* begin the name with a number. Do *not* use the HACMP network type alone as a name; however, you can use the type plus a number. For example, use *Ether1* instead of *Ether*.

   The **Network Name** is a symbolic value that identifies a network in an HACMP environment. Cluster processes use this information to determine which adapters are connected to the same physical network. Use any naming convention you prefer, but be consistent. If several interfaces share the same physical network, make sure that you use the same network name when defining these adapters.

2. Identify the network type in the **Network Type** field (for example, Ethernet, Token Ring, and so forth).

   Note: For SP Switch networks, ARP must be enabled, and the network type must be **hps**.

3. In the **Netmask** field, provide the network mask of each network. The network mask is site dependent.

   The HACMP software uses the subnet facility of TCP/IP to divide a single physical network into separate logical subnets. To use subnets, define a network mask for your system.

   An IP address consists of 32 bits. Some of the bits form the network address; the remainder form the host address. The *network mask* (or netmask) determines which bits in the IP address refer to the network and which bits refer to the host. For example:

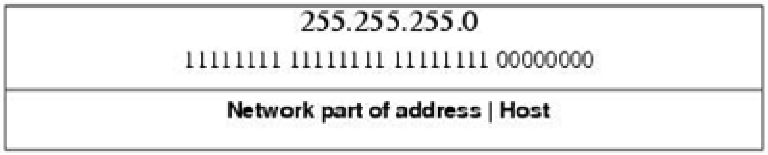


Figure 8. IP Address Example

In the preceding figure, the netmask is shown in both dotted decimal and binary format. A binary 1 indicates that a bit is part of the network address. A binary 0 indicates that a bit is part of the host address. In the IP Address Example figure, the network portion of the address occupies 24 bits; the host portion occupies 8 bits. Thus, addresses 10.10.10.1 and 10.10.10.2 would be on the same subnet; 10.10.20.1 would be on a different subnet, as it has a different *network* part in its address. It is convenient (but *not* necessary) to define a subnet on an octet boundary.

Subnetting is relevant only for the local site. Remote sites view the network portion of the IP address by the network’s class.

The HACMP software supplies a SMIT panel where you can add and remove subnets from the network using UNIX standard subnet syntax (for example, 192.9.200.0/24).

For more information, see the section Managing Communication Interfaces in HACMP in Chapter 13: Managing the Cluster Topology in the *Administration Guide*.

See the *IBM AIX Communication Concepts and Procedures* manual for further information about address classes. Also, ask your network administrator about the class and subnets used at your site.

4. In the **Node Names** field, list the names of the nodes connected to each network. Refer to your cluster diagram.
5. In the **IP Address Takeover via IP Aliases** field, the default is **YES**. Disable this option if you are *not* planning to use this feature.

   For more information, see the section IP Address Takeover via IP Aliases in this chapter.
6. In the **IP Address Offset for Heartbeating over IP Aliases** field, enter the beginning address for HACMP to use to create heartbeat alias addresses.

   For more information, see the section Heartbeating over IP Aliases in this chapter.

Completing the TCP/IP Network Interface Worksheet

The TCP/IP Network Interface Worksheet helps you define the NICs connected to each node in the cluster. Print the TCP/IP Network Interface Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section. Print as copy for each node.

To complete the TCP/IP Network Interface Worksheet:

1. Enter the node name in the **Node Name** field. You assigned this name in Chapter 2: Initial Cluster Planning. For each node, perform the following tasks for each network adapter configured on the node.
2. Assign each interface an IP label and record it in the **IP Label/Address** field. This name can be up to 32 characters long and can include alphanumeric characters (but *not* a leading numeric), hyphens, and underscores.

   If the system hostname *is the same as* the interface label, *do not use underscores* since underscores in hostnames may *not* be allowed with some levels of BIND.

   For more information about IP labels, see the section IP Labels in TCP/IP Networks in this chapter.
3. Enter the name of the network to which this interface is connected in the **Network Interface** field.
4. Identify the interface function as service, non-service, or persistent in the **Interface Function** field.
5. Record the IP address for each interface in the **IP Address** field.

   All service IP addresses for a network can share the same subnet. If your network uses heartbeat via IP aliases, your service and non-service IP addresses can all be on the same subnet, or on separate subnets.

   If your network is *not* using heartbeat via IP aliases:

- On each node, every *non-service* IP address for a given network must be on a separate subnet. You can use the same set of subnets on every node.
- If you have a resource group with the startup policy Online on Highest Available Node or Online Using Node Distribution Policy, fallover policy Fallover to the Next Priority Node in the List and fallback policy Never Fallback, the service IP address should be on the same subnet as one of the non-service IP addresses.
- If you have a resource group with the startup policy Online on Home Node, fallover policy Fallover to the Next Priority Node, and fallback policy Fallback to Highest Priority Node, the service IP address must be on a different subnet from all non-service IP addresses.

6. Enter the NIC Hardware Address.

   If you are using hardware address swapping, enter in the **NIC HW Address** field the alternate hardware address for each service IP label. The hardware address is a 12 or 14-digit hexadecimal value. Usually, hexadecimal numbers are prefaced with “0x” (zero x) for readability. For readability, colons are usually used to separate the numbers in the hardware address. The colons are *not* part of the actual address, and should *not* be entered on the SMIT panel during configuration.

   For more information, see the section Defining Hardware Addresses in this chapter.

   Note: Hardware address swapping is *not* supported on ATM networks, SP Ethernet, and SP Switch.

Completing the Point-to-Point Networks Worksheet

The Point-to-Point Networks Worksheet helps you organize the point-to-point networks for your cluster. Print the Point-to-Point Networks Worksheet from Appendix A: Planning Worksheets, and fill it out using the information in this section.

To complete the Point-to-Point Networks Worksheet:

1. Enter the cluster name in the **Cluster Name** field.
2. In the **Network Name** field, assign each network a symbolic name. Names can contain up to 32 characters alphanumeric characters and underscores. Do *not* begin the name with a number.

   The **Network Name** is a symbolic value that identifies a network in an HACMP environment. Cluster processes use this information to determine which interfaces are connected to the same physical network. Use any naming convention you prefer, but be consistent. If two interfaces share the same physical network, make sure that you use the same network name when defining these interfaces.
3. Identify the network type in the **Network Type** field. For point-to-point networks, you can specify RS232, disk heartbeating (diskhb), Target Mode SCSI (tmscsi), or Target Mode SSA (tmssa).

   The **Network Attribute** field has a preassigned value.
4. In the **Node Names** field, list the names of the nodes connected to each network. Refer to your cluster diagram.
5. (*For diskhb*) In the **Hdisk** field, specify the disk used in the diskhb network.

6. Use the **Miscellaneous Data** field to record any extra information about devices used to extend point-to-point links (for example, modem number or extender information).

Completing the Serial Network Interface Worksheet

The Serial Network Interface Worksheet in Appendix A: Planning Worksheets lets you define the network interfaces connected to each node in the cluster. Complete the following steps for each node on a separate worksheet

To complete the Serial Network Interface Worksheet:

1. Enter the node name in the **Node Name** field. You assigned this name when filling in the section Completing the TCP/IP Networks Worksheet in this chapter. Record the following information for each serial network configured on the node.
2. Enter the number of the slot in which the serial interface is located in the **Slot Number** field.

   You will enter a value in the **Interface Name** field after you configure the adapter following the instructions in the relevant AIX documentation. AIX assigns an interface name to the interface when it is configured. The interface name is made up of two or three characters that indicate the type of interface, followed by a number that AIX assigns in sequence for each interface of a certain type. For example, AIX assigns an interface name such as tty0 for the first tty.
3. Assign each interface a symbolic name and record it in the **Interface Label** field.

   Use a naming convention that helps identify the interface's role in the cluster. For example, such as *nodea-tty1*.
4. In the **Network Name** field, enter the name you assigned to the network in the Point-to-Point Network Worksheet.
5. The appropriate values appear in the **Interface Function** field.

Defining Hardware Addresses

Note: You can*not* use hardware address swapping if you have IP address takeover via IP aliases configured for the network. You can*not* use Hardware Address Takeover on the SP Ethernet or SP Switch networks. Hardware address swapping is *not* supported on ATM networks.

The hardware address swapping facility works in tandem with IP address takeover via IP Replacement. Hardware address swapping maintains the binding between an IP address and a hardware address. This eliminates the need to update the ARP cache of clients and network devices after an IP address takeover. This facility is supported for Ethernet, Token Ring, and FDDI adapters.

Note that hardware address swapping takes about 60 seconds on a Token Ring network and up to 120 seconds on a FDDI network. These periods are longer than the usual time it takes for the Cluster Manager to detect a failure and take action, so you may need to adjust the tuning parameters for these networks. For more information, see the section Setting Failure Detection Parameters in this chapter.

Selecting an Alternate Hardware Address

This section provides information about hardware addressing for Ethernet, Token Ring, and FDDI network interface cards. Note that any alternate hardware address you define for a NIC should be similar in form to the default hardware address the manufacturer assigned to the NIC.

To determine an adapter's default hardware address, use the **netstat -i** command (when the networks are active).

Using netstat

To retrieve hardware addresses using the **netstat -i** command, enter:

```
netstat -i | grep link
```

The command returns output similar to the following:

```
lo0   16896 link#1                          186303     0   186309     0     0
en0   1500  link#2      2.60.8c.2f.bb.93      2925     0     1047     0     0
tr0   1492  link#3      10.0.5a.a8.b5.7b    104544     0    92158     0     0
tr1   1492  link#4      10.0.5a.a8.8d.79     79517     0    39130     0     0
fi0   4352  link#5      10.0.5a.b8.89.4f     40221     0        1     1     0
fi1   4352  link#6      10.0.5a.b8.8b.f4     40338     0        6     1     0
```

Using the arp Command

Use the **arp** command to view the list of nodes, IP addresses, and associated hardware (MAC) addresses in a host's ARP cache. For example:

```
arp -a

flounder (100.50.81.133) at 8:0:4c:0:12:34 [ethernet]
cod (100.50.81.195) at 8:0:5a:7a:2c:85 [ethernet]
seahorse (100.50.161.6) at 42:c:2:4:0:0 [token ring]
pollock (100.50.81.147) at 10:0:5a:5c:36:b9 [ethernet]
```

This output shows what the host node currently believes to be the IP and MAC addresses for nodes flounder, cod, seahorse and pollock. (If IP address takeover occurs without Hardware Address Takeover (HWAT), the MAC address associated with the IP address in the host's ARP cache may become outdated. You can correct this situation by refreshing the host's ARP cache.)

For more information, see the **arp** man page.

Specifying an Alternate Ethernet Hardware Address

To specify an alternate hardware address for an Ethernet interface, begin by using the first five pairs of alphanumeric characters as they appear in the current hardware address. Then substitute a different value for the last pair of characters. Use characters that do *not* occur on any other adapter on the physical network.

For example, you could use 10 and 20 for node A and node B, respectively. If you have multiple adapters for hardware address swapping in each node, you can extend to 11 and 12 on node A, and 21 and 22 on node B.

Specifying an alternate hardware address for adapter interface en0 in the output above thus yields the following value:

Original address	`02608c2fbb93`
New address	`02608c2fbb10`

To define this alternate hardware address to the cluster environment, see the section Configuring Service IP Labels/Addresses in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

Specifying an Alternate Token Ring Hardware Address

To specify an alternate hardware address for a Token Ring interface, set the first two digits to **42**, indicating that the address is set locally.

Specifying an alternate hardware address for adapter interface tr0 in the output above thus yields the following value:

Original address	`10005aa8b57b`
New address	`42005aa8b57b`

To define this alternate hardware address to the cluster environment, see the section Configuring Service IP Labels/Addresses in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

Specifying an Alternate FDDI Hardware Address (for Service Labels)

To specify an alternate FDDI hardware address, enter the new address into the **Adapter Hardware Address** field as follows, *without any decimal separators*:

1. Use 4, 5, 6, or 7 as the first digit (the first nibble of the first byte) of the new address.
2. Use the last 6 octets of the manufacturer’s default address as the last 6 digits of the new address.

Here is a list of some sample valid addresses, shown with decimals for readability:

```
40.00.00.b8.10.89
40.00.01.b8.10.89
50.00.00.b8.10.89
60.00.00.b8.10.89
7f.ff.ff.b8.10.89
```

Avoiding Network Conflicts

Avoid network conflicts on the following:

- Hardware Address. Each network adapter is assigned a unique hardware address when it is manufactured, which ensures that no two adapters have the same network address. When defining a hardware address for a network adapter, ensure that the defined address does *not* conflict with the hardware address of another network adapter in the network. Two network adapters with the same hardware address can cause unpredictable behavior within the network.

- IP Addresses. Verification will notify you if there are duplicate IP addresses. Correct the duplicate address and resynchronize the cluster.

Adding the Network Topology to the Cluster Diagram

You can now add the network topology to your cluster diagram.

Sketch the networks to include all TCP/IP networks (including any TCP/IP point-to-point connections) and non-IP point-to-point networks. Identify each network by name and attribute. In the boxes in each node that represent slots, record the interface label.

You can now add networking to the sample cluster diagram started in Chapter 1: Overview of the Planning Process.

A sample network set up may include the use of five networks:

- A Token Ring network, named *clus1_TR*, used to connect clients to the ten cluster nodes that run the customer database “front end” application. The Token Ring network allows client access.
- An Ethernet network, named *db_net*, used to connect the four cluster nodes that run the database application, which handles the update of the actual database records. The Ethernet network *db_net* is *not* intended for client use.
- The SP Ethernet is configured as a service network, named *sp_ether*. It is *not* available for client access.
- The SP Switch network is configured as a service network and is also used for high speed data transfer between the front end nodes and back end nodes. For example, you may name the SP Switch network *clus1_HPS*.
- To avoid corruption of critical database storage, the *db_net* nodes are also connected by a series of point-to-point networks. These networks provide additional protection against contention situations where IP heart beat packets are discarded.

The RSCT software monitors the status of all defined interfaces by sending heartbeats across the network.

Where You Go from Here

You have now planned the network topology for the cluster. The next step in the planning process is to lay out the shared disk configuration for your cluster, described in Chapter 4: Planning Shared Disk and Tape Devices.

Chapter 4: Planning Shared Disk and Tape Devices

This chapter discusses information to consider before configuring shared external disks in an HACMP cluster and provides information about planning and configuring tape drives as cluster resources. This chapter contains the following sections:

- Prerequisites
- Overview
- Choosing a Shared Disk Technology
- Disk Power Supply Considerations
- Planning for Non-Shared Disk Storage
- Planning a Shared SCSI Disk Installation
- Planning a Shared IBM SSA Disk Subsystem Installation
- Completing the Disk Worksheets
- Adding the Disk Configuration to the Cluster Diagram
- Planning for Tape Drives as Cluster Resources
- Where You Go from Here

Prerequisites

By now, you should have completed the planning steps in the previous chapters:

- Chapter 2: Initial Cluster Planning
- Chapter 3: Planning Cluster Network Connectivity

Refer to AIX documentation for the general hardware and software setup for your disk and tape devices.

Overview

In an HACMP cluster, shared disks are external disks connected to more than one cluster node. In a non-concurrent configuration, only one node at a time *owns* the disks. If the owner node fails, the cluster node with the next highest priority in the resource group nodelist acquires ownership of the shared disks and restarts applications to restore critical services to clients. This ensures that the data stored on the disks remains accessible to client applications.

Typically, takeover occurs within 30 to 300 seconds. This range depends on the number and types of disks used, the number of volume groups, the file systems (whether shared or NFS cross-mounted), and the number of critical applications in the cluster configuration.

When planning the shared external disk for your cluster, the objective is to eliminate single points of failure in the disk storage subsystem. The following table lists the disk storage subsystem components, with recommended ways to eliminate them as single points of failure:

Cluster Object	Eliminated as Single Point of Failure by...
Disk adapter	Using redundant disk adapters
Controller	Using redundant disk controllers
Disk	Using redundant hardware and LVM disk mirroring or RAID mirroring

In this chapter, you perform the following planning tasks:

- Choosing a shared disk technology.
- Planning the installation of the shared disk storage. This includes:
  - Determining the number of disks required to handle the projected storage capacity. You need multiple physical disks on which to put the mirrored logical volumes. Putting copies of a mirrored logical volume on the same physical device defeats the purpose of making copies. For more information about creating mirrored logical volumes, see Chapter 5: Planning Shared LVM Components.
  - Determining the number of disk adapters each node will contain to connect to the disks or disk subsystem.

    Physical disks containing logical volume copies should be on separate adapters. If all logical volume copies are connected to a single adapter, the adapter is potentially a single point of failure. If the single adapter fails, HACMP moves the volume group to an alternate node. Separate adapters prevent the need for this move.
  - Understand the cabling requirements for each type of disk technology.
- Completing planning worksheets for the disk storage.
- Adding the selected disk configuration to the cluster diagram.
- Planning for configuring a SCSI streaming tape drive or a direct Fibre Channel Tape unit attachment as a cluster resource.

Choosing a Shared Disk Technology

For a complete list of supported hardware, including disks and disk adapters, as of the date of publication of this guide, see the following URL:

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

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

The HACMP software supports the following disk technologies as shared external disks in a highly available cluster:

- SCSI drives, including RAID subsystems
- IBM SSA adapters and SSA disk subsystems
- Fibre Channel adapters and disk subsystems

- Data path devices (VPATH)—SDD 1.3.1.3 or greater.

You can combine these technologies within a cluster. Before choosing a disk technology, review the considerations for configuring each technology as described in this section.

HACMP also supports dynamic tracking of Fibre Channel devices. For information about adapters and devices, see the *Adapters, Devices, and Cable Information for Multiple Bus Systems* guide at the following URL:

http://publibfp.boulder.ibm.com/epubs/pdf/38051616.pdf

For information about installing and configuring OEM disks, see the appendix on OEM Disk, Volume Group and File Systems Accommodation in the *Installation Guide*.

Obtaining HACMP APARS

An authorized program analysis report (APAR) contains an account of a problem caused by a suspected defect in a current, unaltered release of a program. You can obtain a list of HACMP APARs and updates for hardware as follows:

1. Go to IBM support website http://www.ibm.com/support/us/
2. Search on “HACMP +APAR”
3. Sort the results by date, newest first

Disk Planning Considerations

This chapter includes information to use the following SCSI disk devices and arrays as shared external disk storage in cluster configurations:

- SCSI Disk Devices
- IBM 2104 Expandable Storage Plus
- IBM 2105 Enterprise Storage Server
- IIBM System p™
- IBM TotalStorage DS6000 Storage Devices
- IBM TotalStorage DS8000 Storage Devices

SCSI Disk Devices

In an HACMP cluster, shared SCSI disks are connected to the same SCSI bus for the nodes that share the devices. They may be used in both concurrent and non-concurrent modes of access. In a non-concurrent access environment, the disks are *owned* by only one node at a time. If the owner node fails, the cluster node with the next highest priority in the resource group nodelist acquires ownership of the shared disks as part of fallover processing. This ensures that the data stored on the disks remains accessible to client applications.

The following restrictions apply to using shared SCSI disks in a cluster configuration:

- Different types of SCSI busses can be configured in an HACMP cluster. Specifically, SCSI devices can be configured in clusters of up to four nodes, where all nodes are connected to the same SCSI bus attaching the separate device types.

- You can connect up to sixteen devices to a SCSI bus. Each SCSI adapter, and each disk, is considered a separate device with its own SCSI ID. The maximum bus length for most SCSI devices provides enough length for most cluster configurations to accommodate the full sixteen-device connections allowed by the SCSI standard.
- Do *not* connect other SCSI devices, such as CD-ROMs, to a shared SCSI bus.
- If you mirror your logical volumes across two or more physical disks, the disks should *not* be connected to the same power supply; otherwise, loss of a single power supply can prevent access to all copies. Plan on using multiple disk subsystem drawers or desk-side units to avoid dependence on a single power supply.
- With quorum enabled, a two-disk volume group puts you at risk for losing quorum and data access. You can use a forced varyon to help ensure data availability.

IBM 2104 Expandable Storage Plus

The IBM 2104 Expandable Storage Plus (EXP Plus) system provides flexible, scalable, and low-cost disk storage for RS/6000 and System p™ servers in a compact package. It is a good choice for small, two-node clusters.

EXP Plus:

- Scales from up to 2055 GB of capacity per drawer or tower to more than 28 TB per rack
- Supports single or split-bus configuration flexibility to one or two servers
- Incorporates high-performance Ultra3 SCSI disk storage with 160 MB/sec. throughput
- Features 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
- Requires a low voltage differential SE (LVD/SE) SCSI connection (for example a 6205 Dual Channel Ultra2 SCSI adapter)
- Can be connected to only two nodes
- Has two separate SCSI buses

  Each node has its own SCSI connection to EXP Plus. This eliminates the need to change any adapter IDs.
- Is internally terminated.

HACMP supports 2104 Expandable Storage Plus with the following adapters:

- 2494
- 6203
- 6205
- 6206
- 6208.

HACMP *has not* been tested with 2104 Expandable Storage Plus and the 2498 adapter.

For full information, see the IBM website at the following URL:

http://www.ibm.com/servers/storage/support/disk/2104/

IBM 2105 Enterprise Storage Server

The IBM 2105 Enterprise Storage Server (ESS) 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.

These systems use IBM SSA disk technology internally. A Fibre Channel or a SCSI connection can be used to access the ESS, depending on the setup of the specific ESS system. Nodes using either access mechanism can share volumes, that is one node could use Fibre Channel and another node use a SCSI connection.

On the ESS, all storage is protected with RAID technology. RAID-5 techniques can be used to distribute parity across all disks in the array. Failover Protection enables one partition, or *storage cluster,* of the ESS to takeover for the other so that data access can continue.

HACMP and the IBM 2105 Enterprise Storage Server supports the following adapters for Fibre Channel attachment:

- Gigabit Fibre Channel Adapter for PCI bus: adapter 6227 with firmware 3.22 A1
- 2 Gigabit Fibre Channel Adapter for 64-bit PCI bus: adapter 6228 with firmware 3.82 A1.

The ESS includes web-based management interface, dynamic storage allocation, and remote services support. For more information on planning, general reference material, and attachment diagrams, see the following URL:

http://www.storage.ibm.com/disk/ess/index.html

IBM TotalStorage DS4000 Storage Server

The DS4000 series (formerly named the FAStT series) has been enhanced to complement the entry and enterprise disk system offerings with the following:

- 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
- EXP100 serial ATA expansion units attached to DS4400s.

DS4000 storage servers support 1-2 connections per node over an FC adapter. Use two adapters. DS4000 storage servers support the following adapters:

- Gigabit Fibre Channel Adapter for PCI bus: adapter 6227 with firmware 3.22 A1
- 2 Gigabit Fibre Channel Adapter for 64-bit PCI bus: adapter 6228 with firmware 3.82 A1.

The IBM DS4400 (formerly FAStT700) delivers superior performance with 2 Gbps Fibre Channel technology. The DS4400 offers protection with advanced functions and flexible facilities. It scales from 36GB to over 32TB to support growing storage requirements and offers advanced replication services.

HACMP supports BladeCenter JS20 with IBM Total Storage DS4000.

The IBM TotalStorage DS4300 (formerly FAStT600) is a mid-level disk system that can scale to over eight terabytes of fibre channel disk using 3 EXP700s, over sixteen terabytes of fibre channel disk with the Turbo feature using 7 EXP700s. It uses the latest in storage networking technology to provide an end-to-end 2 Gbps Fibre Channel solution.

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

IBM TotalStorage DS6000 Storage Devices

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

HACMP/XD supports IBM TotalStorage DSCLI Metro Mirror using DS6000, along with a combination of DS8000 and DS6000 Series Disk Storage Devices.

The DS6000 series is a Fibre Channel based storage system that supports a wide range of IBM and non-IBM server platforms and operating environments. This includes open systems, zSeries, and iSeries servers.

IBM TotalStorage DS8000 Storage Devices

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

HACMP/XD v5 has extended its Metro Mirror support to IBM TotalStorage DS8000 Series Disk Storage Devices.

IBM TotalStorage DS8000 is a high-performance, high-capacity series of disk storage that is designed to support continuous operations. DS8000 series models consist of a storage unit and one or two management consoles, two being the recommended configuration. For high-availability, hardware components are redundant.

IIBM System p™

The disk systems most often supported as external disk storage in cluster configurations are:

- IBM System p™ p5 models 510, 520, 550 and 570
- IBM System p™ p5 model 575
- IBM System p™ p5 models 590 and 595
- IBM System p™ i5 Models 520, 550 and 570 iSeries and System p™ Convergence
- RS/6000 SP System

HACMP supports the following on the IBM System p™:

- 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 System p™ p5 models 510, 520, 550, 570, and 575running AIX v.5.2 and up with applicable APARs installed.

The System p™ p5 Express family uses the IBM POWER5™ microprocessor. The POWER5 processor can run both 32- and 64-bit applications simultaneously. Dynamic logical partitioning (LPAR), helps assign system resources (processors, memory and I/O) for faster, non-disruptive response to changing workload requirements. This allows automatic, non-disruptive balancing of processing power between partitions resulting in increased throughput and consistent response times.

Be aware that there are some limitations to the supported configuration.

- Virtual SCSI (VSCSI) requires the use of Enhance Concurrent mode, which involves some limitations, but may be an acceptable solution. For more information, see:

  http://www.ibm.com/support/techdocs/atsmastr.nsf/Web/TechDocs
- The p5 520, p5 550, p5 570, and p5 575 integrated serial ports are *not* enabled when the HMC ports are connected to a Hardware Management Console. Either the HMC ports or the integrated serial ports can be used, but *not* both. Moreover, the integrated serial ports are supported only for modem and async terminal connections. Any other applications using serial ports, including HACMP, require a separate serial port adapter to be installed in a PCI slot.
- Since there are no integrated serial ports on the p5-575, HACMP can use alternative methods for non-IP heartbeating through asynchronous io adapters and disk heartbeating. HACMP is able to use the four integrated ethernet ports designated for application use.
- For information on HACMP support of virtualization (VLAN, VSCSI), see:

  http://www.ibm.com/support/techdocs/atsmastr.nsf/Web/TechDocs
- HACMP support of DLPAR requires Apar IY69525

IBM System p™ p5 model 575

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

The p5 575 delivers an 8-way, 1.9 GHz POWER5 high-bandwidth cluster node, ideal for many high-performance computing applications. The p5 575 is packaged in a dense 2U form factor, with up to 12 nodes installed in a 42U-tall, 24-inch rack. Multiple racks of p5-575 nodes can be combined to provide a broad range of powerful cluster solutions. Up to 16 p5-575 nodes can be clustered together for a total of 128 processors.

The following lists limitations to the supported System p™ p5 model 575:

- There are no integrated serial ports on the p5 575. HACMP can use alternative methods for non-IP heartbeating through asynchronous I/O adapters and disk heartbeating.
- HACMP is able to use the four integrated ethernet ports designated for application use.
- HACMP does *not* support DLPAR or Virtual LAN (VLAN) on p5 575 at this time.
- HACMP support of Virtual SCSI (VSCSI) requires the use Enhanced Concurrent mode, which involves some limitations but may be an acceptable solution for some customers.

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 IBMs fastest POWER5 processors running at 1.90 GHz or 1.65 GHz, while the p5-590 offers 1.65 GHz processors.

These servers come standard with mainframe-inspired reliability, availability, serviceability (RAS) capabilities and IBM Virtualization Engine™ systems technology with Micro-Partitioning™. Micro-Partitioning allows as many as ten logical partitions (LPARs) per processor to be defined. Both systems can be configured with up to 254 virtual servers with a choice of AIX™, Linux, and i5/OS™ operating systems in a single server, opening the door to vast cost-saving consolidation opportunities.

Beginning with HACMP 5.3, support of VSCSI requires the use Enhanced Concurrent mode, which involves some limitations but may be an acceptable solution for some customers. A whitepaper describing the capabilities and use of Enhanced Concurrent mode in a VSCSI environment is available from the IBM website.

HACMP supports the following adapters for Fibre Channel attachment:

- FC 6239 or FC 5716 2 Gigabit Fibre Channel - PCI-X

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

HACMP 5.1 and up supports the IBM System p™ i5, which is a hardware platform of iSeries and System p™ convergence, on which 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 gives an excellent alternative 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.

HACMP 5.1 up supports the new POWER5-based IBM i5 520, 550, and 570 servers running either AIX v.5.2 or v.5.3 with applicable APARs installed.

HACMP supports the following adapters for Fibre Channel attachment:

- FC 6239 or FC 5716 2 Gigabit Fibre Channel - PCI-X

There are some limitations to the supported configurations of HACMP/AIX on iSeries i5 systems to consider:

- On i5 hardware, HACMP will run only in LPARs that are running supported releases of AIX. In addition, I/O (LAN and disk connections) must be directly attached to the LPAR(s) in which HACMP runs. I/O, which is intended for use with HACMP, is limited to that which is listed as supported in the HACMP Sale Manual 5765-F62.
- HACMP also supports AIX partitions on i5 servers provided the AIX partitions running HACMP have dedicated I/O.
- HACMP does *not* support micro-partitioning, Virtual SCSI (VSCSI) or Virtual LAN (VLAN) on i5 models this time.

The i5 520, 550, and 570 integrated serial ports are *not* enabled when the HMC ports are connected to Hardware Management Console. Either the HMC ports or the integrated serial ports can be used, but *not* both. Moreover, the integrated serial ports are supported only for modem and async terminal connections. Any other applications using serial ports, including HACMP, require a separate serial port adapter to be installed in a PCI slot.

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.

IBM Serial Storage Architecture Disk Subsystem

Serial Storage Architecture (SSA) enables you to minimize single points of failure and achieve high availability in an HACMP environment.

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

The AIX V5.2 MPIO facility can be used to access disk subsystems through multiple paths. Multiple paths can provide both more throughput and higher availability than use of a single path. In particular, when multiple paths are used, failure of a single path due to an adapter, cable or switch failure will not cause applications to lose access to data. While HACMP will attempt to recover from complete loss of access to a volume group, that loss itself is going to be temporarily disruptive. The AIX V5.2 MPIO facility can prevent a single component failure from causing an application outage. When a shared volume group in an HACMP cluster is accessed through MPIO, it must be defined as an enhanced concurrent volume group.

SSA is hot pluggable. Consequently, if you include SSA disks in a volume group using LVM mirroring, you can replace a failed disk drive without powering off the entire system.

The following figure shows the basic SSA loop configuration:

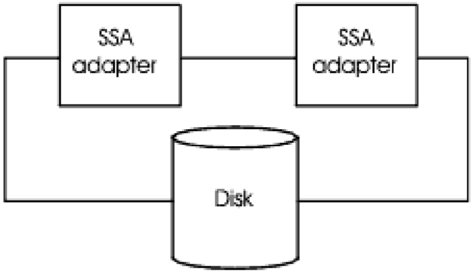


Figure 9. Basic SSA Loop Configuration

Disk Power Supply Considerations

Reliable power sources are critical for a highly available cluster. Each mirrored disk chain in the cluster should have a separate power source. As you plan the cluster, make sure that the failure of any one power source (PDU, power supply, or building circuit) does *not* disable more than one node or mirrored chain.

SCSI Device Power Considerations

If the cluster site has a multiple phase power supply, ensure that the cluster nodes are attached to the same power phase. Otherwise, the ground will move between the systems across the SCSI bus and cause write errors.

The bus and devices shared between nodes are subject to the same operational power surge restrictions as standard SCSI systems. Uninterruptible power supply (UPS) devices are necessary for preventing data loss. When power is first applied to a SCSI device, the attached bus, if actively passing data, may incur data corruption. You can avoid such errors by briefly halting data transfer operations on the bus while a device (disk or adapter) is turned on. For example, if cluster nodes are installed on two different power grids and one node has a power surge that causes it to reboot, the surviving node may lose data if a data transfer is active.

The IBM DS4000 series (formerly named the FAStT series) series, the IBM 2104 Expandable Storage Plus, and the IBM 2105 Enterprise Storage Servers are less prone to power supply problems because they have redundant power supplies.

IBM SSA Disk Subsystem Power Considerations

Clusters with IBM SSA disk subsystems are less prone to power supply problems because they have redundant power supplies.

Planning for Non-Shared Disk Storage

Keep the following considerations in mind regarding non-shared disk storage:

- Internal disks. The internal disks on each node in a cluster must provide sufficient space for:
  - AIX software (approximately 500 MB)
  - HACMP software (approximately 50 MB for a server node)
  - Executable modules of highly available applications.
- Root volume group. The root volume group for each node must *not* reside on the shared SCSI bus.
- AIX Error Notification Facility. Use the AIX Error Notification Facility to monitor the **rootvg** on each node. Problems with the root volume group can be promoted to node failures.

  For more information about using the Error Notification facility, see the chapter on Configuring AIX for HACMP in the *Installation Guide*.
- Disk adapter use. Because shared disks require their own adapters, you can*not* use the same adapter for both a shared and a non-shared disk. The internal disks on each node require one SCSI adapter apart from any other adapters within the cluster.

- Volume group use. Internal disks must be in a different volume group from the external shared disks.

The executable modules of the highly available applications should be on the internal disks and *not* on the shared external disks, for the following reasons:

- Licensing
- Application startup.

Licensing

Vendors may require that you purchase a separate copy of each application for each processor or multi-processor that may run it, and protect the application by incorporating processor-specific information into the application when it is installed.

Thus, if you are running your application executable from a shared disk, it is possible that after a fallover, HACMP will unable to restart the application on another node, because, for example, the processor ID on the new node does *not* matching the ID of the node on which the application was installed.

The application may also require that you purchase what is called a *node-bound license*, that is, a license file on each node that contains information specific to the node.

There may also be a restriction on the number of *floating* (available to any cluster node) licenses available within the cluster for that application. To avoid this problem, be sure that there are enough licenses for all processors in the cluster that may potentially run an application at the same time.

Starting Applications

Applications may contain configuration files that you can customize during installation and store with the application files. These configuration files usually store information, such as pathnames and log files, that are used when the application starts.

You may need to customize your configuration files if your configuration requires both of the following:

- You plan to store these configuration files on a shared file system.
- The application can*not* use the same configuration on every fallover node.

This is typically the case in instances where the application typically runs on more than one node, with different configurations. For example, in a two-node mutual takeover configuration, both nodes may be running different instances of the same application, and standing by for one another. Each node must be aware of the location of configuration files for both instances of the application, and must be able to access them after a fallover. Otherwise, the fallover will fail, leaving critical applications unavailable to clients.

To decrease how much you will need to customize your configuration files, place slightly different startup files for critical applications on local file systems on either node. This allows the initial application parameters to remain static; the application will *not* need to recalculate the parameters each time it is called.

Planning a Shared SCSI Disk Installation

The following sections summarize the basic hardware components required to set up an HACMP cluster that includes the following shared storage systems:

- HACMP and Virtual SCSI
- Disk Adapters
- Cables
- Sample IBM 2104 Expandable Storage Plus Configuration
- Sample IBM DS4000 Storage Server Configuration
- Sample IBM 2105 Enterprise Storage Server Configuration

Your cluster requirements depend on the configuration you specify. To ensure that you account for all required components, complete a diagram for your system. In addition, consult the hardware manuals for detailed information about cabling and attachment for the particular devices you are configuring.

HACMP and Virtual SCSI

HACMP supports VIO (virtual I/O) SCSI with the applicable APARs installed. The following restrictions apply to using Virtual SCSI (VSCSI) in a cluster configuration:

- The volume group must be defined as “Enhanced Concurrent Mode.” In general, Enhanced Concurrent Mode is the recommended mode for sharing volume groups in HACMP clusters because volumes are accessible by multiple HACMP nodes, resulting in faster failover in the event of a node failure.
- If file systems are used on the standby nodes, they are *not* mounted until the point of fallover so accidental use of data on standby nodes is impossible.
- If shared volumes are accessed directly (without file systems) in Enhanced Concurrent Mode, these volumes are accessible from multiple nodes so access must be controlled at a higher layer such as databases.
- If any cluster node accesses shared volumes through VSCSI, all nodes must do so. This means that disks can*not* be shared between an LPAR using VSCSI and a node directly accessing those disks.
- From the point of view of the VIO server, physical disks (hdisks) are shared, *not* logical volumes or volume groups.
- All volume group construction and maintenance on these shared disks is done from the HACMP nodes, *not* from the VIO server.

Disk Adapters

Remove any SCSI terminators on the adapter card. Use external terminators in an HACMP cluster. If you terminate the shared SCSI bus on the adapter, you lose termination when the cluster node that contains the adapter fails.

For a complete list of supported disk adapters, see the section see the following URL:

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

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

Cables

The cables required to connect nodes in your cluster depend on the type of SCSI bus you are configuring. Select cables that are compatible with your disk adapters and controllers. For information on the type and length SCSI cable required, see the hardware documentation that accompanies each device you want to include on the SCSI bus.

Sample IBM 2104 Expandable Storage Plus Configuration

The following figure shows a sample two-node configuration using the 2104 Expandable Storage Plus system.

Note: Configuration for SCSI connections from other storage systems would resemble this one.

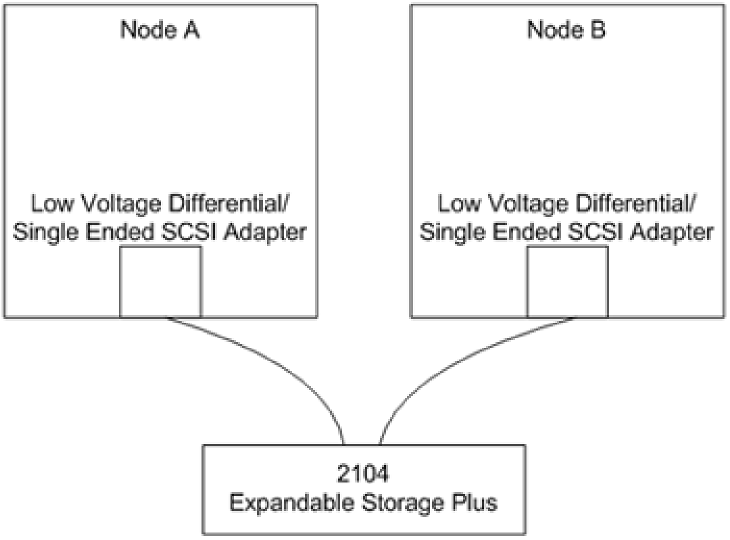

Node A
Low Voltage Differential/ Single Ended SCSI Adapter
Node B
Low Voltage Differential/ Single Ended SCSI Adapter
2104 Expandable Storage Plus

Figure 10. Two-Node Expandable Storage Plus System

Sample IBM DS4000 Storage Server Configuration

The following figure shows a configuration for high availability when using a DS4000 Storage server (formerly FAStT) in an HACMP environment:

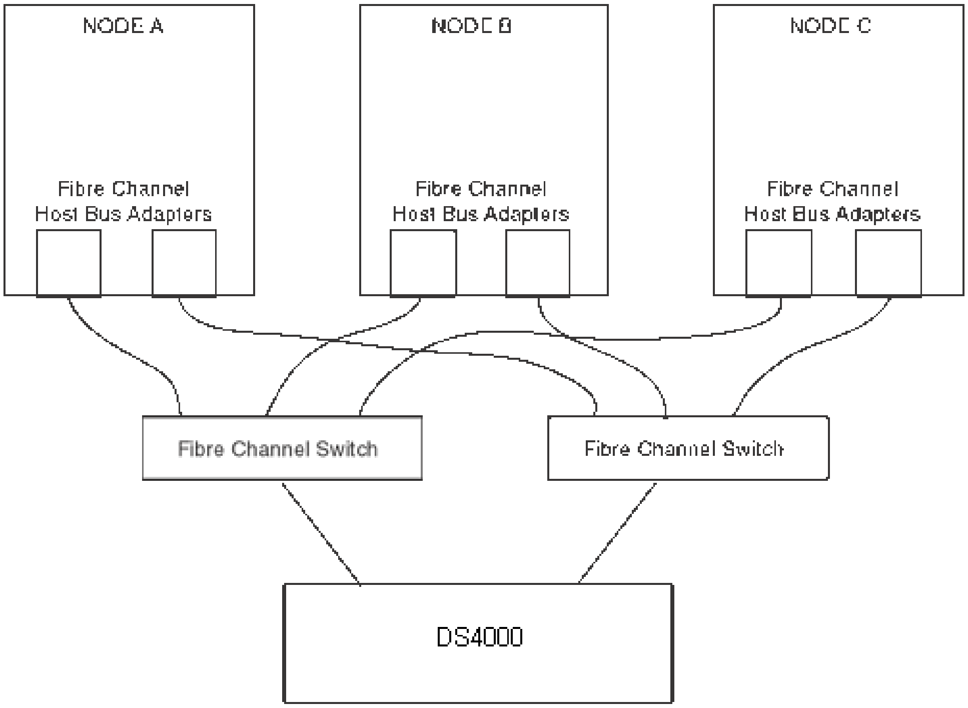

NODE A
Fibre Channel Host Bus Adapters
NODE B
Fibre Channel Host Bus Adapters
NODE C
Fibre Channel Host Bus Adapters
Fibre Channel Switch
Fibre Channel Switch
DS4000

Figure 11. DS4000 Storage Server Environment

Sample IBM 2105 Enterprise Storage Server Configuration

Information on the IBM website includes several diagrams for the IBM 2105 Enterprise Storage Server. Search for the documentation for this model from the following URL:

http://www.storage.ibm.com/solutions/index.html

Using ESS Functions for High Availability

When using the ESS in an HACMP environment, use the following:

- Use the Sparing function to assign disks as spares and reduce the exposure to data loss. When the ESS detects that a disk is failing, it transfers the data from the failing disk to a spare device. You are required to specify at least one disk as a spare per drawer; however you can specify two spares to a drawer for increased availability.
- Configure the two host interface cards in a bay to device interface cards in the same bay.
- Configure the SCSI ports on the same interface card to the same partition of the ESS.

If you are using Switched Fabric:

- Fibre Channel Switch must be configured with host World Wide Node Name (WWN)
- Zoning (similar to routing) must be configured in the switch.

Planning a Shared IBM SSA Disk Subsystem Installation

This section describes using SSA disks with HACMP. It supplements the IBM documentation that covers the specific SSA disk subsystem hardware you are using.

AIX and HACMP Levels

On nodes running AIX v.5.2, the C-SPOC utility does *not* allow *new* SSA concurrent mode volume groups to be created. (If you upgraded from previous versions of HACMP, you can use existing volume groups in SSA concurrent mode, but C-SPOC does *not* allow to create new groups of this type.) You can convert these volume groups to enhanced concurrent mode.

On nodes running AIX v.5.3, convert all volume groups to enhanced concurrent mode.

Disk Adapters

See the IBM manual for your hardware to see how to connect SSA disk subsystems to nodes.

Note: The SSA 4-port adapter (feature code 6218, type 4-J) is *not* suitable for use with HACMP because it supports only one adapter per loop.

Advanced SerialRAID Adapter (Feature Code 6225, Type 4-P)

The 6225 SSA adapter (also called an eight-way adapter) can support SSA loops containing up to 8 eight-way adapters per loop. Most multi-node configurations set up with a minimal number of single points of failure require eight-way adapters. If any drives in the loop are configured for RAID 5, only two adapters can be used in the loop.

These adapters must be at microcode level 1801 or later.

SSA Multi-Initiator RAID/EL Adapters (Feature Codes 6215 Type 6-N)

If the fast write cache or RAID functions of the adapters are used, no other adapter can be connected in an SSA loop with this adapter. If those functions are used, a second SSA Multi-Initiator RAID/EL adapter can be connected in the loop.

Identifying Disk Adapters

The two-way and eight-way disk adapters look the same, but their microcode is different. The easiest way to distinguish between these adapters is to install it in a machine and run either of the following commands:

```
lsdev -Cc adapter
```

or

```
lscfg -vl ssaX
```

where X is the adapter number.

These commands provide identifying information about the microcode.

Bypass Cards

The 7133 Models T40 and D40 disk subsystems contain four bypass cards. Each bypass card has two external SSA connectors. Through these, you connect the bypass cards and, therefore, the disk drive module strings to each other or to a node.

The bypass cards can operate in either bypass or forced inline mode.

Bypass Mode

When you set its jumpers so a bypass card operates in bypass mode, it monitors both of its external connections. If it detects that one of its connectors is connected to a powered-on SSA adapter or device, it switches to inline mode; that is, it connects the internal SSA links to the external connector. This effectively heals the break in the SSA loop.

If the bypass card detects that neither of its connectors is connected to a powered-on SSA adapter or device, it switches to bypass state; that is, it connects the internal disk strings and disconnects them from the external connector.

Forced Inline Mode

When you set its jumpers so a bypass card operates in forced inline mode, it behaves permanently like a signal card of Models 010 and 500; that is, none of its electronic switching circuits are in use. Its internal SSA links connect to the external connector and can never make an internal bypass connection.

Using SSA Facilities for High Availability

This section describes how you can use SSA facilities to make your system highly available.

SSA Loops

Configure so that all SSA devices are in a loop, *not* just connected in a string. Although SSA devices function connected in a string, a loop provides two paths of communications to each device for redundancy. The adapter chooses the shortest path to a disk.

SSA Fiber Optic Extenders

The SSA Fiber Optic Extenders use cables up to 2.4 Km to replace a single SSA cable. The SSA Fiber Optic Extender (Feature code 5500) is supported on all Model 7133 disk subsystems.

Using Fiber Optic extender, you can make the distance between disks greater than the LAN allows. If you do so, you can*not* use routers and gateways. Consequently, under these circumstances, you can*not* form an HACMP cluster between two LANs.

Daisy-chaining the Adapters

In each node, for each loop including that node, daisy-chain all its adapters. The SSAR router device uses another adapter when it detects that one adapter has failed. You need only one bypass switch for the whole daisy chain of adapters in the node rather than a bypass switch for each individual adapter.

Bypass Cards in the 7133, Models D40 and T40 Disk Subsystems

Bypass cards maintain high availability when a node fails, when a node is powered off, or when the adapter(s) of a node fail. Connect the pair of ports of one bypass card into the loop that goes to and from one node. That is, connect the bypass card to only one node. If you are using more than one adapter in a node, remember to daisy-chain the adapters.

Avoid two possible conditions when a bypass card switches to bypass mode:

- Do *not* connect two independent loops through a bypass card. When the bypass card switches to bypass mode, you want it to reconnect the loop inside the 7133 disk subsystem, rather than connecting two independent loops. So both ports of the bypass card must be in the same loop.
- Dummy disks are connectors used to fill out the disk drive slots in a 7133 disk subsystem so the SSA loop can continue unbroken. Make sure that when a bypass card switches to bypass mode, it connects no more than three dummy disks consecutively in the same loop. Put the disks next to the bypass cards and dummy disks between real disks.

Configuring to Minimize Single Points of Failure

To minimize single points of failure, consider the following points:

- Use logical volume mirroring and place logical volume mirror copies on separate disks and in separate loops using separate adapters. In addition, it is a good idea to mirror between the front row and the back row of disks or between disk subsystems.
- Avoid having the bypass card itself be a single point of failure by using one of the following mechanisms:
  - With one loop. Put two bypass cards into a loop connecting to each node.
  - With two loops. Set up logical volume mirroring to disks in a second loop. Set each loop to go through a separate bypass card to each node.

Set the bypass cards to forced inline mode for the following configurations:

- When connecting multiple 7133 disk subsystems.
- When the disk drives in one 7133 Model D40 or Model T40 are *not* all connected to the same SSA loop. In this type of configuration, **forced inline** mode removes the risk of a fault condition, namely, that a shift to **bypass** mode may cause the disk drives of different loops to be connected.

Configuring for Optimal Performance

The following guidelines can help you configure your system for optimal performance:

- Review multiple nodes and SSA domains:
  - A node and the disks it accesses make up an SSA domain. For configurations containing shared disk drives and multiple nodes, minimize the path length from each node to the disk drives it accesses. Measure the path length by the number of disk drives and adapters in the path. Each device has to receive and forward the packet of data.
  - With multiple adapters in a loop, put the disks near the closest adapter and make that the one that access the disks. In effect, try to keep I/O traffic within the SSA domain. Although any host can access any disk it is best to minimize I/O traffic crossing over to other domains.

  - When multiple hosts are in a loop, set up the volume groups so that a node uses the closest disks. This prevents one node's I/O from interfering with another's.
- Distribute read and write operations evenly throughout the loops.
- Distribute disks evenly among the loops.
- Download microcode when you replace hardware.

To ensure that everything works correctly, install the latest filesets, fixes, and microcode for your disk subsystem.

Testing Loops

Test loops in the following way:

- Test all loop scenarios thoroughly, especially in multiple-node loops. Test for loop breakage (failure of one or more adapters).
- Test bypass cards for power loss in adapters and nodes to ensure that they follow configuration guidelines.

Planning for RAID and SSA Concurrent Volume Groups

When concurrent volume groups are created on AIX v.5.2 and up, they are created as enhanced concurrent mode volume groups by default. You should convert SSA concurrent volume groups to enhanced concurrent mode whenever possible to make use of its flexibility.

RAID concurrent mode volume groups are now functionally obsolete, since enhanced concurrent mode provides extra capabilities, but RAID will continue to be supported for some time. HACMP supports both RAID and SSA concurrent mode volume groups with some important limitations:

- A concurrent resource group that includes a node running a 64-bit kernel requires enhanced concurrent mode for any volume groups.
- SSA concurrent mode is only supported on SSA disks with a 32-bit kernel.
- The C-SPOC utility does *not* work with RAID concurrent volume groups. Convert these volume groups to enhanced concurrent mode. (Otherwise, AIX sees them an non-concurrent).

Assign unique non-zero node numbers with **ssar** on each cluster node: The number should match the HACMP node ID. This is true for both concurrent and non-concurrent volume groups. Use the following command to retrieve the HACMP node_id (root@rvhaix2h):

```
odmget -q "name = rvhaixh" HACMPnode
```

Note: If you specify the use of SSA disk fencing in your concurrent resource group, HACMP assigns the node numbers when you synchronize the resources.

For the steps to assign a node number, see Enabling SSA Disk Fencing.

SSA Disk Fencing in Concurrent Access Clusters

Preventing data integrity problems that can result from the loss of TCP/IP network communication is especially important in concurrent access configurations where multiple nodes have simultaneous access to a shared disk. Chapter 3: Planning Cluster Network Connectivity describes using HACMP-specific point-to-point networks to prevent partitioned clusters.

Concurrent access configurations using SSA disk subsystems can also use disk fencing to prevent data integrity problems that can occur in partitioned clusters. Disk fencing can be used with enhanced concurrent mode.

The SSA disk subsystem includes fence registers, one per disk, capable of permitting or disabling access by each of the 32 possible connections. Fencing provides a means of preventing uncoordinated disk access by one or more nodes.

The SSA hardware has a fencing command for automatically updating the fence registers. This command provides a tie-breaking function within the controller for nodes independently attempting to update the same fence register. A compare-and-swap protocol of the fence command requires that each node provide both the current and desired contents of the fence register. If competing nodes attempt to update a register at about the same time, the first succeeds, but the second fails because it does *not* know the revised contents.

Benefits of Disk Fencing

Disk fencing provides the following benefits to concurrent access clusters:

- It enhances *data security* by preventing nodes that are *not* active members of a cluster from modifying data on a shared disk. By managing the fence registers, the HACMP software can ensure that only the designated nodes within a cluster have access to shared SSA disks.
- It enhances *data reliability* by assuring that competing nodes do *not* compromise the integrity of shared data. By managing the fence registers HACMP can prevent uncoordinated disk management by partitioned clusters. In a partitioned cluster, communication failures lead separate sets of cluster nodes to believe they are the only active nodes in the cluster. Each set of nodes attempts to take over the shared disk, leading to race conditions. The disk fencing tie-breaking mechanism arbitrates race conditions, ensuring that only one set of nodes gains access to the disk.

SSA Disk Fencing Implementation

The HACMP software manages the content of the fence registers. At cluster configuration, the fence registers for each shared disk are set to allow access for the designated nodes. As cluster membership changes as nodes enter and leave the cluster, the event scripts call the **cl_ssa_fence** utility to update the contents of the fence register. If the fencing command succeeds, the script continues processing. If the operation fails, the script exits with failure, causing the cluster to go into reconfiguration.

For information on how HACMP processes SSA disk fencing and how this is reflected in the **hacmp.out** file, see the section JOB_TYPE= SSA_FENCE in Chapter 2: Using Cluster Log Files in the *Troubleshooting Guide*.

Disk Fencing with SSA Disks in Concurrent Mode

You can only use SSA disk fencing under these conditions:

- Only disks contained in concurrent mode volume groups will be fenced.
- All nodes of the cluster must be configured to have access to these disks and to use disk fencing.
- All resource groups with the disk fencing attribute enabled must be concurrent access resource groups.
- Concurrent access resource groups must contain *all* nodes in the cluster. The verification utility issues an error if disk fencing is activated and the system finds nodes that are *not* included in the concurrent resource group.

The purpose of SSA disk fencing is to provide a safety lockout mechanism for protecting shared SSA disk resources in the event that one or more cluster nodes become isolated from the rest of the cluster.

Concurrent mode disk fencing works as follows:

- The first node up in the cluster fences out all other nodes of the cluster from access to the disks of the concurrent access volume group(s) for which fencing is enabled, by changing the fence registers of these disks.
- When a node joins a cluster, the active nodes in the cluster allow the joining node access by changing the fence registers of all disks participating in fencing with the joining node.
- When a node leaves the cluster, regardless of how it leaves, the remaining nodes that share access to a disk with the departed node should fence out the departed node as soon as possible.
- If a node is the last to leave a cluster, whether the cluster services are stopped with resource groups brought offline or placed in an UNMANAGED state, it clears the fence registers to allow access by all nodes. Of course, if the last node stops unexpectedly (is powered off or crashes, for example), it does *not* clear the fence registers. In this case, manually clear the fence registers using the appropriate SMIT options. For more information, see Chapter 1: Troubleshooting HACMP Clusters in the *Troubleshooting Guide*.

Enabling SSA Disk Fencing

The process of enabling SSA disk fencing for a concurrent resource group requires that *all volume groups containing SSA disks on cluster nodes must be varied off and the cluster must be down* when the cluster resources are synchronized. Note that this means all volume groups containing any of the SSA disks whether concurrent or non-concurrent, whether configured as part of the cluster or *not*, must be varied off for the disk fencing enabling process to succeed during the synchronization of cluster resources. If these conditions are *not* met, you have to reboot the nodes to enable fencing.

Note: If disk fencing is enabled and *not* all nodes are included in the concurrent access resource group, you receive an error upon verification of cluster resources.

The process of disk fencing enabling takes place on each cluster node as follows:

Assign a **node_number** to the **ssar** that matches the **node_id** of the node in the HACMP configuration. Do the following to assign node numbers:

1. Issue the command:

   ```
   chdev -l ssar -a node_number=x
   ```

   where *x* is the number to assign to that node.

   Any **node_numbers**, set before enabling disk fencing for purposes of replacing a drive or C-SPOC concurrent LVM functions, will be changed for disk fencing operations. The other operations will *not* be affected by this **node_number** change.
2. First remove, then remake all hdisks, pdisks, ssa adapter, and tmssa devices of the SSA disk subsystem seen by the node, thus picking up the node_number for use in the fence register of each disk.
3. Reboot the system.

Disk Fencing and Dynamic Reconfiguration

When a node is added to the cluster through dynamic reconfiguration while cluster nodes are up, the disk fencing enabling process is performed on the added node only, during the synchronizing of topology.

Any **node_numbers** that were set before enabling disk fencing (for purposes of replacing a drive or C-SPOC concurrent LVM functions) will be changed for disk fencing operations. Therefore, when initially setting SSA disk fencing in a resource group, the resources must be synchronized while the cluster is *down*. The other operations will *not* be affected by this **node_number** change.

Completing the Disk Worksheets

After determining the disk storage technology you will include in your cluster, complete all of the appropriate worksheets as follows:

- Completing the Shared SCSI Disk Worksheet
- Completing the Shared SCSI Disk Array Worksheet
- Completing the IBM SSA Disk Subsystems Worksheet.

Completing the Shared SCSI Disk Worksheet

Complete a Shared SCSI Disk Worksheet for each shared SCSI disk array.

To complete a Shared SCSI Disk Worksheet:

1. Enter the **Cluster name** in the appropriate field. This information was determined in Chapter 2: Initial Cluster Planning.
2. Check the appropriate field for the type of SCSI bus.
3. Fill in the host and adapter information including the **node name**, the number of the **slot** in which the disk adapter is installed and the **logical name** of the adapter, such as scsi0. AIX assigns the logical name when the adapter is configured.
4. Determine the SCSI IDs for all the devices connected to the SCSI bus.

5. Record information about the disk drives available over the bus, including the logical device name of the disk on every node. (This `hdisk` name is assigned by AIX when the device is configured and may vary on each node.)

Completing the Shared SCSI Disk Array Worksheet

Complete a Shared SCSI Disk Array Worksheet for each shared SCSI disk array.

To complete the Shared IBM SCSI Disk Arrays Worksheet:

1. Enter the **Cluster name** in the appropriate field. This information was determined in Chapter 2: Initial Cluster Planning.
2. Fill in the host and adapter information including the **node name**, the number of the **slot** in which the disk adapter is installed and the **logical name** of the adapter, such as `scsi0`. AIX assigns the logical name when the adapter is configured.
3. Assign SCSI IDs for all the devices connected to the SCSI bus. For disk arrays, the controller on the disk array are assigned the SCSI ID.
4. Record information about the LUNs configured on the disk array.
5. Record the logical device name AIX assigned to the array controllers.

Completing the IBM SSA Disk Subsystems Worksheet

Complete an IBM SSA Disk Subsystems Worksheet for each shared SSA configuration.

To complete the Shared IBM Serial Storage Architecture Disk Subsystems Worksheet:

1. Enter the **Cluster name** in the appropriate field. This information was determined in Chapter 2: Initial Cluster Planning.
2. Fill in the host and adapter information including the **node name**, the SSA adapter label, and the number of the **slot** in which the disk adapter is installed. Include dual-port number of the connection. This will be needed to make the loop connection clear.

Adding the Disk Configuration to the Cluster Diagram

Once you have chosen a disk technology, add your disk configuration to the cluster diagram you started in Chapter 2: Initial Cluster Planning.

For the cluster diagram, draw a box representing each shared disk; then label each box with a shared disk name.

Planning for Tape Drives as Cluster Resources

You can configure a tape drive as a cluster resource, making it highly available to multiple nodes in a cluster. SCSI streaming tape drives and Direct Fibre Channel Tape unit attachments are supported. Management of shared tape drives is simplified by the following HACMP functionality:

- Configuration of tape drives using SMIT

- 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 dynamic reconfiguration.

For information about completing the Shared Tape Drive Worksheet, see the section on Installing and Configuring Shared Tape Drives in the chapter on Configuring Installed Hardware in the *Installation Guide*.

Limitations

Note the following as you plan to include tape drives as cluster resources:

- Support is limited to SCSI or Direct Fibre Channel tape drives that have hardware reserve and hardware reset/release functions.
- A tape loader/stacker is treated like a simple tape drive by HACMP.
- No more than two cluster nodes can share the tape resource.
- Tape resources may *not* be part of concurrent resource groups.
- The tape drive must have the same name (for example, **/dev/rmt0**) on both nodes sharing the tape device.
- When a tape special file is closed, the default action is to release the tape drive. HACMP is *not* responsible for the state of the tape drive once an application has opened the tape.
- No means of synchronizing tape operations and application servers is provided. If you decide that a tape reserve/release should be done asynchronously, provide a way to notify the application server to wait until the reserve/release is complete.
- Tape drives with more than one SCSI interface are *not* supported. Therefore, only one connection exists between a node and a tape drive. The usual functionality of adapter fallover does *not* apply.

Reserving and Releasing Shared Tape Drives

When a resource group with tape resources is activated, the tape drive is reserved to allow its exclusive use. This reservation is held until an application releases it, or the node is removed from the cluster:

- When the special file for the tape is closed, the default action is to release the tape drive. An application can open a tape drive with a “do *not* release on close” flag. HACMP will *not* be responsible for maintaining the reservation after an application is started.
- Upon stopping cluster services on a node and bringing resource groups offline, the tape drive is released, allowing access from other nodes.
- Upon unexpected node failure, a forced release is done on the takeover node. The tape drive is then reserved as part of resource group activation.

Setting Tape Drives to Operate Synchronously or Asynchronously

If a tape operation is in progress when a tape reserve or release is initiated, it may take many minutes before the reserve or release operation completes. HACMP allows synchronous or asynchronous reserve and release operations. Synchronous and asynchronous operation is specified separately for reserve and release.

Synchronous Operation

With synchronous operation, (the default value), HACMP waits for the reserve or release operation, including the execution of a user defined recovery procedure, to complete before continuing.

Asynchronous Operation

With asynchronous operation, HACMP creates a child process to perform the reserve or release operation, including the execution of a user defined recovery procedure, and immediately continues.

Recovery Procedures

Recovery procedures are highly dependent on the application accessing the tape drive. Rather than trying to predict likely scenarios and develop recovery procedures, HACMP provides for the execution of user defined recovery scripts for the following:

- Tape start
- Tape stop.

Tape Start Scripts and Stop Scripts

Tape start and stop occurs during node start and stop, node fallover and reintegration, and dynamic reconfiguration. These scripts are called when a resource group is activated (tape start) or when a resource group is deactivated (tape stop). Sample start and stop scripts can be found in the **/usr/es/sbin/cluster/samples/tape** directory:

```
tape_resource_start_example
tape_resource_stop_example
```

- During tape start, HACMP reserves the tape drive, forcing a release if necessary, and then calls the user-provided tape start script.
- During tape stop, HACMP calls the user-provided tape stop script, and then releases the tape drive.

Note: You are responsible for correctly positioning the tape, terminating processes or applications writing to the tape drive, writing end of tape marks, etc., within these scripts.

Other application-specific procedures should be included as part of the start server and stop server scripts.

Adapter Fallover and Recovery

Tape drives with more than one SCSI interface are *not* supported. Therefore, only one connection exists between a node and a tape drive. The usual notion of adapter fallover does *not* apply.

Node Fallover and Recovery

If a node that has tape resources that are part of an HACMP resource group fails, the takeover node will reserve the tape drive, forcing a release if necessary, and then calls the user-provided tape start script.

On reintegration of a node, the takeover node runs the tape stop script and then releases the tape drive. The node being reintegrated reserves the tape drive and calls the user-provided tape start script.

Network Fallover and Recovery

HACMP does *not* provide tape fallover and recovery procedures for network failure.

Where You Go from Here

You have now planned your shared disk configuration. The next step is to plan the shared volume groups for your cluster. This step is described in Chapter 5: Planning Shared LVM Components.

Chapter 5: Planning Shared LVM Components

This chapter describes planning shared volume groups for an HACMP cluster. This chapter contains the following sections:

- Prerequisites
- Overview
- Planning for LVM Components
- Planning LVM Mirroring
- Planning for Disk Access
- Using Fast Disk Takeover
- Using Quorum and Varyon to Increase Data Availability
- Using NFS with HACMP
- Completing the Shared LVM Components Worksheets
- Adding LVM Information to the Cluster Diagram
- Where You Go from Here

Note: If you are planning an IBM General Parallel File System (GPFS) cluster, see the appendix on network requirements in GPFS Cluster Configuration in the *Installation Guide*.

If you are planning to use OEM disks, volume groups, or file systems in your cluster (including Veritas volumes) see the appendix on OEM Disk, Volume Group, and File Systems Accommodation in the *Installation Guide*.

Prerequisites

At this point, you should have completed the planning steps in the previous chapters:

- Chapter 2: Initial Cluster Planning
- Chapter 3: Planning Cluster Network Connectivity
- Chapter 4: Planning Shared Disk and Tape Devices

You should also be familiar with how to use the Logical Volume Manager (LVM). For information about AIX LVM, see the *AIX System Management Guide*.

Overview

Planning shared LVM components for an HACMP cluster depends on both of the following:

- The type of shared disk device
- The method of shared disk access

To avoid a single point of failure for data storage, use data redundancy as supported by LVM or your storage system.

Planning for LVM Components

The LVM controls disk resources by mapping data between physical and logical storage. *Physical storage* refers to the actual location of data on a disk. *Logical storage* controls how data is made available to the user. Logical storage can be discontiguous, expanded, replicated, and can span multiple physical disks. These facilities provide improved availability of data.

The LVM organizes data into the following components:

- Physical Volumes
- Volume Groups
- Logical Volumes
- File Systems

Physical Volumes

A physical volume is a single physical disk or a logical unit presented by a storage array. The physical volume is partitioned to provide AIX with a way of managing how data is mapped to the volume. The following figure shows a conventional use of physical partitions within a physical volume.

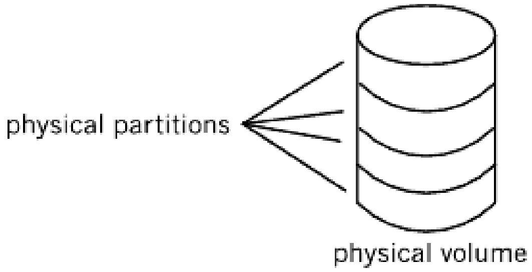


Figure 12. Physical Partitions on a Physical Volume

When planning shared physical volumes, ensure that:

- The list of PVIDs for a volume group is identical on all cluster nodes that have access to the shared physical volume
- The setting for the concurrent attribute of the volume group is consistent across all related cluster nodes.

Volume Groups

A volume group is a set of physical volumes that AIX treats as a contiguous, addressable disk region. You can place from one to 32 physical volumes in the same volume group.

The following figure shows a volume group of three physical volumes:

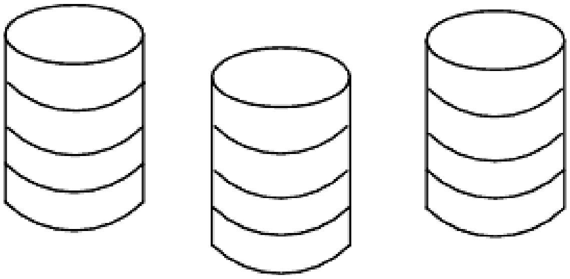


Figure 13. Volume Group of Three Physical Volumes

In the HACMP environment, a *shared volume group* is a volume group that resides entirely on the external disks shared by the cluster nodes. A non-concurrent shared volume group can be varied on by only one node at a time.

When working with a shared volume group:

- Do *not* include an internal disk in a shared volume group, because it cannot be accessed by other nodes. If you include an internal disk in a shared volume group, the **varyonvg** command fails.
- Do *not* activate (vary on) the shared volume groups in an HACMP cluster manually at system boot. Use cluster event scripts to do this.
- Ensure that the automatic varyon attribute in the AIX ODM is set to **No** for shared volume groups listed within a resource group. The HACMP cluster verification utility automatically corrects this attribute for you upon verification of cluster resources and sets the automatic varyon attribute to **No**.
- If you define a volume group to HACMP, do *not* manage it manually on any node outside of HACMP while HACMP is running on other nodes. This can lead to unpredictable results. If you want to perform actions on a volume group independent of HACMP, stop the cluster services, perform a manual volume group management task, leave the volume group varied off, and restart HACMP. To ease the planning of HACMP's use of physical volumes, the verification utility checks for:
  - Volume group consistency
  - Disk availability.

Logical Volumes

A *logical volume* is a set of logical partitions that AIX makes available as a single storage unit—that is, the logical view of a disk. A logical partition is the logical view of a physical partition. 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 or a raw device.

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.

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

You want to plan shared file systems to 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.

The order in which file systems are mounted is usually not important. However, if this is important to your cluster, you need to plan for some things:

- File systems that exist within a single resource group are mounted in alphanumeric order when the resource group comes online. They are also unmounted in reverse alphanumeric order when the resource group is taken offline.
- If you have shared, nested file systems, then additional care is needed. If you have shared, nested file systems within a single resource group, then you must set the **Filesystems Recovery Method** for the resource group to **sequential** to guarantee the proper mount order.
- If you have nested file systems that reside in different resource groups, then you must additionally plan a parent-child relationship for those resource groups to guarantee the proper mount order.

Planning LVM Mirroring

LVM mirroring provides the ability to allocate more than one copy of a physical partition to increase the availability of the data. When a disk fails and its physical partitions become unavailable, you still have access to mirrored data on an available disk. The LVM performs mirroring within the logical volume. Within an HACMP cluster, you mirror:

- Logical volume data in a shared volume group
- The log logical volume for each shared volume group with file systems.

Note: LVM mirroring does *not* apply to the IBM 2105 Enterprise Storage Servers, TotalStorage DS4000, 6000, 8000 and other disk devices that use RAID, which provide their own data redundancy.

Mirroring Physical Partitions

To improve the availability of the logical volume, you allocate one, two, or three copies of a physical partition to mirror data contained in the partition. If a copy is lost due to an error, the other undamaged copies are accessed, and AIX continues processing with an accurate copy. After access is restored to the failed physical partition, AIX resynchronizes the contents (data) of the physical partition with the contents (data) of a consistent mirror copy.

The following figure shows a logical volume composed of two logical partitions with three mirrored copies. In the diagram, each logical partition maps to three physical partitions. Each physical partition should be designated to reside on a separate physical volume within a single volume group. This configuration provides the maximum number of alternative paths to the mirror copies and, therefore, the greatest availability.

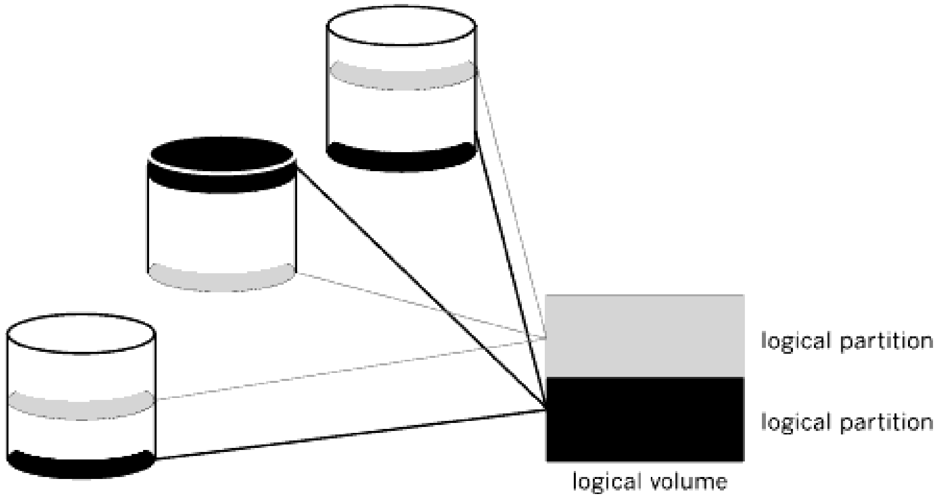


Figure 14. Logical Volume of Two Logical Partitions with Three Mirrored Copies

The mirrored copies are transparent, meaning that you cannot isolate one of these copies. For example, if a user deletes a file from a logical volume with multiple copies, the deleted file is removed from all copies of the logical volume.

The following configurations increase data availability:

- Allocating three copies of a logical partition rather than allocating one or two copies.
- Allocating the copies of a logical partition on different physical volumes rather than allocating the copies on the same physical volume.
- Allocating the copies of a logical partition across different physical disk enclosures instead of the same enclosure, if possible.
- Allocating the copies of a logical partition across different disk adapters. rather than using a single disk adapter.

Although using mirrored copies spanning multiple disks (on separate power supplies) together with multiple disk adapters ensures that no disk is a single point of failure for your cluster, these configurations may increase the time for write operations.

Specify the **superstrict** disk allocation policy for the logical volumes in volume groups for which forced varyon is specified. This configuration:

- Guarantees that copies of a logical volume always reside on separate disks
- Increases the chances that forced varyon will be successful after a failure of one or more disks.

If you plan to use forced varyon for the logical volume, apply the **superstrict** disk allocation policy for disk enclosures in the cluster.

For more information about forced varyon, see the section Using Quorum and Varyon to Increase Data Availability.

Mirroring Journal Logs

Non-concurrent access configurations support journaled file systems and enhanced journaled file systems. AIX uses journaling for its file systems. In general, this means that the internal state of a file system at startup (in terms of the block list and free list) is the same state as at shutdown. In practical terms, this means that when AIX starts up, the extent of any file corruption can be no worse than at shutdown.

Each volume group contains a **jfslog** or **jfs2log**, which is itself a logical volume. This log typically resides on a different physical disk in the volume group than the journaled file system. However, if access to that disk is lost, changes to file systems after that point are in jeopardy.

To avoid the possibility of that physical disk being a single point of failure, you can specify mirrored copies of each **jfslog** or **jfs2log**. Place these copies on separate physical volumes.

Mirroring across Sites

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). 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. These remote disks can be combined into a volume group via the AIX Logical Volume Manager, and this volume group may 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. The automatic synchronization is *not* possible for all cases, but you can use C-SPOC to synchronize the data from the surviving mirrors to stale mirrors after a disk or site failure and subsequent reintegration.

See the *Administration Guide* for information about configuration. Plan the sites and nodes ahead of time, and include this information on the Shared Volume Group/File System Worksheet.

Note: In HACMP/XD, you can also use mirroring in a cluster that spans two sites, using the Geographic Logical Volume Manager (GLVM) mirroring function. For more information on GLVM, see Planning Cluster Sites or the *HACMP/XD for GLVM Planning and Administration Guide*.

Planning for Disk Access

You can configure disks to have the following types of access:

- Enhanced Concurrent Access. The data on the disks is available to all connected nodes concurrently and all the nodes have access to the metadata on the disks. This access mode allows for fast disk takeover, because the volume group can be brought online before the metadata is read.

  Typically, all volume groups should be configured for enhanced concurrent mode. In HACMP 5.1 and up, enhanced concurrent mode is the default for creating concurrent volume groups. You can also convert your migrated volume groups to enhanced concurrent mode.

  Concurrent access configurations do *not* support journaled file systems. Concurrent access configurations that use IBM 7131-405 and 7133 SSA serial disk subsystems should use LVM mirroring.

  Concurrent access configurations that use IBM TotalStorage DS Series or IBM 2105 Enterprise Storage Servers do *not* use LVM mirroring; instead, these systems provide their own data redundancy.

  Note: See the IBM website for announcements and information about new storage devices.

- Non-Concurrent Access. Only one node at a time can access information on the disks.

  If the resource group containing those disks moves to another node, the new node can then access the disks, read the metadata (information about the current state of the volume groups and other components), and then vary on the volume groups and mount any associated file systems.

  Non-concurrent access configurations typically use journaled file systems. (In some cases, a database application running in a non-concurrent environment may bypass the journaled file system and access the raw logical volume directly.)

Enhanced Concurrent Access

Any disk supported by HACMP for attachment to multiple nodes can be an enhanced concurrent mode volume group, and can be used in either concurrent or non-concurrent environments (as specified by the type of resource group):

- Concurrent. An application runs on all active cluster nodes at the same time.

  To allow such applications to access their data, concurrent volume groups are varied on all active cluster nodes. The application has the responsibility to ensure consistent data access.

- Non-concurrent. An application runs on one node at a time.

  The volume groups are *not* concurrently accessed, they are still accessed by only one node at any given time.

When you vary on the volume group in enhanced concurrent mode on all nodes that own the resource group in a cluster, the LVM allows access to the volume group on all nodes. However, it restricts the higher-level connections, such as NFS mounts and JFS mounts, on all nodes, and allows them only on the node that currently owns the volume group in HACMP.

About Enhanced Concurrent Mode

In AIX v.5.2 and up, all concurrent volume groups are created as enhanced concurrent mode volume groups by default. For enhanced concurrent volume groups, the Concurrent Logical Volume Manager (CLVM) coordinates 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.

Enhanced concurrent mode replaces the special facilities provided by concurrent mode for SSA. Also, note that:

- SSA concurrent mode is *not* supported on operating systems with 64-bit kernels.
- If you are running AIX v.5.2 or greater, you cannot create new SSA concurrent mode volume groups. You can convert these volume groups to enhanced concurrent mode.

  If you are running AIX v. 5.2, you can continue to use SSA concurrent mode volume groups created on AIX v.5.1. If you are running AIX v.5.3, you *must convert* all volume groups to enhanced concurrent mode.

Use C-SPOC to convert both SSA and RAID concurrent volume groups to enhanced concurrent mode. For information about converting volume groups to enhanced concurrent mode, see the section Converting Volume Groups to Enhanced Concurrent Mode in Chapter 12: Managing Shared LVM Components in a Concurrent Access Environment in the *Administration Guide*.

Partitioned Clusters with Enhanced Concurrent Access

Because Group Services protocols flow over the communications links between cluster nodes and *not* through the disks themselves, take steps to avoid partitioned clusters that include enhanced concurrent mode volume groups:

- Use multiple IP networks.
- *Do not* make online changes to an enhanced concurrent mode volume group unless all cluster nodes are online.

When fast disk takeover is used, the SCSI disk reservation functionality is *not* used. If the cluster becomes partitioned, nodes in each partition could accidentally vary on the volume group in active state. Because active state varyon of the volume group allows mounting of file systems and changing physical volumes, this situation can result in different copies of the same volume group. For more information about fast disk takeover and using multiple networks, see the section Using Fast Disk Takeover.

Non-Concurrent Access

Journaled file systems support only non-concurrent access. The JFS and JFS2 file systems do *not* coordinate their access between nodes. As a result, if a JFS or JFS2 file system was mounted on two or more nodes simultaneously, the two nodes could allocate the same block to different files.

Using Fast Disk Takeover

In HACMP 5.1 and up, HACMP automatically detects failed volume groups and initiates a fast disk takeover for enhanced concurrent mode volume groups that are included as resources in non-concurrent resource groups. Fast disk takeover requires:

- AIX v.5.2 and up
- HACMP 5.1 and up with the Concurrent Resource Manager component installed on all nodes in the cluster
- Enhanced concurrent mode volume groups in non-concurrent resource groups.

For existing volume groups included in non-concurrent resource groups, convert these volume groups to enhanced concurrent volume groups *after upgrading* your HACMP software. For information about converting volume groups, see the section Converting Volume Groups to Enhanced Concurrent Mode in Chapter 11: Managing Shared LVM Components in a Concurrent Access Environment in the *Administration Guide*.

Fast disk takeover is especially useful for fallover of enhanced concurrent volume groups made up of a large number of disks. This disk takeover mechanism is faster than disk takeover used for standard volume groups included in non-concurrent resource 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.

Fast disk takeover has been observed to take no more than ten seconds for a volume group with two disks. This time is expected to increase very slowly for larger numbers of disks and volume groups. The actual time observed in any configuration depends on factors outside of HACMP control, such as the processing power of the nodes and the amount of unrelated activity at the time of the fallover. The actual time observed for completion of fallover processing depends on additional factors, such as whether or *not* a file system check is required, and the amount of time needed to restart the application.

Note: Enhanced concurrent mode volume groups are *not* concurrently accessed, they are still accessed by only one node at any given time. The fast disk takeover mechanism works at the volume group level, and is thus independent of the number of disks used.

Fast Disk Takeover and Active and Passive Varyon

An enhanced concurrent volume group can be made active on a node, or varied on, in two ways:

- Active
- Passive.

To enable fast disk takeover, HACMP activates enhanced concurrent volume groups in the active and passive states:

Active Varyon

Active varyon behaves the same as ordinary varyon, and makes the logical volumes available. When an enhanced concurrent volume group is varied on in active state on a node, it allows the following:

- Operations on file systems, such as file system mounts
- Operations on applications

- Operations on logical volumes, such as creating logical volumes
- Synchronizing volume groups.

Passive Varyon

When an enhanced concurrent volume group is varied on in passive state, the LVM provides the equivalent of disk fencing for the volume group at the LVM level.

Passive state varyon allows only a limited number of read-only operations on the volume group:

- LVM read-only access to the volume group's special file
- LVM read-only access to the first 4K of all logical volumes that are owned by the volume group.

The following operations are *not* allowed when a volume group is varied on in passive state:

- Operations on file systems, such as file systems mounting
- Any operations on logical volumes, such as having logical volumes open
- Synchronizing volume groups.

HACMP and Active and Passive Varyon

HACMP correctly varies on the volume group in active state on the node that owns the resource group, and changes active and passive states appropriately as the state and location of the resource group changes.

- Upon cluster startup:
  - On the node that owns the resource group, HACMP activates the volume group in active state. Note that HACMP activates a volume group in active state only on one node at a time.
  - HACMP activates the volume group in passive state on all other nodes in the cluster.
- Upon fallover:
  - If a node releases a resource group, or, if the resource group is being moved to another node for any other reason, HACMP switches the varyon state for the volume group from active to passive on the node that releases the resource group, and activates the volume group in active state on the node that acquires the resource group.
  - The volume group remains in passive state on all other nodes in the cluster.
- Upon node reintegration, HACMP does the following:
  - Changes the varyon state of the volume group from active to passive on the node that releases the resource group
  - Varies on the volume group in active state on the joining node
  - Activates his volume group in passive state on all other nodes in the cluster.

Note: The switch between active and passive states is necessary to prevent mounting file systems on more than one node at a time.

Disk Takeover with Breaking Disk Reserves

In HACMP 5.1 and up, processing for *regular* disk takeover takes place (as opposed to fast disk takeover) in the following cases:

- Concurrent Resource Manager as part of HACMP 5.1 and up is *not* installed on the nodes in the cluster

- You did *not* convert the volume groups that are included in non-concurrent resource groups to enhanced concurrent mode.

The regular disk takeover processing requires breaking the disk reserves and checking the logical partitions to determine changes made to the volume groups. Also, prior to fallover, HACMP uses lazy update to update the LVM information on cluster nodes. When a lazy update is performed, prior to fallover HACMP processes changes made to the volume groups, and synchronizes the LVM information on cluster nodes.

Using Quorum and Varyon to Increase Data Availability

How you configure quorum and varyon for volume groups can increase the availability of mirrored data.

Using Quorum

Quorum ensures that more than half of the physical disks in a volume group are available. It does *not* keep track of logical volume mirrors, and is therefore *not* a useful way to ensure data availability. You can lose quorum when you still have all your data. Conversely, you can lose access to some of your data, and *not* lose quorum.

Quorum is beneficial for volume groups on RAID arrays, such as the ESS and IBM TotalStorage DS Series. Note that the RAID device provides data availability and recovery from loss of a single disk. Mirroring is typically *not* used for volume groups contained entirely within a single RAID device. If a volume group is mirrored between RAID devices, forced varyon can bring a volume group online despite loss of one of the RAID devices.

Decide whether to enable or disable quorum for each volume group. The following table shows how quorum affects when volume groups vary on and off:

	Condition for volume group to vary on	**Condition for volume group to vary off**
Quorum enabled	More than 50% of the disks in the volume group are available	Access is lost to 50% or more of the disks
Quorum disabled	All of the disks in the volume group are available	Access is lost to all of the disks

Quorum checking is enabled by default. You can disable quorum by using the **chvg -Qn vgname** command, or by using the **smit chvg** fastpath.

Quorum in Concurrent Access Configurations

Quorum must be enabled for an HACMP concurrent access configuration. Disabling quorum could result in data corruption. Any concurrent access configuration where multiple failures could result in no common shared disk between cluster nodes has the potential for data corruption or inconsistency.

The following figure shows a cluster with two sets of IBM SSA disk subsystems configured for no single point of failure. The logical volumes are mirrored across subsystems and each disk subsystem is connected to each node with separate NICs.

Figure 15. IBM SSA Concurrent Access Configuration

If multiple failures result in a communications loss between each node and one set of disks in such a way that Node A can access subsystem 1 but *not* subsystem 2, and Node B can access subsystem 2 but *not* subsystem 1. Both nodes continue to operate on the same baseline of data from the mirrored copy they can access. However, each node does *not* see modifications made by the other node to data on disk. As a result, the data becomes inconsistent between nodes.

With quorum protection enabled, the communications failure results in one or both nodes varying off the volume group. Although an application does *not* have access to data on the volume group that is varied off, data consistency is preserved.

Selective Fallover Triggered by Loss of Quorum

HACMP selectively provides recovery for non-concurrent resource groups (with the startup policy *not* Online on All Available Nodes) that are affected by failures of specific resources. HACMP 4.5 and up automatically reacts to a "loss of quorum" LVM_SA_QUORCLOSE error associated with a volume group going offline on a cluster node. In response to this error, a non-concurrent resource group goes offline on the node where the error occurred.

If the AIX Logical Volume Manager takes a volume group in the resource group offline due to a loss of quorum for the volume group on the node, HACMP selectively moves the resource group to another node. You can change this default behavior by customizing resource recovery to use a notify method instead of fallover. For more information, see Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*.

Note: HACMP launches selective fallover and moves the affected resource group *only* in the case of the LVM_SA_QUORCLOSE error. This error occurs if you use mirrored volume groups with quorum enabled. However, other types of "volume group failure" errors could occur. HACMP does *not* react to any other type of volume group errors automatically. In these cases, you still need to configure customized error notification methods, or use AIX Automatic Error Notification methods to react to volume group failures.

For more information about LVM_SA_QUORCLOSE errors, see the section on Error Notification Method Used for Volume Group Loss in the *Installation Guide*.

For more information about selective fallover triggered by loss of quorum for a volume group on a node, see the section Selective Fallover for Handling Resource Groups in the Appendix B: Resource Group Behavior During Cluster Events in the *Administration Guide*.

Using Forced Varyon

HACMP 5.1 and up provides a forced varyon function to use in conjunction with AIX Automatic Error Notification methods. The forced varyon function enables you to have the highest possible data availability. Forcing a varyon of a volume group lets you keep a volume group online as long as there is one valid copy of the data available. Use a forced varyon only for volume groups that have mirrored logical volumes.

Note: Use caution when using this facility to avoid creating a partitioned cluster.

You can use SMIT to *force* a varyon of a volume group on a node if the normal **varyon** command fails on that volume group due to a lack of quorum but with one valid copy of the data available. Using SMIT to force a varyon is useful for local disaster recovery—when data is mirrored between two disk enclosures, and one of the disk enclosures becomes unavailable.

For more information about the circumstances under which you can forcefully activate a volume group in HACMP safely, see the section Forcing a Varyon of a Volume Group in Chapter 14: Managing Resource Groups in a Cluster in the *Administration Guide.*

Note: You can specify a forced varyon attribute for volume groups on SSA or SCSI disks that use LVM mirroring, and for volume groups that are mirrored between separate RAID or ESS devices.

If you want to force the volume group to vary on when disks are unavailable, use **varyonvg -f**, which will force the volume group to vary on, whether or *not* there are copies of your data. You can specify forced varyon in SMIT for volume groups in a resource group.

Forced Varyon and Cluster Partitioning

If you have enabled forced varyon in HACMP, ensure that a heartbeating network exists. A heartbeating network ensures that each node always has a communication path to the other nodes—even if a network fails. This prevents your cluster from becoming partitioned. Otherwise, a network failure may cause nodes to attempt to take over resource groups that are still active on other nodes. In this situation, if you have set a forced varyon setting, you may experience data loss or divergence.

Other Ways to Force a Varyon

To achieve a forced varyon of a volume group, you can continue using methods that existed before HACMP 5.1 by using either:

- Pre- or post- event scripts
- Event recovery routines to respond to failure of the activation and acquisition of raw physical volumes and volume groups on a node.

Using HACMP forced varyon eliminates the need for a *quorum buster disk*, which was added to the cluster to avoid the problems associated with the loss of quorum. A quorum buster disk was a single additional disk added to the volume group, on a separate power and field replaceable unit (FRU) from either of the mirrors of the data. This disk contained no data, it simply served as a *quorum buster* so that if one enclosure failed, or connectivity to it was lost, quorum was maintained and the data remained available on the other disk enclosure.

Using NFS with HACMP

The HACMP software provides availability enhancements to NFS handling:

- Reliable NFS server capability that allows a backup processor to recover current NFS activity should the primary NFS server fail, preserving the locks on NFS file systems and the duplicate request cache. This functionality is restricted to two-node Resource Groups if it contains NFSv2/v3 exports. Resource Groups with only NFSv4 can support up to 32-node configurations.
- NFS Configuration Assist to ease the setup and configuration.
- Pre-configured application server application monitor (**clam_nfsv4**) to monitor NFSv4 exports and the NFS daemons.
- Ability to specify a network for NFS mounting.
- Ability to define NFS exports and mounts at the directory level.
- Ability to specify export options for NFS-exported directories and file systems.

For NFS to work as expected on an HACMP cluster, there are specific configuration requirements, so you can plan accordingly for:

- Creating shared volume groups
- Exporting NFS file systems
- NFS mounting and fallover.

The HACMP scripts address default NFS behavior. You may need to modify the scripts to handle your particular configuration. The following sections provide suggestions for planning for a variety of situations.

You can configure NFS in all resource groups that behave as non-concurrent; that is, they do *not* have an Online on All Available Nodes startup policy.

For information about using NFS Version 4, see Chapter 2 of the *Installation Guide*.

Relinquishing Control over NFS File Systems in an HACMP Cluster

Once you configure resource groups that contain NFS file systems, you relinquish control over NFS file systems to HACMP.

Once NFS file systems become part of resource groups that belong to an active HACMP cluster, HACMP takes care of cross-mounting and unmounting the file systems, during cluster events (such as fallover of a resource group containing the file system to another node in the cluster).

If for some reason you stop the cluster services and must manage the NFS file systems manually, the file systems must be unmounted before you restart the cluster services. This enables management of NFS file systems by HACMP once the nodes join the cluster.

Reliable NFS Server Capability

An HACMP cluster can take advantage of AIX extensions to the standard NFS functionality that enable it to handle duplicate requests correctly and restore lock state during NFS server fallover and reintegration.

When NFS clients use NFS locking to arbitrate access to the shared NFS file system, there is a limit of two nodes per resource group; each resource group that uses Reliable NFS contains one pair of HACMP nodes.

Independent pairs of nodes in the cluster can provide Reliable NFS services. For example, in a four node cluster, you can set up two NFS client/server pairs (for example, Node A/NodeB provides one set of Reliable NFS services, and NodeC/NodeD can provide another set of Reliable NFS services.) Pair 1 can provide reliable NFS services for one set of NFS file systems, and Pair 2 can provide Reliable NFS services to another set of NFS file systems.This is true whether or not NFS cross-mounting is configured, per se. HACMP does not impose a limit to the number of resource groups or NFS file systems, as long as the nodes participating in the resource groups follow the above constraints.

Specifying an IP Address for NFS

Although HACMP has no dependency on hostname, the hostname must be able to be resolved to an IP address that is always present on the node and always active on an interface. This means that it cannot be a service IP address that may move to another node, for example. To ensure that the IP address that is going to be used by NFS always resides on the node you can:

- Use an IP address that is associated with a persistent label
- For an IPAT via Aliases configuration, use the IP address used at boot time
- Use an IP address that resides on an interface that *is not controlled* by HACMP.

Shared Volume Groups

When creating shared volume groups, typically you can leave the **Major Number** field blank and let the system provide a default. However, NFS uses volume group major numbers to help uniquely identify exported file systems. Therefore, all nodes to be included in a resource group containing an NFS-exported file system must have the same major number for the volume group on which the file system resides.

In the event of node failure, NFS clients attached to an HACMP cluster operate the same way as when a standard NFS server fails and reboots. That is, accesses to the file systems hang, then recover when the file systems become available again. However, if the major numbers are *not* the same, when another cluster node takes over the file system and re-exports the file system, the client application will *not* recover, since the file system exported by the node will appear to be different from the one exported by the failed node.

NFS Exporting File Systems and Directories

The process of NFS-exporting file systems and directories in HACMP is different from that in AIX. Keep in mind the following points when planning for NFS-exporting in HACMP:

- File systems and directories to NFS-export:

In AIX, you specify file systems and directories to NFS-export by using the **smit mknfsexp** command (which creates the **/etc/exports** file). In HACMP, you specify file systems and directories to NFS-export by including them in a resource group in HACMP. For more information, see the section NFS Cross-Mounting in HACMP.

- Export options for NFS exported file systems and directories:

  If you want to specify special options for NFS-exporting in HACMP, you can create a **/usr/es/sbin/cluster/etc/exports** file. This file has the same format as the regular AIX **/etc/exports** file.

  Note: Using this alternate exports file is optional. HACMP checks the **/usr/es/sbin/cluster/etc/exports** file when NFS-exporting a file system or directory. If there is an entry for the file system or directory in this file, HACMP uses the options listed, except the HACMP 5.4.1 and later might ignore the version option as described in Steps for Adding Resources and Attributes to Resource Groups (Extended Path) in Chapter 5: Adding Resources and Attributes to Resource Groups Using the Extended Path of the *Administration Guide*. If the file system or directory for NFS-export is *not* listed in the file; or, if the alternate file does *not* exist, the file system or directory is NFS-exported with the default option of root access for all cluster nodes.

- A resource group that specifies file systems to export:

  In SMIT, set the **Filesystems Mounted before IP Configured** field for the resource group to **true**. This ensures that IP address takeover is performed *after* exporting the file systems. If the IP addresses were managed first, the NFS server would reject client requests until the file systems had been exported.

- Stable Storage for NFSv4 exports:

  It is recommended that you use Stable Storage for NFSv4 exports and have it accessible to all the participating nodes of the Resource Group. NFSv4 uses this file system space to store the state information related to the NFS client transactions. The state information in the Stable Storage is crucial for the smooth fallover/fallback/move of the Resource Group from one node to other node, while keeping the NFSv4 client's state unaffected.

  While the Resource Group is online, the location of the Stable Storage cannot be changed.

NFS and Fallover

For HACMP and NFS to work properly together:

- NFS requires configuring resource groups with IP Address Takeover (IP Replacement or IP Aliases).
- IPAT via IP Aliases with NFS has specific requirements. For information about these requirements, see the section NFS Cross-Mounting and IP Labels.

To ensure the best NFS performance, NFS file systems used by HACMP should include the entry `vers = <version number>` in the **options** field in the **/etc/filesystems** file.

NFS Cross-Mounting in HACMP

NFS cross-mounting is an HACMP-specific NFS configuration where each node in the cluster can act as both NFS server and NFS client. While a file system is being exported from one node, the file system is NFS mounted on all the nodes of the Resource Group, including the one that is exporting it. Another file system can also be exported from other node, and be mounted on all nodes.

In the presence of NFSv2/v3 exports in the Resource Group, this cross-mount capability is restricted to only two node Resource Groups. If the Resource Group contains only NFSv4 exports, then the cross-mount capability can be extended up to 32-node Resource Groups.

Essentially, each node in the Resource group is part of a mutual takeover (or active-active) cluster configuration, providing and mounting an NFS file system.

By default, resource groups that contain NFS exported file systems automatically cross-mount these file systems (if both export and import are configured):

- On the node currently hosting the resource group, all NFS file systems in the group are NFS exported.
- Each node that may host this resource group NFS mounts all the NFS file systems in the resource group.

This lets applications access the NFS file systems on any node that is part of the resource group.

With IP address takeover configured for the resource group, on fallover:

- The NFS file system is locally mounted by the takeover node and re-exported.
- All other nodes in the resource group maintain their NFS mounts.

Two Node NFS Cross-Mounting Example

In the following figure:

- NodeA currently hosts a non-concurrent resource group, RG1, which includes:
  - `/fs1` as an NFS exported file system
  - `service1` as an service IP label
- NodeB currently hosts a non-concurrent resource group, RG2, which includes:
  - `/fs2` as an NFS exported file system
  - `service2` as an service IP label

Both resource groups contain both nodes as possible owners of the resource groups.

Figure 16. Two Node NFS Cross-Mounting

The two resource groups would be defined in SMIT as follows:

Resource Group	**RG1**	**RG2**
Participating node names	`NodeA NodeB`	`NodeB NodeA`
File Systems The file systems to be locally mounted by the node currently owning the resource group.	`/fs1`	`/fs2`
File systems to export The file system to NFS-export by the node currently owning resource group. The file system is a subset of the file system listed above.	`/fs1`	`/fs2`
File Systems to NFS mount The file systems/directories to be NFS-mounted by all nodes in the resource group. The first value is NFS mount point. The second value is local mount point.	`/mnt1;/fs1`	`/mnt2;/fs2`
File Systems Mounted before IP Configured	`true`	`true`

In this scenario:

- NodeA locally mounts and exports `/fs1, then over-mounts on /mnt1`.
- NodeB NFS-mounts `/fs1,` on /mnt1 from NodeA.

Setting up a resource group like this ensures the expected default node-to-node NFS behavior described at the beginning of this section.

When NodeA fails, NodeB closes any open files in NodeA:`/fs1`, unmounts it, mounts it locally, and re-exports it to waiting clients.

After takeover, NodeB has:

- `/fs2` locally mounted
- `/fs2` NFS-exported
- `/fs1` locally mounted
- `/fs1` NFS-exported
- `service1:/fs1` NFS mounted over `/mnt1`
- `service2:/fs2` NFS mounted over `/mnt2b`

On reintegration, `/fs1` is passed back to NodeA, locally mounted and exported. NodeB mounts it over NFS again.

NFS Cross-Mounting and IP Labels

To enable NFS cross-mounting, each cluster node may act as an NFS client. Each of these nodes must have a valid route to the service IP label of the NFS server node. That is, to enable NFS cross-mounting, an IP label must exist on the client nodes, and this IP label must be configured on the same subnet as the service IP label of the NFS server node.

If the NFS client nodes have service IP labels on the same network, this is *not* an issue. However, in certain cluster configurations, you need to create a valid route.

The following sections describe these cluster configurations and also include two ways to configure valid routes.

Cluster Configurations That Require Creating a Valid Route

The following cluster configuration may *not* have a route to the IP label on the NFS server node:

If heartbeating over IP Aliases is *not* configured, non-service interfaces must be on a different subnet than service interfaces. This creates a situation where the NFS client nodes may *not* have an interface configured on the subnet used to NFS export the file systems.

For non-concurrent resource groups with IPAT via IP Aliases to support NFS cross-mounting, you must create a route between the NFS client nodes and the node that is exporting the file systems. The following section provides options for creating the valid route.

Ways to Create a Route to the NFS Server

The easiest way to ensure access to the NFS server is to have an IP label on the client node that is on the same subnet as the service IP label of the NFS server node.

To create a valid route between the NFS client node and the node that is exporting the file system, you can configure either of the following:

- A separate NIC with an IP label configured on the service IP network and subnet

or

- A persistent node IP label on the service IP network and subnet.

Note: If the client node has a non-service IP label on the service IP network, configuring heartbeat over IP aliases allows the non-service IP label to be on the same subnet as the service IP label. See section Heartbeating over IP Aliases in Chapter 3: Planning Cluster Network Connectivity.

Be aware that these solutions do *not* provide automatic root permissions to the file systems because of the export options for NFS file systems that are set in HACMP by default.

To enable root level access to NFS mounted file systems on the client node, add all of the node's IP labels or addresses to the `root =` option in the cluster exports file: **/usr/es/sbin/cluster/etc/exports**. You can do this on one node, synchronizing the cluster resources propagates this information to the other cluster nodes. For more information on the **/usr/es/sbin/cluster/etc/exports** file, see the section NFS Exporting File Systems and Directories.

Resource Group Takeover with Cross-Mounted NFS File Systems

This section describes how to set up non-concurrent resource groups with cross-mounted NFS file systems so that NFS file systems are handled correctly during takeover and reintegration. In addition, non-concurrent resource groups support automatic NFS mounting across servers during fallover.

Setting Up NFS Mount Point Different from Local Mount Point

HACMP handles NFS mounting in non-concurrent resource groups as follows:

- The node that currently owns the resource group mounts the file system over the file system's local mount point, and this node NFS exports the file system.
- All the nodes in the resource group (including the current owner of the group) NFS mount the file system over a different mount point.

  Therefore, the owner of the group has the file system mounted twice—once as a local mount and once as an NFS mount.

Note: The NFS mount point must be outside the directory tree of the local mount point.

Since IPAT is used in resource groups that have NFS mounted file systems, the nodes will *not* unmount and remount NFS file systems during a fallover. When the resource group falls over to a new node, the acquiring node locally mounts the file system and NFS exports it. (The NFS mounted file system is temporarily unavailable to cluster nodes during fallover.) As soon as the new node acquires the IPAT label, access to the NFS file system is restored.

All applications must reference the file system through the NFS mount. If the applications used must always reference the file system by the same mount point name, you can change the mount point for the local file system mount (for example, change it to mount point_local and use the previous local mount point as the new NFS mount point).

Default NFS Mount Options for HACMP

The default options used by HACMP when performing NFS mounts are `hard`, `intr`.

To set `soft` mounts or any other options on the NFS mounts:

1. Enter `smit mknfsmnt`
2. In the **MOUNT now, add entry to /etc/filesystems or both?** field, select the **file systems** option
3. In the **/etc/filesystems entry will mount the directory on system RESTART** field, accept the default value of **no**.

This procedure adds the options you have chosen to the **/etc/filesystems** entry created. The HACMP scripts then read this entry to pick up any options you may have selected.

Creating and Configuring NFS Mount Points on Clients

An NFS mount point is required to mount a file system via NFS. In a non-concurrent resource group all the nodes in the resource group NFS mount the file system. You create an NFS mount point on each node in the resource group. The NFS mount point must be outside the directory tree of the local mount point.

Once the NFS mount point is created on all nodes in the resource group, configure the **NFS Filesystem to NFS Mount** attribute for the resource group.

To create NFS mount points and to configure the resource group for the NFS mount:

1. On each node in the resource group, create an NFS mount point by executing the following command:

   ```
   mkdir /mountpoint
   ```

   where *mountpoint* is the name of the local NFS mountpoint over which the remote file system is mounted.

2. In the **Change/Show Resources and Attributes for a Resource Group** SMIT panel, the **Filesystem to NFS Mount** field must specify both mount points.

   Specify the nfs mount point, then the local mount point, separating the two with a semicolon. For example:

   ```
   /nfspoint;/localpoint
   ```

   If there are more entries, separate them with a space:

   ```
   /nfspoint1;/local1 /nfspoint2;/local2
   ```

3. (*Optional*) If there are nested mount points, nest the NFS mount points in the same manner as the local mount points so that they match up properly.

4. (*Optional*) When cross-mounting NFS file systems, set the **Filesystems Mounted before IP Configured** field in SMIT for the resource group to **true**.

Creating and Configuring Stable Storage for NFSv4 Exports

Stable Storage is a file system space that is used to save the state information by the NFSv4 server. This is very crucial for maintaining NFSv4 client's state information to facilitate smooth and transparent fallover/fallback/move of the Resource group from one node to other.

Requirements of Stable Storage:

- Recommended size is 512 MB.
- It is recommended that you have a dedicated file system for the Stable Storage, but a sub-directory of an existing file system is acceptable.
- The file system/volume group where the Stable Storage resides should be part of the Resource Group.
- HACMP makes a best-effort attempt to verify that the path specified for stable storage belongs to a file system in the resource group; however, these checks are not able to take symbolic links into account since the file systems might not be mounted locally when the verification occurs. Avoid using symbolic links that can interfere with verification in HACMP.

- Ensure that the Stable Storage directory is empty (free of any prior data.) prior to adding it to the Resource Group.
- While Stable Storage must be stored in file systems managed by the resource group, it should not be stored in a directory that is NFS exported by the resource group.
- Configuration Assistant offers an **AUTO_SELECT** option. If you select this option, HACMP uses a VG from the list of VGs that are part of the given Resource Group. HACMP then creates a Logical Volume and a file system to use as the Stable Storage Location.

Completing the Shared LVM Components Worksheets

After you identify the physical and logical storage components for your cluster, complete all of the appropriate worksheets from the following list:

- Non-Shared Volume Group Worksheet
- Shared Volume Group/File System Worksheet
- NFS-Exported File System/Directory Worksheet.

Appendix A: Planning Worksheets contains the worksheets referenced in the following procedures.

Refer to the completed worksheets when you define the shared LVM components following the instructions in the chapter on Defining Shared LVM Components in the *Installation Guide*, and the cluster resource configuration following the instructions in chapters 2–4 of the *Administration Guide*.

Planning for LVM Components

Consider the following guidelines as you plan shared LVM components:

- In general, planning for logical volumes concerns the availability of your data. However, creating logical volume copies is *not* a substitute for regularly scheduled backups. Backups protect against loss of data regardless of cause. Logical volume copies protect against loss of data from physical access failure.
- All operating system files should reside in the root volume group (**rootvg**) and all user data should reside outside that group. This makes it more manageable to update or reinstall the operating system and to back up data.
- Volume groups that contain at least three physical volumes provide the maximum availability when implementing mirroring.
- If you plan to specify the **Use Forced Varyon of Volume Groups, if Necessary** attribute in SMIT for the volume groups, use the **super strict** disk allocation policy for mirrored physical volumes.
- When using copies, each physical volume containing a copy should get its power from a separate source. If one power source fails, separate power sources maintain the no “single point of failure” objective.
- Consider quorum issues when laying out a volume group. With quorum enabled, a two-disk volume group puts you at risk for losing quorum and data access. Either build three-disk volume groups or disable quorum.
- Plan for NFS mounted file systems and directories.

- Keep in mind the cluster configurations that you have designed. A node whose resources are *not* taken over should *not* own critical volume groups.

Completing the Non-Shared Volume Group Worksheet

For each node in the cluster, complete a Non-Shared Volume Group Worksheet for each volume group residing on a local (non-shared) disk:

1. Fill in the node name in the **Node Name** field.
2. Record the name of the volume group in the **Volume Group Name** field.
3. List the device names of the physical volumes comprising the volume group in the **Physical Volumes** field.

   In the remaining sections of the worksheet, enter the following information for each logical volume in the volume group. Use additional sheets if necessary.
4. Enter the name of the logical volume in the **Logical Volume Name** field.
5. If you are using LVM mirroring, indicate the number of logical partition copies (mirrors) in the **Number Of Copies Of Logical Partition** field. You can specify one or two copies (in addition to the original logical partition, for a total of three).
6. If you are using LVM mirroring, specify whether each copy will be on a separate physical volume in the **On Separate Physical Volumes?** field.
7. Record the full path mount point of the file system in the **Filesystem Mount Point** field.
8. Record the size of the file system in 512-byte blocks in the **Size** field.

Completing the Shared Volume Group and File System Worksheet

For each volume group that will reside on the shared disks, complete a separate Shared Volume Group/File System Worksheet for each volume group residing on a local (non-shared) disk.

To complete the Group and File System Worksheet:

1. Enter the name of each node in the cluster in the **Node Names** field. You determined the node names in Chapter 2: Initial Cluster Planning. Note that all nodes must participate in a concurrent resource group, if disk fencing is enabled. If disk fencing is *not* enabled, you can include a subset of nodes in the group.
2. Assign a name to the shared volume group and record it in the **Shared Volume Group Name** field. The name of the shared volume group must be unique within the cluster and distinct from the service IP label/address and resource group names; it should relate to the application it serves, as well as to any corresponding device, such as `websphere_service_address`.
3. Leave the **Major Number** field blank for now. You will enter a value in this field when you address NFS issues in the following Chapter 6: Planning Resource Groups.
4. Record the name of the log logical volume (**jfslog** or **jfs2log**) in the **Log Logical Volume Name** field.
5. Pencil-in the planned physical volumes in the **Physical Volumes** field. You will enter exact values for this field after you have installed the disks following the instructions in the chapter on Configuring Installed Hardware in the *Installation Guide*.

Physical volumes are known in the AIX operating system by sequential **hdisk** numbers assigned when the system boots. For example, */dev/hdisk0* identifies the first physical volume in the system, */dev/hdisk1* identifies the second physical volume in the system, and so on.

When sharing a disk in an HACMP cluster, the nodes sharing the disk each assign an **hdisk** number to that disk. These **hdisk** numbers may *not* match, but refer to the same physical volume. For example, each node may have a different number of internal disks, or the disks may have changed since AIX was installed.

The HACMP software does *not* require that the **hdisk** numbers match across nodes (although your system is easier to manage if they do). In situations where the **hdisk** numbers must differ, be sure that you understand each node's view of the shared disks. Draw a diagram that indicates the **hdisk** numbers that each node assigns to the shared disks and record these numbers on the appropriate volume group worksheets in Appendix A: Planning Worksheets. When in doubt, use the hdisk's PVID to identify it on a shared bus.

In the remaining sections of the worksheet, enter the following information for each logical volume in the volume group. Use additional sheets as necessary.

6. Assign a name to the logical volume and record it in the **Logical Volume Name** field.

   A shared logical volume must have a unique name within an HACMP cluster. By default, AIX assigns a name to any logical volume that is created as part of a journaled file system (for example, *lv01*). If you rely on the system generated logical volume name, this name could cause the import to fail when you attempt to import the volume group containing the logical volume into another node's ODM structure, especially if that volume group already exists. The chapter on Defining Shared LVM Components in the *Installation Guide* describes how to change the name of a logical volume.

7. If you are using LVM mirroring, indicate the number of logical partition copies (mirrors) in the **Number Of Copies of Logical Partition** field. You can specify that you want one or two copies (in addition to the original logical partition, for a total of three).

8. If you are using LVM mirroring, specify whether each copy will be on a separate physical volume in the **On Separate Physical Volumes?** field. If you are planning to use a forced varyon option for the volume groups, make sure that each copy will be mirrored on a separate physical volume.

9. Record the full-path mount point of the file system in the **File System Mount Point** field.

10. Record the size of the file system in 512-byte blocks in the **Size** field.

11. Record whether this volume group will have cross-site LVM mirroring enabled. When a volume group is enabled for cross-site LVM mirroring, cluster verification ensures that the volume group and logical volume structure is consistent and there is at least one mirror of each logical volume at each site.

    The volume group must also be configured as a resource in a resource group. Cross-site LVM mirroring supports two-site clusters where LVM mirroring through a Storage Area Network (SAN) replicates data between disk subsystems at geographically separated sites.

Completing the NFS-Exported File System Worksheet

Print the NFS-Exported File System or Directory Worksheet (Non-Concurrent Access) from Appendix A: Planning Worksheets, and fill it out using the information in this section. Print one copy for each application you want to keep highly available in the cluster.

For file systems or directories to be NFS-exported from a node, complete an NFS-Exported File System or Directory Worksheet. The information you provide will be used to update the **/usr/es/sbin/cluster/etc/exports** file.

To complete an NFS-Exported File System or Directory Worksheet:

1. Record the name of the resource group from which the file systems or directories will be NFS-exported in the **Resource Group** field.
2. In the **Network for NFS Mount** field, record the preferred network to NFS mount the file systems or directories.
3. In the **File System Mounted Before IP Configured** field, specify *true* if you want the takeover of file systems to occur before the takeover of IP address(es). Specify *false* for the IP address(es) to be taken over first.
4. Record the full pathname of the file system or directory to be exported in the **Exported Directory (NFSv2/3)** field.
5. Record the full pathname of the file system or directory to be exported in the **Exported Directory (NFSv4)** field.
6. (*Optional*) Record the export options you want to assign the directories, file systems, or both to be NFS-exported. See the **exports** man page for a full list of export options.
7. Repeat steps 4 and 5 for each file system or directory to be exported.
8. If the **Exported Directory (NFSv4)** is not empty, record the Stable Storage path.

Completing Concurrent Access Worksheets

Complete your concurrent access worksheets as explained in the following sections:

- Completing the Non-Shared Volume Group Worksheet (Concurrent Access)
- Completing the Shared Volume Group Worksheet (Concurrent Access).

Completing the Non-Shared Volume Group Worksheet (Concurrent Access)

For each node, complete a Non-Shared Volume Group Worksheet (Concurrent Access) for each volume group that resides on a local (non-shared) disk:

1. Enter the node name in the **Node Name** field.
2. Record the name of the volume group in the **Volume Group Name** field.
3. Enter the name of the logical volume in the **Logical Volume Name** field.
4. List the device names of the physical volumes that comprise the volume group in the **Physical Volumes** field.

   In the remaining sections of the worksheet, enter the following information for each logical volume in the volume group. Use additional sheets if necessary.
5. Enter the name of the logical volume in the **Logical Volume Name** field.
6. If you are using LVM mirroring, indicate the number of logical partition copies (mirrors) in the **Number Of Copies Of Logical Partition** field. You can specify one or two copies (in addition to the original logical volume, for a total of three).

7. If you are using LVM mirroring, specify whether each copy will be on a separate physical volume in the **On Separate Physical Volumes?** field. Specifying this option is especially important if you plan to force a varyon of volume groups, if a normal varyon operation fails due to a lack of quorum.
8. Record the full path mount point of the file system in the **File System Mount Point** field.
9. Record the size of the file system in 512-byte blocks in the **Size** field.

Completing the Shared Volume Group Worksheet (Concurrent Access)

Complete a separate Shared Volume Group and File System Worksheet for each volume group that will reside on the shared disks.

If you plan to create concurrent volume groups on SSA disk subsystem, assign unique non-zero node numbers with ssar on each cluster node.

If you specify the use of SSA disk fencing in your concurrent resource group, HACMP verifies that all nodes are included in the resource group and assigns the node numbers when you synchronize the resources.

If you do *not* specify the use of SSA disk fencing in your concurrent resource group, assign the node numbers with the following command:

```
chdev -l ssar -a node_number=x
```

where *x* is the number to assign to that node. Then reboot the system.

To complete a Shared Volume Group and File System Worksheet (Concurrent Access):

1. Enter the name of each node in the cluster in the **Node Names** field.
2. Record the name of the shared volume group in the **Shared Volume Group Name** field.
3. Pencil in the planned physical volumes in the **Physical Volumes** field. You will enter exact values for this field after you have installed the disks following the instructions in the chapter Installing HACMP on Server Nodes in the *Installation Guide*.

   In the remaining sections of the worksheet, enter the following information for each logical volume in the volume group. Use additional sheets as necessary.
4. Enter the name of the logical volume in the **Logical Volume Name** field.
5. Identify the number of logical partition copies (mirrors) in the **Number Of Copies Of Logical Partition** field. You can specify one or two copies (in addition to the original logical partition, for a total of three).
6. Specify whether each copy will be on a separate physical volume in the **On Separate Physical Volumes?** field. Specifying this option is especially important if you plan to force a varyon of volume groups, if a normal varyon operation fails due to a lack of quorum.

Adding LVM Information to the Cluster Diagram

Add the LVM information to the cluster diagram, including volume group and logical volume definitions. Include the site information if you are using cross-site LVM mirroring.

Where You Go from Here

You have now planned the shared LVM components for your cluster. Use this information when you define the volume groups, logical volumes, and file systems during the install.

In the next step of the planning process, you address issues relating to planning for your resource groups. Chapter 6: Planning Resource Groups describes this step of the planning process.

Chapter 6: Planning Resource Groups

This chapter describes how to plan resource groups within an HACMP cluster. This chapter contains the following sections:

- Prerequisites
- Overview
- General Rules for Resources and Resource Groups
- Two Types of Resource Groups: Concurrent and Non-Concurrent
- Resource Group Policies for Startup, Fallover and Fallback
- Resource Group Attributes
- Moving Resource Groups to Another Node
- Planning for Replicated Resources
- Planning Cluster Networks and Resource Groups
- Planning Parallel or Serial Order for Processing Resource Groups
- Planning Resource Groups in Clusters with Sites
- Planning for Workload Manager
- Completing the Resource Group Worksheet
- Where You Go from Here.

Prerequisites

By now, you should have completed the planning steps in the previous chapters:

- Chapter 2: Initial Cluster Planning
- Chapter 3: Planning Cluster Network Connectivity
- Chapter 4: Planning Shared Disk and Tape Devices
- Chapter 5: Planning Shared LVM Components.

Overview

HACMP organizes resources into resource groups. Each resource group is handled as a unit that contains shared resources such as IP labels, applications, file systems and volume groups. You define the policies for each resource group that define when and how it will be acquired or released.

In Chapter 2: Initial Cluster Planning, you made preliminary choices about the resource group policies and the takeover priority for each node in the resource group nodelists. In this chapter you do the following:

- Identify the individual resources that constitute each resource group.

- For each resource group, identify which type of group it is: concurrent or non-concurrent.
- Define the participating nodelist for the resource groups. The nodelist consists of the nodes assigned to participate in the takeover of a given resource group.
- Identify the resource group startup, fallover, and fallback policy.
- Identify applications and their resource groups for which you want to set up location dependencies, parent/child dependencies, or both.
- Identify the inter-site management policies of the resource groups. Is the group site-aware, are there replicated resources to consider.
- Identify other attributes and runtime policies to refine resource group behavior.

Note: For information about how resource groups policies and attributes from versions of HACMP prior to 5.2 are mapped to the resource group policies in the current version, see the chapter on Upgrading an HACMP Cluster in the *Installation Guide*.

The following definitions are used in this section:

- *Participating nodelist*. (A list of nodes that can host a particular resource group, as defined in the **Participating Node Names** for a resource group in SMIT). Be aware that the combination of the different resource group policies and the current cluster conditions also affect the resource group placement on the nodes in the cluster.
- *Home node (or the highest priority node for this resource group)*. The first node that is listed in the participating nodelist for any non-concurrent resource groups.

HACMP resource groups support NFS file systems. For information about using resource groups with NFS, see the section NFS Cross-Mounting and IP Labels in Chapter 5: Planning Shared LVM Components.

General Rules for Resources and Resource Groups

The following rules and restrictions apply to resources and resource groups:

- In order for HACMP to keep a cluster resource highly available, it must be part of a resource group. If you want a resource to be kept separate, define a group for that resource alone. A resource group may have one or more resources defined.
- A resource may *not* be included in more than one resource group.
- The components of a resource group must be unique. Put the application along with the resources it requires in the same resource group.
- The service IP labels, volume groups and resource group names must be both unique within the cluster and distinct from each other. It is recommended that he name of a resource should relate to the application it serves, as well as to any corresponding device, such as `websphere_service_address`.
- If you include the same node in participating nodelists for more than one resource group, make sure that the node has the memory, network interfaces, etc. necessary to manage all resource groups simultaneously.

Two Types of Resource Groups: Concurrent and Non-Concurrent

To categorize and describe resource group behavior, we first divide the resource groups into two types: concurrent and non-concurrent:

Concurrent Resource Groups

A concurrent resource group may be online on multiple nodes. All nodes in the nodelist of the resource group acquire that resource group when they join the cluster. There are no priorities among nodes. Concurrent resource groups are supported in clusters with up to 32 nodes.

The only resources included in a concurrent resource group are volume groups with raw logical volumes, raw disks, and application servers that use the disks. The device on which these logical storage entities are defined must support concurrent access.

Concurrent resource groups have the startup policy Online on All Available Nodes and do not fallover or fallback from one node to another.

Non-Concurrent Resource Groups

Non-concurrent resource groups may not be online on multiple nodes. You can define a variety of startup, fallover, and fallback policies for these resource groups.

You can fine tune the non-concurrent resource group behavior for node preferences during a node startup, resource group fallover to another node in the case of a node failure, or when the resource group falls back to the reintegrating node. See Resource Group Policies for Startup, Fallover and Fallback for more information.

Resource Group Policies for Startup, Fallover and Fallback

Resource group behaviors are separated into three kinds of node policies:

- *Startup* policy defines on which node the resource group will be activated when a node joins the cluster and the resource group is *not* active on any node.
- *Fallover* policy defines to which node the resource group will fall over when the resource group must leave the node where it is currently online due to a failure condition (or if you stop the cluster services on a node using the fallover option).
- *Fallback* policy defines to which node the resource group will fall back when a node joins and the resource group is already active on another node.

HACMP allows you to configure only *valid* combinations of startup, fallover, and fallback behaviors for resource groups. The following table summarizes the basic startup, fallover, and fallback behaviors you can configure for resource groups in HACMP 5.4.1:

Startup Behavior	Fallover Behavior	Fallback Behavior
Online only on home node (first node in the nodelist)	• Fallover to next priority node in the list or • Fallover using Dynamic Node Priority	• Never fall back or • Fall back to higher priority node in the list
Online using node distribution policy	• Fallover to next priority node in the list or • Fallover using Dynamic Node Priority	Never fall back
Online on first available node	Any of these: • Fallover to next priority node in the list • Fallover using Dynamic Node Priority	• Never fall back or • Fall back to higher priority node in the list
Online on all available nodes	Bring offline (on error node only)	Never fall back

In addition to the node policies described in the previous table, other issues may determine the resource groups that a node acquires. For more information about the event processing logic for resource groups, see Chapter 7: Planning for Cluster Events.

Resource Group Attributes

This section provides an overview of the following resource group attributes that you can use to fine tune the startup, fallover, and fallback policies of resource groups:

How Resource Group Attributes Relate to Startup, Fallover, and Fallback. Each attribute affects resource group startup, resource group fallover to another node in the case of a node failure, or resource group fallback to the reintegrating node

- Settling Time for Startup. You can modify a resource group's startup behavior by specifying a settling time for a resource group that is currently offline. With a settling time specified, you can avoid having a resource group activated on the first available node; a higher priority node for the resource group may join the cluster during this time period.
- Node Distribution Policy. You can configure a startup behavior of a resource group to use the node distribution policy during startup. This policy ensures that only one resource group with this policy enabled is acquired on a node during startup.

- Dynamic Node Priority Policy. You can configure a resource group's fallover behavior to use dynamic node priority. It allows you to use a predefined RSCT resource variable such as "lowest CPU load" to select the takeover node.
- Delayed Fallback Timer. You can configure a resource group's fallback behavior 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.
- Parent and Child Dependent Resource Groups. Related applications in different resource groups are configured to be processed in logical order.
- Resource Group Location Dependencies. Certain applications in different resource groups stay online together on a node or on a site, or stay online on different nodes.

How Resource Group Attributes Relate to Startup, Fallover, and Fallback

The following table summarizes which resource group startup, fallover or fallback policies are affected by a given attribute or run-time policy..

Attribute	**Startup Policy**	**Fallover Policy**	**Fallback Policy**
Settling Time	✓		
Node Distribution Policy	✓		
Dynamic Node Priority		✓	
Delayed Fallback Timer			✓
Resource Groups Parent/Child Dependency	✓	✓	✓
Resource Groups Location Dependency	✓	✓	✓

See Parent and Child Dependent Resource Groups and Resource Group Location Dependencies for guidelines.

Settling Time for Startup

Settling Time lets the Cluster Manager wait for a specified amount of time before activating a resource group. Use this attribute to ensure that a resource group does *not* bounce among nodes, as nodes with increasing priority for the resource group are brought online.

If the node that is starting is the first node in the nodelist for this resource group, the settling time period is skipped and HACMP immediately attempts to acquire the resource group on this node.

The settling time has the following characteristics:

- Affects only those resource groups that are currently offline, and for which you have specified the startup policy to be Online on First Available Node. You configure one settling time for *all* such resource groups.

- Activates when the first node that can acquire the resource group joins the cluster, unless this is the first node in the nodelist (then the settling time is ignored and the group is acquired).

  If the first node that joins the cluster and can potentially acquire the resource group fails, this condition can either cancel the settling time, or reset it.
- Delays the activation of the group during **node_up** events, in case higher priority nodes join the cluster.
- If a settling time period is specified for a resource group and a resource group is currently in the ERROR state, the Cluster Manager waits for the settling time period before attempting to bring the resource group online during a **node_up** event.

Node Reintegration with Settling Time Configured

In general, when a node joins the cluster, it can acquire resource groups. The following list describes the role of the settling time in this process:

- If the node is the highest priority node for a specific resource group, the node immediately acquires that resource group and the settling time is ignored. (*This is only one circumstance under which HACMP ignores the setting*).
- If the node is able to acquire some resource groups, but is *not* the highest priority node for those groups, the resource groups do *not* get acquired on that node. Instead, they wait during the settling time interval to see whether a higher priority node joins the cluster.

  When the settling time interval expires, HACMP moves the resource group to the highest priority node which is currently available and which can take the resource group. If HACMP does *not* find appropriate nodes, the resource group remains offline.

See the *Administration Guide* for details on configuring settling time.

Node Distribution Policy

You can use *node distribution policy* for cluster startup, to ensure that HACMP activates only one resource group with this policy enabled on each node. This policy helps you distribute your CPU-intensive applications on different nodes.

The facts about node distribution policy are:

- If you plan to use a single adapter network that will be configured with IPAT via Replacement, the startup policy for your resource group should be set to Online using Distribution Policy.
- If two resource groups with this policy enabled are offline at the time when a particular node joins, only one of the two resource groups is acquired on a node. HACMP gives preference to the resource group that has fewer nodes in the nodelist and then sorts the list of resource groups alphabetically.
- If one of the resource groups is a parent resource group (has a child resource group), HACMP gives preference to the parent resource group and it is activated on a node.
- To ensure that your resource groups are distributed *not* only at startup but also for recovery events (fallover and fallback), use location dependencies. See Resource Group Dependencies.

Dynamic Node Priority Policy

Setting a dynamic node priority policy allows you to use a predefined RMC resource variable such as “lowest CPU load” to select the takeover node. 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 RMC resource variable. You can set different policies for different groups or the same policy for several groups.

If you decide to define dynamic node priority policies using RMC resource variables to determine the fallover node for a resource group, consider the following points:

- Dynamic node priority policy is most useful in a cluster where all the nodes have equal processing power and memory
- Dynamic node priority policy is irrelevant for clusters of fewer than three nodes
- Dynamic node priority policy is irrelevant for concurrent resource groups.

Remember that selecting a takeover node also depends on such conditions as the availability of a network interface on that node.

Delayed Fallback Timer

You can use a *Delayed Fallback Timer* to set the time for a resource group to fall back to a higher priority node. You can configure the fallback behavior for a resource group to occur at one of a predefined recurring time (daily, weekly, monthly, specific date).

The delayed fallback timer has the following characteristics:

- Specifies the time when a resource group that is online and residing on a non-home or low priority node falls back its home node or a higher priority node
- Affects the movement of the resource group to another node. For example, if you move the non-concurrent resource group (that has a fallback timer attribute) to another node using the Resource Group Management utility (**clRGmove**), the group stays on the destination node (unless you reboot the cluster, which is rarely done). If the destination node goes down and then reintegrates, the resource group also falls back to this node at the specified time.

Node Reintegration with a Delayed Fallback Timer Set

The resource group does *not* fallback to its higher priority node immediately under the following condition:

- You have configured a delayed fallback timer for a resource group, and
- A higher priority node joins the cluster.

At the time specified in the Delayed Fallback Timer attribute, one of two scenarios takes place:

- *A higher priority node is found.* If a higher priority node is available for the resource group, HACMP attempts to move the resource group to this node when the fallback timer expires. If the acquisition is successful, the resource group is acquired on that node.

  However, if the acquisition of the resource group on the node fails, HACMP attempts to move the resource group to the next higher priority node in the group nodelist, and so on. If the acquisition of the resource group on the last node that is available fails, the resource group goes into an error state. You must take action to fix the error and bring such a resource group back online. For more information, see the chapter on Managing Resource Groups in a Cluster in the *Administration Guide*.

- *A higher priority node is not found.* If there are no higher priority nodes available for a resource group, the resource group remains online on the same node until the fallback timer expires again. for example, if a daily fallback timer expires at 11:00 p.m. and there are no higher priority nodes available for the resource group to fallback on, the fallback timer reoccurs the next night at 11:00 p.m.

  A fallback timer that is set to a specific date does *not* reoccur.

Resource Group Dependencies

HACMP offers a wide variety of configurations where you can specify the relationships between resource groups that you want to maintain at startup, fallover, and fallback. You can configure:

- Parent and Child Dependent Resource Groups. Related applications and other resources in different resource groups are configured to be processed in the proper order.
- Resource Group Location Dependencies. Certain applications in different resource groups stay online together on a node or on a site, or stay online on different nodes.

Keep the following points in mind when planning how to configure these dependencies:

- Although by default all resource groups are processed in parallel, HACMP processes dependent resource groups according to the order dictated by the dependency, and *not necessarily in parallel*. Resource group dependencies are honored cluster-wide and override any customization for serial order of processing of any resource groups included in the dependency. For more information, see Dependent Resource Groups and Parallel or Serial Order.
- Dependencies between resource groups offer a predictable and reliable way of building clusters with multi-tiered applications.

The following limitations apply to configurations that combine dependencies. Verification will fail if you do *not* follow these guidelines:

- Only *one* resource group can belong to an Online on Same Node dependency set and an Online On Different Nodes dependency set at the same time
- Only resource groups with the same Priority within an Online on Different Nodes dependency set can participate in an Online on Same Site dependency set.

For information on configuring dependent resource groups see the *Administration Guide*.

Parent and Child Dependent Resource Groups

Configuring a resource group dependency allows for better 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. For more information, see Planning Considerations for Multi-Tiered Applications in Chapter 2: Initial Cluster Planning.

The following example illustrates the parent/child dependency behavior:

- If resource group A depends on resource group B, resource group B must be brought online before resource group A is acquired on any node in the cluster. Note that resource group A is defined as a *child resource group*, and resource group B as a *parent resource group*.

- If child resource group A depends on parent resource group B, during a node startup or node reintegration, child resource group A cannot come online before parent resource group B gets online. If parent resource group B is taken offline, the child resource group A is taken offline first, since it depends on resource group B.

Business configurations that use multi-tiered applications can utilize parent/child dependent resource groups. For example, the database must be online before the application server. In this case, if the database 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. For more information about examples of multi-tiered applications, see the *Concepts and Facilities Guide*.

If a child resource group contains an application that depends on resources in the parent resource group and the parent resource group falls over to another node, the child resource group is temporarily stopped and automatically restarted. Similarly, if the child resource group is concurrent, HACMP takes it offline temporarily on all nodes, and brings it back online on all available nodes. If the fallover of the parent resource group is *not* successful, both the parent and the child resource groups go into an ERROR state.

Consider the following when planning for parent/child dependent resource groups:

- Plan applications you need to keep highly available and consider whether your business environment requires one application to be *running* before another application can be started.
- Ensure that those applications that require sequencing are included in different resource groups. This way, you can establish dependencies between these resource groups.
- Plan for application monitors for each application that you are planning to include in a child or parent resource group. For an application in a parent resource group, configure a monitor in the **monitoring startup** mode.

  To minimize the chance of data loss during the application stop and restart process, customize your application server scripts to ensure that any uncommitted data is stored to a shared disk temporarily during the application stop process and read back to the application during the application restart process. It is important to use a shared disk as the application may be restarted on a node other than the one on which it was stopped.

Resource Group Location Dependencies

If failures do occur over the course of time, HACMP distributes resource groups so that they remain available, but *not* necessarily on the nodes you originally specified, unless they have the same home node and the same fallover and fallback policies.

Resource group location dependency offers you an explicit way to specify that certain resource groups will always be online on the same node, or that certain resource groups will always be online on different nodes. You can combine these location policies with parent/child dependencies, to have all child resource groups online on the same node while the parent is online on a different node; or, to have all child resource groups be online on different nodes for better performance.

If you have replicated resources, you can combine resource groups into a site dependency to keep them online at the same site. For more information, see the section Special Considerations for Using Sites in this chapter.

For detailed examples of how resource group location dependencies affect the handling of resource groups, see Appendix B in the *Administration Guide*.

HACMP supports three types of resource group location dependencies between resource groups:

- Online on Same Node

  The following rules and restrictions apply to the Online On Same Node Dependency set of resource groups. Verification will fail if you do *not* follow these guidelines:

  - All resource groups configured as part of a given Same Node dependency set must have the same nodelist (the same nodes in the same order).
  - All non-concurrent resource groups in the Same Node dependency set must have the same Startup/Fallover/Fallback policies.
  - Online Using Node Distribution Policy is *not* allowed for Startup.
  - If a Dynamic Node Priority Policy is configured as Fallover Policy, then all resource groups in the set must have the same policy.
  - If one resource group has a fallback timer configured, it applies to the set of resource groups. All resource groups in the set must have the same fallback time setting.
  - Both concurrent and non-concurrent resource groups can be included.
  - You can have more than one Same Node dependency set in the cluster.
  - HACMP enforces the condition that all resource groups in the Same Node dependency set that are active (ONLINE) are required to be ONLINE on the same node. Some resource groups in the set can be OFFLINE or in the ERROR state.
  - If one or more resource groups in the Same Node dependency set fail, HACMP tries to place all resource groups in the set on the node that can host all resource groups that are currently ONLINE (the ones that are still active) plus one or more failed resource groups.
- Online on Same Site

  The following rules and restrictions are applicable to Online On Same Site Dependency set of resource group. Verification will fail if you do *not* follow these guidelines:

  - All resource groups in a Same Site dependency set must have the same Inter-Site Management Policy but may have different Startup/Fallover/Fallback Policies. If fallback timers are used, these must be identical for all resource groups in the set.
  - The fallback timer does *not* apply to moving a resource group across site boundaries.
  - All resource groups in the Same Site Dependency set must be configured so that the nodes that can own the resource groups are assigned to the same primary and secondary sites.
  - Online Using Node Distribution Policy Startup policy is supported.
  - Both concurrent and non-concurrent resource groups can be included.
  - You can have more than one Same Site dependency set in the cluster.
  - All resource groups in the Same Site dependency set that are active (ONLINE) are required to be ONLINE on the same site, even though some resource groups in the set may be OFFLINE or in the ERROR state.

  - If you add a resource group that is included in a Same Node dependency set to a Same Site Dependency set, then all the other resource groups in the Same Node Dependency set must be added to the Same Site dependency set.

  Also see the section Planning Resource Groups in Clusters with Sites.

- Online on Different Nodes

  When you configure resource groups in the Online On Different Nodes dependency set you assign priorities to each resource group in case there is contention for a given node at any point in time. You can assign High, Intermediate, and Low priority. Higher priority resource groups take precedence over lower priority groups at startup, fallover, and fallback.

  The following rules and restrictions apply to the Online On Different Nodes dependency set of resource groups. Verification will fail if you do *not* follow these guidelines:

  - Only one Online On Different Nodes dependency set is allowed per cluster.
  - Plan startup policies so that each resource group in the set will start up on a different node.
  - If a parent/child dependency is specified, then the child resource group cannot have a higher priority than its parent resource group.

  Once the cluster is running with these groups configured, be aware that:

  - If a resource group with High Priority is ONLINE on a node, then no other lower priority resource group in the Different Nodes dependency set can come ONLINE on that node.
  - If a resource group with a higher priority falls over or falls back to a given node, the resource group with the higher priority will come ONLINE and the Cluster Manager takes the lower priority resource group OFFLINE and moves it to another node if this is possible.
  - Resource groups with the same priority cannot come ONLINE (startup) on the same node. Priority of a resource group for a node *within the same Priority Level* is determined by the groups' alphabetical order in the set.
  - Resource groups with the same priority do *not* cause one another to be moved from the node after a fallover or fallback.

- Combination of Same Site with Node Location Dependencies.

  - You can have resource groups that belong to both an Online on Same Node and Online on Same Site policy. You can also have resource groups that belong to both Online on Same Site and Online on Different Nodes policy.

Moving Resource Groups to Another Node

When you tell HACMP to move the resource group to another node, you have the following options:

- Resource groups remain on the nodes to which they are moved.

  You can move resource groups that have the Never Fallback policy to another node. When you do so, you can tell HACMP to leave the resource group on the destination node until you decide to move the group again.

- When you move a resource group with **RG_move**, it will remain on the node to which it was moved either indefinitely (until you tell HACMP to move it to another node) or until you reboot the cluster.

  Note: If you do stop the cluster services, which rarely has to be done, and you wish to *permanently* change the resource group's nodelist and highest priority node, change the resource group's attributes and restart the cluster.

- If you take the resource group online or offline on any of the node(s), it will remain online or offline either until the next cluster reboot, or until you manually bring the group online elsewhere in the cluster.
- If your resource group has the Fallback to Highest Priority Node policy, the group falls back to its destination node, after you move it.

  For instance, if the group has node A configured as its highest priority node, and you move it to node B, then this group will remain on node B and will treat this node now as its highest priority node. You can always choose to move the group again to node A. When you use SMIT to do so, HACMP informs you if the "original" highest priority node (node A) is now available to host the group.

  You can keep track of all resource groups that were manually moved by using the **clRGinfo –p** command.

Using clRGmove to Move Resource Groups

You can use the command **clRGmove** to move a resource group to another node or to another site, or to take a resource group online or offline. The resource group remains on the node until the cluster reboot. You can run the **clRGmove** command via SMIT or from the command line.

If you use **clRGmove** with resource groups that have the fallback policy Never Fallback, the resource group remains on that node until you move it elsewhere.

Moving Parent/Child Dependent Resource Groups with clRGmove

The following rules apply to resource groups with a parent/child dependency:

- If the parent resource groups are offline due to your request made through **clRGmove**, HACMP rejects manual attempts to bring the child resource groups that depend on these resource groups online. The error message lists the parent resource groups that must be brought online first.
- If you have a parent and a child resource group online, and would like to move the parent resource group to another node or take it offline, HACMP prevents you from doing so before a child resource group is taken offline.

For more information, see the sections on moving resource groups in the chapter on Managing Resource Groups in a Cluster in the *Administration Guide.*

Moving Location Dependent Resource Groups with clRGmove

The following rules apply to resource groups with a location dependency:

- If you move a same-site dependent resource group to the other site, the entire set of resource groups in the dependency is moved to the other site.

- If you move a same-node dependent resource group to another node, the entire set of resource groups in the dependency is moved.
- You cannot move a resource group to any node that hosts a resource group that is online and part of a different-node dependency. You have to take the resource group that is included in the different node dependency offline on the selected node first.

Planning Cluster Networks and Resource Groups

You cannot mix IPAT via IP Aliases and IPAT via IP Replacement labels in the same resource group. This restriction is enforced during verification of cluster resources.

IPAT of any type does *not* apply to concurrent resource groups.

This section contains the following topics:

- Aliased Networks and Resource Groups
- IPAT via IP Replacement Networks and Resource Groups
- Planning Service IP Labels in Resource Groups.

Aliased Networks and Resource Groups

A resource group may include multiple service IP labels. When a resource group configured with IPAT via IP Aliases is moved on an aliased network, all service labels in the resource group are moved as aliases to an available network interface, according to the resource group management policies in HACMP.

For information on planning IPAT via IP Aliases, see IP Address Takeover via IP Aliases in Chapter 3: Planning Cluster Network Connectivity.

If you configure aliased networks in your cluster, see Appendix B: Resource Group Behavior during Cluster Events in the *Administration Guide* for information on how the service IP label is moved at cluster startup and during fallover.

IPAT via IP Replacement Networks and Resource Groups

All non-concurrent resource groups can have their service IP labels configured on IPAT via IP Replacement networks.

In addition, the following planning considerations apply:

In addition, the following planning considerations apply if you plan to configure resource groups with service labels on IPAT via IP Replacement networks:

- If you plan to use a single adapter network that will be configured with IPAT via Replacement, be aware that the startup policy for your resource group should be set to Online using Distribution Policy.
- For each IPAT via IP Replacement network, you can configure a node to serve as the highest priority node for only one non-concurrent resource group in which a service IP label is also configured.

Such resource groups should have a startup policy of either Online on First Available Node or Online on Home Node Only. These resource groups cannot have a startup policy of Online using Distribution Policy.

- Place service IP labels that are configured on the same IPAT via IP replacement network into *different* resource groups. HACMP issues an error if it finds more than one service IP label configured on the IPAT via IP replacement network.
- The resource group cannot contain more than one service IP label from the same IPAT via Replacement network.

Planning Service IP Labels in Resource Groups

The subnet requirements for boot and service IP labels/addresses managed by HACMP depend on the following:

- The method that you choose for IP label/address recovery: IPAT via IP Aliases, or IPAT via IP Replacement with or without Hardware Address Takeover (HWAT)
- The type of resource group in which the IP label is included.

Note: The boot IP label/address is the initial IP label/address that is used at boot time. The term “boot” is used in this section to explain the HACMP requirements for subnets on which different IP labels (boot and service) are configured.

The following requirements are placed on the cluster configurations in order for HACMP to monitor the health of network interfaces and to properly maintain subnet routing during fallover and fallback:

Configuration	**Subnet Requirements Placed on Service IP Labels**
Configuration includes: • Heartbeating over IP Aliases • IPAT • Any type of non-concurrent resource group	In this case HACMP places *no* subnet requirements on any boot or service IP labels/addresses. Boot and service addresses can coexist on the same subnet or on different subnets. Because HACMP automatically generates the proper addresses required for heartbeating, all other addresses are free of constraints. Heartbeating over IP aliases works with all types of IPAT: IPAT via IP Aliases, IPAT via IP Replacement, and IPAT via IP Replacement with HWAT. The resource group management policy places no additional restrictions on IP labels. All service labels are handled according to the group behaviors and are *not* affected by subnetting. Selecting IP aliasing for heartbeating provides the greatest flexibility for configuring boot and service IP addresses at the cost of reserving a unique IP address and subnet range specifically for sending heartbeats.
Configuration includes: • Heartbeating over IP interfaces • IPAT via IP Aliases • Any type of non-concurrent resource group	With IPAT via IP Aliases, HACMP has the following requirements regardless of the resource group type: • All boot interfaces on any node must be configured on different subnets. This is required for correct operation of heartbeating. (Otherwise, having multiple interfaces on the same subnet would produce multiple subnet routes. This prevents reliable heartbeating and failure detection.) • All service IP addresses must be configured on different subnets than any of the boot addresses. This requirement prevents any possibility of having multiple routes leading to the same subnet. • Multiple service IP labels can be configured on the same subnet, because HACMP sends heartbeats only across the boot IP addresses.

The following table continues to summarize subnet requirements for each resource group and network configuration:

Configuration	Subnet Requirements Placed on Service IP Labels
Configuration includes: • Heartbeating over IP interfaces • IPAT via IP Replacement • Resource group with the startup policy Online on Home Node Only, the fallover policy Fallover to Next Priority Node in the List, and the fallback policy Fallback to Highest Priority Node.	The highest priority node in the resource group must be the node that contains the boot IP address/label. Select the boot IP address so that it is on the same subnet as the service IP address to be used in the resource group. All other network interfaces on the highest priority node must be placed on subnet(s) that are different from that used by the boot/service IP addresses. HACMP verifies these requirements.
Configuration includes: • Heartbeating over IP interfaces • IPAT via IP Replacement • Resource group with the startup policies Online on First Available Node or Online Using Node Distribution Policy, the fallover policy Fallover to Next Priority Node in the List, and the fallback policy Never Fallback.	For all nodes, the network interfaces should be configured to use one subnet for boot/service IP addresses and another subnet for the standby. Also, if you are planning to use a single-adapter network that will be configured with IPAT via Replacement, be aware that the startup policy for your resource group should be set to Online using Distribution Policy. HACMP verifies these requirements.

Planning Parallel or Serial Order for Processing Resource Groups

Note: The option to configure a resource group for parallel or serial order of processing may be discontinued in the future. Configure resource group dependencies to ensure the proper order of processing instead of using this option.

By default, HACMP acquires and releases all individual resources configured in your cluster in parallel. However, you can specify a specific serial order according to which some or all of the *individual* resource groups should be acquired or released. In this case, during acquisition:

1. HACMP acquires the resource groups serially in the order that you specified in the list
2. HACMP acquires the remaining resource groups in parallel.

During release, the process is reversed:

1. HACMP releases the resource groups for which you *did not* define a specific serial order in parallel.
2. The remaining resource groups in the cluster are processed in the order that you specified for these resource groups in the list.

If you upgraded your cluster from a previous version of HACMP, for more information on which processing order is used in this case, see the chapter on Upgrading an HACMP Cluster.

Note: Even if you specify the order of resource group processing on a single node, the actual fallover of the resource groups may be triggered by different policies. Therefore, it is *not* guaranteed that resource groups are processed cluster-wide in the order specified because *the serial customized processing order of resource groups applies to their processing on a particular node only.*

When resource groups are processed in parallel, fewer cluster events occur in the cluster. In particular, events such as **node_up_local** or **get_disk_vg_fs** do *not* occur if resource groups are processed in parallel.

As a result, using parallel processing reduces the number of particular cluster events for which you can create customized pre- or post-event scripts. If you start using parallel processing for some of the resource groups in your configuration, be aware that your existing pre- or post-event scripts may *not* work for these resource groups. For more information on parallel processing of resource groups and using event scripts, see Chapter 7: Planning for Cluster Events.

Parallel and serial processing of resource groups is reflected in the event summaries in the **hacmp.out** file. For more information, see the section Understanding the hacmp.out Log File in Chapter 2: Using Cluster Log Files in the *Troubleshooting Guide*.

For more information about how to configure customized serial acquisition and release order of resource groups, see Configuring Processing Order for Resource Groups in Chapter 4: Configuring HACMP Resource Groups (Extended) in the *Administration Guide*.

Dependent Resource Groups and Parallel or Serial Order

Although by default HACMP processes resource groups in parallel, if you establish dependencies between some of the resource groups in the cluster, processing may take longer than it does for clusters without dependent resource groups as there may be more processing to do to handle one or more **rg_move** events

Upon acquisition, first the parent or higher priority resource groups are acquired, then the child resource groups are acquired. Upon release, the order is reversed. The remaining resource groups in the cluster (those that do *not* have dependencies themselves) are processed in parallel.

Also, if you specify serial order or processing and have dependent resource groups configured, make sure that the serial order does *not* contradict the dependency specified. The resource groups dependency overrides any serial order in the cluster.

Planning Resource Groups in Clusters with Sites

The combination of Inter-Site Management Policy and the node startup, fallover and fallback policies that you select determines the resource group startup, fallover, and fallback behavior.

Site support in HACMP allows a variety of resource group configurations.

Concurrent Resource Groups and Sites

You can use the following policies for concurrent resource groups:

Inter-Site Management Policy	Online on Both Sites Online on Either Site Prefer Primary Site Ignore
Startup Policy	Online on All Available Nodes
Fallover Policy	Bring Offline (on Error node only)
Fallback Policy	Never FallBack

Non-Concurrent Resource Groups and Sites

For non-concurrent resource groups, you can use the following policies:

Inter-Site Management Policy	Online on Either Site Prefer Primary Site Ignore
Startup Policy	Online on Home Node Online on First Available Node Online using Node Distribution Policy
Fallover Policy	Fallover to Next Priority Node (in the nodelist)
Fallback Policy	Fallback to higher priority node (in the nodelist) Never Fallback

General Resource Group Behavior in Clusters with Sites

Non-concurrent resource groups defined with an Inter-Site Management Policy of Prefer Primary Site or Online on Either Site have two instances when the cluster is running:

- A primary instance on a node at the primary site
- A secondary instance on a node at the secondary site.

The **clRGinfo** command shows these instances as:

- ONLINE
- ONLINE SECONDARY.

Concurrent resource groups (Online on All Nodes) with an Inter-Site Management Policy of Online on Both Sites have multiple ONLINE instances and no ONLINE SECONDARY instances when the cluster is running at both locations.

Concurrent resource groups with an Inter-Site Management Policy of Prefer Primary Site or Online on Either Site have primary instances on each node at the primary site and secondary instances on nodes at the secondary site.

Resource groups with replicated resources are processed in parallel on **node_up**, according to their site management and node startup policies, taking any dependencies into account. When resource groups fall over between sites, the secondary instances are acquired and brought ONLINE SECONDARY on the highest priority node available at the new backup site. (More than one secondary instance can be on the same node.) Then the primary instance of the resource group is acquired and brought ONLINE on the highest priority node available to host this resource group on the new "active" site. This order of processing ensures that the backup site is ready to receive backup data as soon as the primary instance starts.

If the secondary instance cannot be brought to ONLINE SECONDARY state for some reason, the primary instance will still be brought ONLINE, if possible.

Special Considerations for Using Sites

This section discusses considerations if your resource group has:

- A Startup Policy of Online using Node Distribution Policy
- Dependencies specified.

Dependent Resource Groups and Sites

You can specify a dependency between two or more resource groups that reside on nodes on different sites. In this case, if either parent or child moves to the other site, the dependent group moves also. If for some reason the parent group cannot be activated on the fallover site, the child resource group will remain inactive also.

Note that the dependency only applies to the state of the primary instance of the resource group. If the parent group's primary instance is OFFLINE, and the secondary instance is ONLINE SECONDARY on a node, the child group's primary instance will be OFFLINE.

In cases of resource group recovery, resource groups can fall over to node(s) on either site. The sequence for acquiring dependent resource groups is the same as for clusters without sites, with the parent resource group acquired first and the child resource group acquired after it. The release logic is reversed: the child resource group is released before a parent resource group is released.

Note that you *cannot* use a resource group node distribution policy for startup for resource groups with a *same node* dependency. You *can* use this policy for resource groups with a *same site* dependency.

As in clusters without sites, if you have sites defined, configure application monitoring for applications included in resource groups with dependencies.

Resource Group Behavior Examples in Clusters with Sites

The Fallback Policy applies to the ONLINE and ONLINE SECONDARY instances of the resource group.

The Inter-Site Management Policy for a resource group determines the fallback behavior of the ONLINE instances of the resource groups between sites, thus governing the location of the secondary instance.

The ONLINE SECONDARY instance is located at the site that does *not* have the ONLINE instance. The following table shows the expected behavior of resource groups during site events according to the Startup and Inter-Site Management policies:

NodeStartup Policy (applies within site)	Inter-Site Management Policy	Startup/Fallover/Fallback Behavior
Online on Home Node Only	Prefer Primary Site	**Cluster Startup** *Primary site:* The home node acquires the resource group in the ONLINE state. Non-home nodes leave the resource group OFFLINE. *Secondary site*: The first node that joins on this site acquires the resource group in ONLINE SECONDARY state. **Inter-site Fallover** The ONLINE instance falls between sites when no nodes at the local site can acquire the resource group. The secondary instance moves to the other site and is brought ONLINE SECONDARY on the highest priority node available, if possible. **Inter-site Fallback** The ONLINE instance falls back to the primary site when a node from the primary site joins. The secondary instance moves to the other site and is brought ONLINE SECONDARY on the highest priority node available, if possible.

NodeStartup Policy (applies within site)	**Inter-Site Management Policy**	**Startup/Fallover/Fallback Behavior**
Online on First Available Node *or* Online Using Node Distribution Policy	Prefer Primary Site	**Cluster Startup** *Primary site:* The node that joins first from the primary site (and meets the criteria) acquires the resource group in the ONLINE state. The resource group is OFFLINE on all other nodes at the primary site. Note that the node distribution policy applies only to the primary instance of the resource group. *Secondary site*: The first node to join the cluster in this site acquires all secondary instances of resource groups with this startup policy in ONLINE_SECONDARY state (no distribution). **Inter-site Fallover** The ONLINE instance falls between sites when no nodes at the local site can acquire the resource group. The secondary instance moves to the other site and is brought ONLINE SECONDARY on the highest priority node available, if possible. **Inter-site Fallback** The ONLINE instance falls back to primary site when a node on the primary site joins. The secondary instance moves to the other site and is brought ONLINE SECONDARY on the highest priority node available, if possible.

NodeStartup Policy (applies within site)	Inter-Site Management Policy	Startup/Fallover/Fallback Behavior
Online on all Available Nodes	Prefer Primary Site	**Cluster Startup** *Primary site:* All nodes acquire the resource group in the ONLINE state. *Secondary site:* All nodes acquire the resource group in ONLINE_SECONDARY state. **Inter-site Fallover** The ONLINE instances fall between sites when all nodes at the local site go OFFLINE or fail to start the resource group. The secondary instances move to the other site and are brought ONLINE SECONDARY where possible. **Inter-site Fallback** ONLINE instances fall back to primary site when a node on the primary site rejoins. Nodes at the secondary site acquire the resource group in the ONLINE_SECONDARY state.
Online on Home Node Only	Online on Either Site	**Cluster Startup** The home node that joins the cluster (from either site) acquires the resource group in the ONLINE state. Non-home nodes leave the resource group OFFLINE. *Secondary site:* The first node to join from the other site acquires the resource group in ONLINE_SECONDARY state. **Inter-site Fallover** The ONLINE instance falls between sites when no nodes at the local site can acquire the resource group. The secondary instance moves to the other site and is brought ONLINE SECONDARY on the highest priority node available, if possible. **Inter-site Fallback** The ONLINE instance does *not* fall back to the primary site when a node on the primary site rejoins. The highest priority rejoining node acquires the resource group in the ONLINE _SECONDARY state.

NodeStartup Policy (applies within site)	**Inter-Site Management Policy**	**Startup/Fallover/Fallback Behavior**
Online on First Available Node *or* Online Using Node Distribution Policy	Online on Either Site	**Cluster Startup** The node that joins first from either site (that meets the distribution criteria) acquires the resource group in the ONLINE state. *Secondary site:* Once the resource group is ONLINE, the first joining node from the other site acquires the resource group in ONLINE_SECONDARY state. **Inter-site Fallover** The ONLINE instance falls between sites when no nodes at the local site can acquire the resource group. **Inter-site Fallback** The ONLINE instance does *not* fall back to the primary site when the primary site joins. Rejoining node acquires resource group in the ONLINE _SECONDARY state.
Online on all Available Nodes	Online on Either Site	**Cluster Startup** The node that joins first from either site acquires the resource group in the ONLINE state. Once an instance of the group is active, the rest of the nodes in the same site also activate the group in the ONLINE state. *Secondary site:* All nodes acquire the resource group in ONLINE_SECONDARY state. **Inter-site Fallover** The ONLINE instance falls between sites when all nodes at the local site go OFFLINE or fail to start the resource group. **Inter-site Fallback** The ONLINE instance does *not* fall back to the primary site when the primary site joins. Rejoining nodes acquire the resource group in the ONLINE _SECONDARY state.

NodeStartup Policy (applies within site)	Inter-Site Management Policy	Startup/Fallover/Fallback Behavior
Online on all Available Nodes	Online at Both Sites	**Cluster Startup** All nodes at both sites activate the resource group in the ONLINE state. **Inter-site Fallover** No fallover. Resource group is either OFFLINE or in ERROR state. **Inter-site Fallback** No fallback.

See also Appendix B of the *Administration Guide* for several sample configurations and use cases with different configuration using resource group dependencies.

Customizing Inter-Site Resource Group Recovery

For a *new* installation of HACMP 5.4.1, inter-site resource group recovery is enabled by default.

Fallover of resource groups between sites is disabled by default when you upgrade to HACMP 5.4.1 from a release prior to version 5.3. This is the pre-5.3 release behavior for non-Ignore site management policy. A particular instance of a resource group can fall over within one site, but cannot move between sites. If no nodes are available on the site where the affected instance resides, that instance goes into ERROR or ERROR_SECONDARY state. It does *not* stay on the node where it failed. This behavior applies to both primary and secondary instances.

Note that in HACMP 5.3 and up, the Cluster Manager will move the resource group if a **node_down** or **node_up** event occurs, even if fallover between sites is disabled. You can also manually move a resource group between sites.

Enabling or Disabling Fallover between Sites

If you migrated from a previous release of HACMP, you can change the resource group recovery policy to allow the Cluster Manager to move a resource group to another site to avoid having the resource group go into ERROR state.

Recovery of Primary Instances of Replicated Resource Groups across Sites

When fallover across sites is enabled, HACMP tries to recover the primary instance of a resource group in situations where an interface connected to an inter-site network fails or becomes available.

Recovery of Secondary Instances of Replicated Resource Groups across Sites

When fallover across sites is enabled, HACMP tries to recover the secondary instance as well as the primary instance of a resource group in these situations:

- If an acquisition failure occurs while the secondary instance of a resource group is being acquired, the Cluster Manager tries to recover the resource group's secondary instance, as it does for the primary instance. If no nodes are available for the acquisition, the resource group's secondary instance goes into global ERROR_SECONDARY state.

- If quorum loss is triggered, and the resource group has its secondary instance online on the affected node, HACMP tries to recover the secondary instance on another available node.
- If a **local_network_down** occurs on an **XD_data** network, HACMP moves resource groups that are ONLINE on the particular node that have GLVM or HAGEO resources to another available node on that site. This functionality of the primary instance is mirrored to the secondary instance so that secondary instances may be recovered via selective fallover.

Using SMIT to Enable or Disable Inter-Site Resource Group Recovery

To enable or disable inter-site Resource Group Recovery, use the following path in HACMP SMIT: **Extended Configuration > Extended Resource Configuration > HACMP Extended Resources Configuration > Customize Resource Group and Resource Recovery > Customize Inter-site Resource Group Recovery**.

Planning for Replicated Resources

HACMP 5.4.1 offers much fuller support for replicated resources, including configuration and processing improvements. Many previous limitations have been eliminated for HACMP/XD replicated resources. In addition HACMP 5.4.1:

- Enables you to dynamically reconfigure resource groups that contain replicated resources with HACMP/XD and HACMP site configurations.
- Consolidates HACMP/XD verification into standard cluster verification by automatically detecting and calling the installed XD product's verification utilities.

See more information in Planning Resource Groups in Clusters with Sites.

Configuration of Replicated Resources

If you have installed an HACMP/XD product, the following configurations are supported:

- Resource groups with concurrent node policy can have non-concurrent site management policy.
- Inter-site recovery of resource groups containing HACMP/XD replicated resources is allowed by default for new installations of HACMP 5.3 and 5.4.1. Configurations updated and migrated from previous releases maintain the pre-existing behavior. You can configure this behavior to be **fallover** or **notify** on cluster-initiated resource group movement. If you select the **notify** option, you need to configure a pre- or post-event script or a remote notification method. See the section on Customizing Resource Group Recovery in the *Administration Guide.*
- Parent/child and location dependency configurations for replicated resource groups.
- Node-based resource group distribution startup policy for resource groups with HACMP sites. (Network-based resource group distribution is no longer an available option.)

The following rules and restrictions apply to replicated resource groups:

- You can have a service IP in a resource group with GeoMirror devices (GMDs) but the service IP cannot be placed on an XD_data network.

- You cannot configure a resource group to use a non-concurrent node policy and a concurrent inter-site management policy.

Note: See the HACMP/XD documentation for complete information on configuring HAGEO, Metro Mirror, or GLVM resources and resource groups.

Processing of Replicated Resources

HACMP provides the following functionality for processing of replicated resources:

- Whenever possible, HACMP processes events in parallel by default. Events are dynamically phased so that HACMP processes the primary and secondary instances of a resource group in proper order (release_primary, release_secondary, acquire_secondary, acquire_primary).
- HACMP now recovers the secondary instances (as well as the primary instances) of the replicated resource groups during volume group losses, acquisition failures, and **local_network_down** events if another node or network is available.
- HACMP has a better chance of acquiring the secondary instance of a resource group upon site fallover. Now HACMP can consider all nodes at the secondary site as targets, *not* just the node that previously hosted the primary instance as in the previous version of HACMP.

  For complete information about HACMP/XD facilities and how they integrate with HACMP, see the HACMP/XD documentation at the following URL:

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

  Also see the section Planning Resource Groups in Clusters with Sites in this chapter.

Moving Resource Groups with Replicated Resources

You can move the primary instance of a resource group with replicated resources to another site. The Cluster Manager uses dynamic event phasing and first moves the secondary instance from that site to the other site, as long as a node is available there to host it. Every attempt is made to maintain the secondary instance in SECONDARY ONLINE state. Even if a node at a given site is configured so it cannot host more than one primary instance, it may host more than one secondary instance in order to keep them all SECONDARY ONLINE.

Recovering Resource Groups on Node Startup

Prior to HACMP 5.2, when a node joined the cluster, the node did *not* make an attempt to 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.

In HACMP 5.2 and up, resource group recovery is improved. An attempt is made to automatically bring online the resource groups that are currently in the ERROR state. This further increases the chances of bringing the applications back online. If a resource group is in the ERROR state on any node in the cluster, the node attempts to acquire it during a **node_up** event. 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 after all nodes have been tried the resource group was *not* acquired, it goes into an ERROR state.
- If the starting node fails to activate a *concurrent resource group* that is in the ERROR state, the concurrent resource group is left in the ERROR state.

Planning for Workload Manager

IBM offers AIX Workload Manager (WLM) as a system administration resource included with AIX 4.3.3 and above. WLM allows users to set targets for and limits on CPU, physical memory usage, and disk I/O bandwidth for different processes and applications. This provides better control over the use of critical system resources at peak loads. HACMP allows you to configure WLM classes into HACMP resource groups so that the starting and stopping of WLM and the active WLM configuration can be under cluster control.

HACMP does *not* verify every aspect of your WLM configuration; therefore, it remains your responsibility to ensure the integrity of the WLM configuration files. After you add the WLM classes to an HACMP resource group, the verification utility checks only whether the required WLM classes exist. Therefore, you must fully understand how WLM works, and configure it carefully. Incorrect but legal configuration parameters can impede the productivity and availability of your system.

For complete information on how to set up and use Workload Manager, see the IBM *AIX Workload Manager (WLM)* Redbook at the following URL:

http://www.redbooks.ibm.com

Workload Manager distributes system resources among processes that request them according to the class they are in. Processes are assigned to specific classes according to class assignment rules. Planning for WLM integration with HACMP includes two basic steps:

1. Using AIX SMIT panels to define the WLM classes and class assignment rules related to highly available applications.
2. Using HACMP SMIT panels to establish the association between the WLM configuration and the HACMP resource groups. See Chapter 4: Configuring HACMP Resource Groups (Extended) in the *Administration Guide* for more information.

About Workload Manager Classes

Workload Manager distributes system resources among processes that request them according to the class to which the processes are assigned. The properties of a class include:

- Name of the class. A unique alphanumeric string no more than 16 characters long.
- Class tier. A number from 0 to 9. This number determines the relative importance of a class from most important (tier 0) to least important (tier 9).

- Number of the CPU and physical memory “shares.” The actual amount of resources allotted to each class depends on the total number of shares in all classes (thus, if two classes are defined on a system, one with two shares of target CPU usage and the other with three shares, the first class will receive 2/5 and the second class will receive 3/5 of the CPU time).
- Configuration limits. Minimum and maximum percentage limits of CPU time, physical memory, and disk I/O bandwidth accessible to the process.

You set up class assignment rules that tell WLM how to classify all new processes (as well as those already running at the time of WLM startup) according to their gid, uid and the full pathname.

Workload Manager Reconfiguration, Startup, and Shutdown

This section describes the way WLM is reconfigured or started or stopped once you have placed WLM under the control of HACMP.

Workload Manager Reconfiguration

After WLM classes are added to an HACMP resource group, then at the time of cluster synchronization on the node, HACMP reconfigures WLM to use the rules required by the classes associated with the node. In the event of dynamic resource reconfiguration on the node, WLM will be reconfigured in accordance with any changes made to WLM classes associated with a resource group.

Workload Manager Startup

WLM startup occurs either when the node joins the cluster or when a dynamic reconfiguration of the WLM configuration takes place.

The configuration is node-specific and depends upon the resource groups in which the node participates. If the node cannot acquire any resource groups associated with WLM classes, WLM will be *not* be started.

For all non-concurrent resource groups that do *not* have the Online Using Node Distribution Policy Startup, the startup script determines whether the resource group is running on a primary or on a secondary node and adds the corresponding WLM class assignment rules to the WLM configuration. For all other non-concurrent resource groups, and for concurrent access resource groups that the node can acquire, the primary WLM class associated with each resource group is placed in the WLM configuration; the corresponding rules are added to the rules table.

Finally, if WLM is currently running and was *not* started by HACMP, the startup script will restart WLM from the user-specified configuration, saving the previous configuration. When HACMP is stopped, it returns WLM back to its previous configuration.

Failure to start up WLM generates an error message logged in the **hacmp.out** log file, but node startup and the resource reconfiguration will proceed.

Workload Manager Shutdown

WLM shutdown occurs either when the node leaves the cluster or on dynamic cluster reconfiguration. If WLM is currently running, the shutdown script will check if the WLM was running before being started by the HACMP and what configuration it was using. It will then either do nothing (if WLM is *not* currently running), or will stop WLM (if it was *not* running before HACMP startup), or stop it and restart it in the previous configuration (if WLM was previously running).

Limitations and Considerations

Keep in mind the following when planning your Workload Manager configuration:

- Some WLM configurations may impede HACMP performance. Be careful when designing your classes and rules, and be alert to how they may affect HACMP.
- You can have no more than 27 non-default WLM classes across the cluster, since one configuration is shared across the cluster nodes.
- An HACMP Workload Manager configuration does *not* support sub-classes, even though WLM allows them in AIX v.5.2. If you configure sub-classes for any WLM classes that are placed in a resource group, a warning will be issued upon cluster verification, and the sub-classes will *not* be propagated to other nodes during synchronization.
- On any given node, only the rules for classes associated with resource groups that can be acquired by a node are active on that node.

Assigning WLM Classes to HACMP Resource Groups

As you plan how to assign the previously configured WLM classes to cluster resource groups, start by filling in the **Primary Workload Manager Class** and **Secondary Workload Manager Class** fields for each resource group in the Resource Groups Worksheets. For information about completing the Resource Groups Worksheet, see the section Completing the Resource Group Worksheet.

The procedure for adding WLM classes as resources in resource groups is described in the section Configuring Workload Manager in Chapter 4: Configuring HACMP Resource Groups (Extended) in the *Administration Guide*.

Completing the Resource Group Worksheet

The Resource Groups Worksheet helps you plan the resource groups for the cluster. Complete one for each resource group.

Note: For examples of location dependency and resource group behavior, see Appendix B: Resource Group Behavior During Cluster Events in the *Administration Guide*.

Appendix A: Planning Worksheets has blank copies of the Resource Group Worksheet. Make copies of the worksheet to record configuration information.

To complete a Resource Group Worksheet:

1. Record the cluster name in the **Cluster Name** field. You first assigned this value in Chapter 2: Initial Cluster Planning.

2. Assign a name to the resource group and record it in the **Resource Group Name** field. Use no more than 32 characters. You can use alphabetic or numeric characters and underscores, but do *not* use a leading numeric. Duplicate entries are *not* allowed.

3. Record the names of the nodes you want to be members of the resource group nodelist for this resource group in the **Participating Nodes/Default Node Priority** field. List the node names in order from highest to lowest priority (this does *not* apply to concurrent resource groups).

4. (*Optional*) Record the inter-site management policy. This choice is only available if you are using the Extended Configuration path. Cluster configurations are *not* typically set up with multiple sites unless you are installing HACMP/XD. Otherwise, unless appropriate methods or customization are provided to handle site operations, **Ignore** should be used for the **Inter-Site Management Policy** field.

   Ignore (default). Use this option if sites are *not* being used in the cluster. This option is also allowed for XD/Metro Mirror configurations.

   Prefer Primary Site. The primary instance of the resource group is brought online on a node at this site and will fallback to this site when it rejoins the cluster after a fallover. The secondary instance is maintained on the other site.

   Online on Either Site. The resource group node policy determines where the primary instance of the resource group will startup, fallover, and fallback. The secondary instance is maintained on the other site.

   Online on Both Sites. Select this option if you want replicated resources to be accessible from all sites. If the site relationship is **Online on Both Sites**, the node policy must be Available on All Nodes.

5. Specify a **Startup Policy**, a **Fallover Policy,** and **Fallback Policy**.

   For information about these settings, see the section Resource Group Policies for Startup, Fallover and Fallback.

6. *(Optional)* You can also specify a **Delayed Fallback Timer** and **Settling Time**.

   For information about these settings, see the section Resource Group Policies for Startup, Fallover and Fallback.

7. (*Optional*) Record the resource group runtime policy. Runtime policies are only available on the Extended Configuration Path and include:

   - Dynamic node priority policy
   - Dependencies between resource groups
   - Workload Manager
   - Resource group processing order

8. (*Optional*) Record the Dynamic Node Priority Policy you plan to use for this resource group. (This field appears on **Extended Configuration** path **Change/Show a Resource/Attribute field**.)The default is blank. The ordered nodelist is the default policy. For concurrent resource groups, this is the only choice. To use a dynamic node priority policy, select one of the predefined dynamic node priority policies.

9. *(Optional)* Record the dependency (parent/child or location) you plan to use for this resource group. See the appropriate section in HACMP offers a wide variety of configurations where you can specify the relationships between resource groups that you want to maintain at startup, fallover, and fallback. You can configure: to review the guidelines for these configurations.

10. (*Optional*) Record the resource group processing order. In the **Processing Order: Parallel, Customized or Serial** field, identify whether you would like HACMP to acquire and release this resource group in parallel (default) or serially.

    For more information, see Planning Parallel or Serial Order for Processing Resource Groups.

11. List the resources to be included in the resource group. You have identified the resources in previous chapters. In this section of the Resource Group Worksheet, you record the following resources/attributes:

    - Enter the **Service IP Label** in this field if your cluster uses IP Address Takeover.
    - *This field relates only to non-concurrent resource groups.* Leave the field **Filesystems (default is All)** blank, if you want *all* file systems in the specified volume groups to be mounted by default when the resource group, containing this volume group, is brought online.

    Note: If you leave the **Filesystems (default is All)** field blank *and* specify the shared volume groups in the **Volume Groups** field, then all file systems will be mounted in the volume group. If you leave the **Filesystems** field blank and do *not* specify the volume groups in the **Volume Groups** field, no file systems will be mounted.

    You may list the individual file systems to include in this resource group in the **Filesystems (default is All)** field. In this case, only the specified file systems will be mounted when the resource group is brought online.

    - In the **Filesystems Consistency Check** field, specify **fsck** or **logredo**. If you choose **logredo** and it fails, then **fsck** is run instead.
    - In the **Filesystems Recovery Method** field, specify **parallel** or **sequential**.
    - In the **Filesystems to Export (NFSv2/3)** field, enter the mount points of the file systems, the directories that are exported using NFSv2/3 protocol to all the nodes in the resource group nodelist when the resource is initially acquired, changed, or both.
    - In the **Filesystems to Export (NFSv4)** field, enter the mount points of the file systems or directories that are exported using NFSv4 protocol to all the nodes in the resource group nodelist when the resource is initially acquired, changed, or both. A given file system can be present in the **Filesystems to Export(NFSv2/3)** and the **Filesystems to Export (NFSv4)** fields. In this scenario, the file system or directory is exported using NFSv2/3 and NFSv4 protocols.
    - In the **Stable Storage Path (NFSv4)** field, enter a path that can be used to store the state information by the NFSv4 server.
    - List the file systems that should be NFS-mounted by the nodes in the resource group nodelist *not* currently holding the resource in the **Filesystems to NFS Mount** field. All nodes in the resource group nodelist that do *not* currently hold the resource will attempt to NFS-mount these file systems.

- In the **Network for NFS Mount** field, enter the preferred network to NFS-mount the file systems specified.
- List in the **Volume Groups** field the names of the volume groups containing raw logical volumes or raw volume groups that are varied on when the resource is initially acquired.

  Specify the shared volume groups in the **Volume Groups** field if you want to leave the field **Filesystems (default is All)** blank, *and* want to mount all file systems in the volume group. If you specify more than one volume group in this field, then you cannot choose to mount all file systems in one volume group and *not* in another; all file systems in all specified volume groups will be mounted.

  For example, in a resource group with two volume groups, *vg1* and *vg2*, if the **Filesystems (default is All)** is left blank, then all the file systems in *vg1* and *vg2* will be mounted when the resource group is brought up. However, if the **Filesystems (default is All)** has only file systems that are part of the vg1 volume group, then none of the file systems in *vg2* will be mounted, because they were *not* entered in the **Filesystems (default is All)** field along with the file systems from *vg1*.

  If you plan to use raw logical volumes, you only need to specify the volume group in which the raw logical volume resides in order to include the raw logical volumes in the resource group.
- In the **Concurrent Volume Groups** field, enter the names of the concurrent volume groups that are to be varied on by the owner node.
- If you plan on using an application that directly accesses raw disks, list the physical volume IDs of the raw disks in the **Raw Disk PVIDs** field.
- If you are using **Fast Connect Services,** define the resources in this field.
- If using **Tape Resources,** enter the name of the tape resource in this field.
- List the names of the application servers to include in the resource group in the **Application Servers** field.
- List the **Communications Links** for SNA-over-LAN, SNA-over-X.25, or X.25.
- In the **Primary Workload Manager Class** and **Secondary Workload Manager Class** fields, fill in the name of a class associated with the HACMP WLM configuration that you want to add to this resource group.

  For non-concurrent resource groups that do *not* have the Online Using Node Distribution Policy startup, if no secondary WLM class is specified, all nodes will use the primary WLM class. If a secondary class is also specified, only the primary node will use the primary WLM class. Secondary classes cannot be assigned to non-concurrent resource groups with the Online Using Node Distribution Policy startup and concurrent resource groups; for these resource group types, all nodes in the resource group use the primary WLM class.

  Note: Before adding WLM classes to a resource group, specify a WLM configuration in the **Change/Show HACMP WLM runtime Parameters** SMIT panel. The picklists for the Primary/Secondary WLM Classes are populated with the classes defined for the specified WLM configuration. For more information, see Chapter 4: Configuring HACMP Resource Groups (Extended) in the *Administration Guide*.

- **Miscellaneous Data** is a string placed into the environment along with the resource group information and is accessible by scripts.
- The **Automatically Import Volume Groups** field is set to **false** by default. Definitions of "available for import volume groups" are presented here from the file created the last time the information was collected from the **Discover Current Volume Group** menu. No updating of volume group information is done automatically.

  If reset to **true,** causes the definition of any volume groups entered in the **Volume Groups** field or the **Concurrent Volume Groups** field to be imported to any resource group nodes that don't already have it.

  When **Automatically Import Volume Groups** is set to **true**, the final state of the volume group will depend on the initial state of the volume group (varied on or varied off) and the state of the resource group to which the volume group is to be added (online or offline).
- Set the **Disk Fencing Activated** field to **true** to activate SSA Disk Fencing for the disks in this resource group. Set to **false** to disable. SSA Disk Fencing helps prevent partitioned clusters from forcing inappropriate takeover of resources.
- **Filesystems Mounted before IP Configured**. HACMP handles node failure by taking over the failed node's IP address(es) and then taking over its file systems. This results in "Missing File or File System" errors for NFS clients since the clients can communicate with the backup server before the file systems are available. Set to **true** to takeover file systems before taking over IP address(es) that will prevent an error. Set to **false** to keep the default order.

Where You Go from Here

You have now planned the resource groups for the cluster. You next customize the cluster event processing for your cluster. Chapter 7: Planning for Cluster Events discusses this step.

Chapter 7: Planning for Cluster Events

This chapter describes the HACMP cluster events.

Prerequisites

Before reading this chapter, read the overview information about cluster events in the *Concepts and Facilities Guide*.

By now, you should have completed the planning steps in the previous chapters:

- Chapter 2: Initial Cluster Planning
- Chapter 3: Planning Cluster Network Connectivity
- Chapter 4: Planning Shared Disk and Tape Devices
- Chapter 5: Planning Shared LVM Components
- Chapter 6: Planning Resource Groups.

Overview

The following sections describe the main types of HACMP events and the logic for recovery actions.

This chapter contains the following sections:

- Planning Site and Node Events
- Planning node_up and node_down Events
- Network Events
- Network Interface Events
- Cluster-Wide Status Events
- Resource Group Event Handling and Recovery
- Customizing Cluster Event Processing
- Custom Remote Notification of Events
- Customizing Event Duration Time Until Warning
- User-Defined Events
- Event Summaries and Preamble
- Completing the Cluster Event Worksheet

HACMP provides two ways to manage event processing:

- Customize predefined events
- Define new events.

In HACMP 5.4.1, resource groups are processed in parallel by default, if possible, unless you specify a customized serial processing order for all or some of the resource groups in the cluster.

The logic and sequence of events as described in examples may *not* list *all* events. For additional information on event processing, see Appendix B, Resource Group Behavior during Cluster Events in the *Administration Guide* and Chapter: 2 Using Cluster Log Files in the *Troubleshooting Guide.*

See Chapter 8: Starting and Stopping Cluster Services in the *Administration Guide* for information on:

- Steps you take to start and stop cluster services
- Interaction with the AIX shutdown command and interaction of HACMP cluster services with RSCT.

Planning Site and Node Events

Defining a site is optional unless you have installed HACMP/XD or are using cross-site mirroring. HACMP/XD includes HACMP/XD for Metro Mirror, GLVM or HAGEO; all of these make use of sites for replicated resources.

Site event scripts are included in the HACMP software. If sites are *not* defined, no site events are generated. The HACMP **site_event** scripts run as follows if sites are defined:

- The first node in a site runs **site_up** before it completes **node_up** event processing. The **site_up_complete** event runs after **node_up_complete**.
- When the last node in a site goes down, the **site_down** event runs before **node_down**, and **site_down_complete** runs after **node_down_complete**.

Without installing HACMP/XD, you can define pre- and post-events to run when a site changes state. In this case, you define all site-related processes.

Site events (including **check_for_site_up** and **check_for_site_down**) are logged in the **hacmp.out** log file. Some other site events, such as **site_isolation** and **site_merge**, may occur in clusters with sites. HACMP does *not* run event scripts and no additional events are initiated for these events. These are places where you may want to customize actions for your site.

If sites are defined, then **site_up** runs when the first node in the site comes up and **site_down** runs when the last node in the site goes down. The event script sequence for handling resource groups in general is:

site_up

site_up_remote

node_up

rg_move events to process resource group actions

node_up_complete

site_up_complete

site_up_remote_complete

----

site_down

site_down_remote

node_down

rg_move events to process resource group actions

node_down_complete

site_down

site_down_remote_complete

Planning node_up and node_down Events

A **node_up** event is initiated by a node joining the cluster at cluster startup, or by rejoining the cluster at a later time.

Establishing Initial Cluster Membership

This section describes the steps taken by the Cluster Manager on each node when the cluster starts and the initial membership of the cluster is established. It shows how the Cluster Managers establish communication among the member nodes, and how the cluster resources are distributed as the cluster membership grows.

First Node Joins the Cluster

1. HACMP cluster services are started on Node A. The RSCT subsystem examines the state of the network interfaces and begins communicating with the RSCT subsystem on the other cluster nodes. The Cluster Manager on Node A accumulates the initial state information and then broadcasts a message indicating that it is ready to join the cluster on all configured networks to which it is attached (**node_up**).
2. Node A interprets the lack of a response to mean that it is the first node in the cluster.
3. Node A then initiates a **process_resources** script, which processes the node’s resource configuration information.

4. When the event processing has completed, Node A becomes a member of the cluster. HACMP runs **node_up_complete**.

   All resource groups defined for Node A are available for clients at this point.

   If the **Online on First Available Node** is specified as a startup behavior for resource groups, then Node A will take control of all these resource groups.

   If Node A is defined as part of a non-concurrent resource group that has the **Online Using Node Distribution Policy** startup, then this node takes control of the first resource group listed in the node environment.

   If Node A is defined as part of a concurrent access resource configuration, it makes those concurrent resources available.

   For resource groups with **Online on First Available Node** startup policy and the settling time configured, Node A waits for the settling time interval before acquiring such resource groups. The settling time facilitates waiting for a higher priority node to join the cluster.

Second Node Joins the Cluster

5. HACMP Cluster Services are started on Node B. Node B broadcasts a message indicating that it is ready to join the cluster on all configured networks to which it is attached (**node_up**).
6. Node A receives the message and sends an acknowledgment.
7. Node A adds Node B to a list of active nodes, starts keepalive communications with Node B.
8. Node B receives the acknowledgment from Node A. The message includes information identifying Node A as the only other member of the cluster. (If there were other members, Node B would receive the list of members.)
9. Node B processes the **process_resources** script and sends a message to let other nodes know when it is finished.

   Processing the **process_resources** script may include Node A releasing resources it currently holds, if both nodes are in the resource group nodelist for one or more resource groups and Node B has a higher priority for one or more of those resources. This is true only for resource groups that support fallback.

   Note that if the delayed fallback timer is configured, any resource group that is online on node A and for which Node B is a higher priority node will fall back to node B at the time specified by the delayed fallback timer.

10. Meanwhile, Node B has been monitoring and sending keepalives, and waiting to receive messages about changes in the cluster membership. When Node B finishes its own **process_resources** script, it notifies Node A.

    During its **node_up** processing, Node B claims all resource groups configured for it (see step 3). Note that if the delayed fallback timer is configured, the resource group will fall back to a higher priority node at the time specified by the timer.

11. Both nodes process a **node_up_complete** event simultaneously.

    At this point, Node B includes Node A in its list of member nodes and its list of keepalive.

12. Node B sends a “new member” message to all possible nodes in the cluster.

13. When Node A gets the message, it moves Node B from its list of active nodes to its list of member nodes.

    At this point, all resource groups configured for Node A and Node B are available to cluster clients.

Remaining Nodes Join the Cluster

14. As HACMP cluster Services start on each remaining cluster node, steps 4 through 9 repeat, with each member node sending and receiving control messages, and processing events in the order outlined. Note especially that all nodes must confirm the **node_up_complete** event before completing the processing of the event and moving the new node to the cluster member list.

    As new nodes join, the RSCT subsystem on each node establishes communications and begins sending heartbeats. Nodes and adapters join RSCT heartbeat rings that are formed according to the definition in the HACMP configuration. When the status of a NIC or node changes, the Cluster Manager receives the state change and generates the appropriate event.

Rejoining the Cluster

When a node rejoins the cluster, the Cluster Managers running on the existing nodes initiate a **node_up** event to acknowledge that the returning node is up. When these nodes have completed processing their **process_resources** script, the new node then processes a **node_up** event so that it can resume providing cluster services.

This processing is necessary to ensure the proper balance of cluster resources. As long as the existing Cluster Managers first acknowledge a node rejoining the cluster, they can release any resource groups belonging to that node if necessary. Whether or *not* the resource groups are actually released in this situation depends on how the resource groups are configured for takeover (or dependencies). The new node can then start its operations.

Sequence of node_up Events

The following list describes the sequence of **node_up** events:

node_up

This event occurs when a node joins or rejoins the cluster.

process_resources

This script calls the sub_events needed for the node to acquire the service address (or shared address), gets all its owned (or shared) resources, and take the resources. This includes making disks available, varying on volume groups, mounting file systems, exporting file systems, NFS-mounting file systems, and varying on concurrent access volume groups.

It may take other actions depending on the resource type configuration, such as getting Fast Connect, and so forth.

process_resources_complete

Each node runs this script when resources have been processed.

node_up_complete

This event occurs after the resources are processed and the **node_up** event has successfully completed. Depending on whether the node is local or remote, this event calls the **start_server** script to start application servers on the local node, or allows the local node to do an NFS mount only after the remote node is completely up.

node_up Events with Dependent Resource Groups or Sites

If either sites or dependencies between any resource groups are configured in the cluster, HACMP processes all events related to resource groups in the cluster with the use of **rg_move** events that are launched for all resource groups when **node_up** events occur.

The Cluster Manager then takes into account all node and site policies, especially the configuration of dependencies and sites for resource groups, and the current distribution and state of resource groups on all nodes in order to properly handle any acquiring, releasing, bringing online or taking offline of resource groups before **node_up_complete** can run.

Parent and child or location dependencies between resource groups offer a predictable and reliable way of building clusters with multi-tiered applications. However, **node_up** processing in clusters with dependencies could take more time than the parallel processing in clusters without resource groups' dependencies. You may need to adjust the **config_too_long** warning timer for **node_up**.

For more information on processing in clusters where dependencies are specified between some of the resource groups, see the *Administration Guide*.

node_down Events

Cluster nodes exchange keepalives through the RSCT subsystem with peer nodes so that the Cluster Manager can track the status of the nodes in the cluster. A node that fails or is stopped purposefully no longer sends keepalives. When RSCT indicates all network interfaces are down, or a node does *not* respond to heartbeats, the Cluster Managers then run a **node_down** event. Depending on the cluster configuration, the peer nodes then take the necessary actions to get critical applications up and running and to ensure data remains available.

A **node_down** event can be initiated by a node:

- Stopping cluster services and bringing resource groups offline
- Stopping cluster services and moving resource groups to another node
- Stopping cluster services and placing resource groups in an UNMANAGED state.
- Failing.

Stopping Cluster Services and Bringing Resource Groups Offline

When you stop cluster services and bring resource groups offline,, HACMP stops on the local node after the **node_down_complete** event releases the stopped node's resources. The other nodes run the **node_down_complete** event and do *not* take over the resources of the stopped node.

Stopping Cluster Services and Moving Resource Groups

When you stop cluster services and move the resource groups to another node,, HACMP stops after the **node_down_complete** event on the local node releases its resource groups. The surviving nodes in the resource group nodelist take over these resource groups.

Stopping Cluster Services and Placing Resource Groups in an UNMANAGED State

When you stop cluster services and place resource groups in an UNMANAGED state,, HACMP software stops immediately on the local node. **Node_down_complete** is run on the stopped node. The Cluster Managers on remote nodes process **node_down** events, but do *not* take over any resource groups. The stopped node does *not* release its resource groups.

Node Failure

When a node fails, the Cluster Manager on that node does *not* have time to generate a **node_down** event. In this case, the Cluster Managers on the surviving nodes recognize a **node_down** event has occurred (when they realize the failed node is no longer communicating) and trigger **node_down** events.

This initiates a series of sub_events that reconfigure the cluster to deal with that failed node. Based upon the cluster configuration, surviving nodes in the resource group nodelist will take over the resource groups.

Sequence of node_down Events

The following list describes the default parallel sequence of **node_down** events:

node_down

This event occurs when a node intentionally leaves the cluster or fails.

In some cases, the **node_down** event receives the **forced** parameter. For more information, see the section Stopping Cluster Services and Placing Resource Groups in an UNMANAGED State.

All nodes run the **node_down** event.

> All nodes run the **process_resources** script. Once the Cluster Manager has evaluated the status of affected resource groups and the configuration, it initiates a series of sub_events, to redistribute resources as configured for fallover or fallback.
>
> All nodes run the **process_resources_complete** script.

node_down_complete

Network Events

HACMP distinguishes between two types of network failure, *local* and *global*, and uses different network failure events for each type of failure. The network failure event script is often customized to send mail.

Sequence of Network Events

The following list shows the network events:

network_down (Local)

This event occurs when only a particular node has lost contact with a network. The event has the following format:

```
network_down node_name network_name
```

The Cluster Manager takes selective recovery action to move affected resource groups to other nodes. The results of the recovery actions are logged to **hacmp.out.**

network_down (Global)

This event occurs when all of the nodes connected to a network have lost contact with a network. It is assumed in this case that a network-related failure has occurred rather than a node-related failure. This event has the following format:

```
network_down -1 network_name
```

Note: The **-1** argument is - *one*. This argument indicates that the **network_down** event is global.

The global network failure event mails a notification to the system administrator, but takes no further action since appropriate actions depend on the local network configuration.

network_down_complete (Local)

This event occurs after a local network failure event has completed. It has the following format:

```
network_down_complete node_name network_name
```

When a local network failure event occurs, the Cluster Manager takes selective recovery actions for resource groups containing a service NIC connected to that network.

network_down_complete (Global)

This event occurs after a global network failure event has completed. It has the following format:

```
network_down_complete -1 network_name
```

The default processing for this event takes no actions since appropriate actions depend on the network configuration.

network_up

This event occurs when the Cluster Manager determines a network has become available for use.

Whenever a network becomes available again, HACMP attempts to bring resource groups containing service IP labels on that network back online.

network_up_complete

This event occurs only after a **network_up** event has successfully completed. This event is often customized to notify the system administrator that an event demands manual attention.

Whenever a network becomes available again, HACMP attempts to bring resource groups containing service IP labels on that network back online.

For more information, see Appendix B: Resource Group Behavior during Cluster Events in the *Administration Guide*.

Network Interface Events

The Cluster Manager reacts to the failure, unavailability, or joining of network interfaces by initiating one of the following events. (For exceptions, see the section that follows on single interface situations.)

swap_adapter

This event occurs when the interface hosting a service IP label on a node fails. The **swap_adapter** event moves the service IP label onto a non-service interface on the same HACMP network and then reconstructs the routing table. If the service IP label is an IP alias, it is put onto the non-service interface as an additional IP label. Otherwise, the non-service IP label is removed from the interface and placed on the failed interface.

If the interface now holding the service IP label later fails, **swap_adapter** can switch to another non-service interface if one exists.

If a persistent node IP label was assigned to the failed interface, it moves with the service label to the non-service interface.

Note: HACMP removes IP aliases from interfaces at shutdown. It creates the aliases again when the network becomes operational. The **hacmp.out** file records these changes.

swap_adapter_complete

This event occurs only after a **swap_adapter** event has successfully completed. The **swap_adapter_complete** event ensures that the local ARP cache is updated by deleting entries and pinging cluster IP addresses.

fail_standby

This event occurs if a non-service interface fails or becomes unavailable as the result of an IP address takeover. The **fail_standby** event displays a console message indicating that a non-service interface has failed or is no longer available.

join_standby

This event occurs if a non-service interface becomes available. The **join_standby** event displays a console message indicating that a non-service interface has become available.

In HACMP 4.5 and up, whenever a network interface becomes available, HACMP attempts to bring resource groups back online. For more information, see Resource Group Recovery when the Network or Interface is Up in Appendix B: Resource Group Behavior During Cluster Events in the *Administration Guide*.

fail_interface

This event occurs if an interface fails and there is no non-service interface available to recover the service address.

Takeover service addresses are monitored. It is possible to have an interface failure *and* no available interface for recovery *and* another interface up on the same network.

This event applies to all networks, including those using IP aliasing for recovery. Note that when a non-service NIC fails on a network configured for IPAT via IP Aliases, the **fail_interface** event is run. An **rg_move** event is then triggered if the interface that failed was a service label.

join_interface

This event occurs if a non-service interface becomes available or recovers. This event applies to all networks, including those using IPAT via IP Aliases for recovery.

Note that networks using IP aliases by definition do *not* have non-service interfaces defined, so the **join_interface** event that is run in this case simply indicates that a non-service interface joins the cluster.

Failure of a Single Network Interface Does *Not* Generate Events

If you have only one network interface active on a network, the Cluster Manager cannot generate a failure event for that network interface, as it has no peers with which to communicate to determine the health of the interface. Situations that have a single network interface include:

- One-node clusters
- Multi-node clusters with only one node active
- Multi-node clusters with virtual Ethernet interfaces
- Failure of all but one interface on a network, one at a time.

For example, starting a cluster with all service or non-service interfaces disconnected produces the following results:

1. First node up: No failure events are generated.
2. Second node up: One failure event is generated.
3. Third node up: One failure event is generated.
4. And so on.
5. See Identifying Service Adapter Failure for Two-Node Clusters in Chapter 3: Planning Cluster Network Connectivity for information on handling this situation.

Cluster-Wide Status Events

By default, the Cluster Manager recognizes a time limit for reconfiguring a cluster and processing topology changes. If the time limit is reached, the Cluster Manager initiates a **config_too_long** event. Whole cluster status events include:

config_too_long

This system warning occurs each time a cluster event takes more time to complete than a specified time-out period. This message is logged in the **hacmp.out** file.

The time-out period for all events is set to 360 seconds by default. You can use SMIT to customize the time period allowed for a cluster event to complete before HACMP issues a **config_too_long** warning for it. For more information, see Chapter 5: Configuring Cluster Events in the *Administration Guide*.

reconfig_topology_start

This event marks the beginning of a dynamic reconfiguration of the cluster topology.

reconfig_topology_complete

This event indicates that a cluster topology dynamic reconfiguration has completed.

reconfig_resource_acquire

This event indicates that cluster resources that are affected by dynamic reconfiguration are being acquired by appropriate nodes.

reconfig_resource_release This event indicates that cluster resources affected by dynamic reconfiguration are being released by appropriate nodes.

reconfig_resource_complete This event indicates that a cluster resource dynamic reconfiguration has successfully completed.

cluster_notify This event is triggered by verification when automatic cluster configuration monitoring detects errors in cluster configuration. The output of this event is logged in **hacmp.out** throughout the cluster on each node that is running cluster services.

event_error Prior to version HACMP 5.2, non-recoverable event script failures resulted in the **event_error** event running on the cluster node where the failure occurred. The remaining cluster nodes do *not* indicate the failure. All cluster nodes run the **event_error** event if any node has a fatal error. All nodes log the error and call out the failing node name in the **hacmp.out** log file.

Resource Group Event Handling and Recovery

The Cluster Manager keeps track of the resource group node priority policies, site policies, and any dependencies configured as well as the necessary topology information and resource group status so that it can take a greater variety of recovery actions, often avoiding the need for user intervention. Event logging includes a detailed summary for each high-level event to help you understand exactly what actions were taken for each resource group during the handling of failures.

For more information about how resource groups are handled in HACMP, see Appendix B: Resource Group Behavior During Cluster Events in the *Administration Guide*. It contains information about the following HACMP functions:

- Selective fallover for handling resource groups
- Handling of resource group acquisition failures
- Handling of resource groups configured with IPAT via IP Aliases
- Handling of HACMP/XD resource groups.

Note: 1. If dependencies between resource groups or sites are specified, HACMP processes events in a different sequence than usual. For more information see the section on the **resource_state_change** event below.

2. The lists in this section do *not* include all possible resource group states: If sites are defined, a primary and a secondary instance of the resource group could be online, offline, in the error state, or unmanaged. Also, the resource group instances could be in the process of acquiring or releasing. The corresponding resource group states are *not* listed here, but have descriptive names that explain which actions take place.

Resource Group Events

The Cluster Manager may move resource groups as a result of recovery actions taken during the processing of events such as node down.

rg_move
This event moves a specified resource group from one node to another.

rg_move_complete
This action indicates that the **rg_move** event has successfully completed.

resource_state_change
This trigger event is used for resource group recovery if resource group dependencies or sites are configured in the cluster. This action indicates that the Cluster Manager needs to change the state of one or more resource groups, or there is a change in the state of a resource managed by the Cluster Manager. This event runs on all nodes if one of the following situations occurs:

- Application monitoring failure
- Selective fallover for loss of volume group
- Local network down
- WAN failure
- Resource Group Acquisition Failure
- Resource Group Recovery on IP Interface Availability
- Expiry of Settling timer for a resource group
- Expiry of fallback timer for a resource group

While the event runs, the state of the resource group is changed to TEMP_ERROR. This is broadcast to all nodes.

resource_state_change_complete
This event runs when the **resource_state_change** event completes successfully. You can add pre- or post-events here if necessary. You may want to be notified about resource state changes, for example.

external_resource_state_change
This event runs when the user moves a resource group and HACMP uses the dynamic processing path to handle the request since resource group dependencies or sites are configured in the cluster.

external_resource_state_change_complete
This event runs when the **external_esource_state_change** event completes successfully.

Resource Group sub_events

Handling of individual resources during the processing of an event may include the following actions. For example, when a file system is in the process of being unmounted and mounted it is taken offline and then released by one node. Then it is acquired by another node and brought online. This list includes some but *not* all possible states of resource groups:

releasing This action indicates that a resource group is being released either to be brought offline or to be acquired on another node.

acquiring This action is used when a resource group is being acquired on a node.

rg_up This action indicates that the resource group is online.

rg_down This action indicates that the resource group is offline.

rg_error This action indicates that the resource group is in error state.

rg_temp_error_state This action indicates that the resource group is in a temporary error state. For example, it occurs due to a local network or an application failure. This state informs the Cluster Manager to initiate an **rg_move** event for this resource group. Resource groups should *not* be in this state when the cluster is stable.

rg_acquiring _secondary The resource group is coming online at the target site (only the replicated resources are online).

rg_up_secondary The resource group is online in the secondary role at the target site (only replicated resources are online).

rg_error_secondary The resource group at the site receiving the mirror data is in error state.

rg_temp_error_secondary The resource group at the site receiving the mirror data is in temporary error state.

After the completion of an event, the Cluster Manager has the state of resources and resource groups involved in the event. The Cluster Manager then analyzes the resource group information that it maintains internally and determines whether recovery events need to be queued for any of the resource groups. The Cluster Manager also uses status of individual resources in resource groups to print out a comprehensive event summary to the **hacmp.out** log file.

For each resource group, the Cluster Manager keeps track of the nodes on which the resource group has tried to come online and failed. This information is updated when recovery events are processed. The Cluster Manager resets the nodelist for a resource group as soon as the resource group moves to the online or error states.

In HACMP 5.4.1, the resource group ERROR states are displayed with more detail than before:

Resource Group is in ERROR because	**HACMP Displays this Message**
Parent group is NOT ONLINE; as a result, the child resource group is unavailable	`OFFLINE due to parent offline`
Higher priority **Different-Node Dependency** group is ONLINE	`OFFLINE due to lack of available node`
Another distributed group was acquired	`OFFLINE`
Group is falling over and in the OFFLINE state temporarily	`OFFLINE`

Manual intervention is only required when a resource group remains in ERROR state after the event processing finishes.

When a resource group is in the process of being moved, application monitoring is suspended and resumed appropriately. The Application Monitor sees that the application is in “recovery” state while the event is being processed.

resume_appmon This action is used by the Application Monitor to resume monitoring of an application.

suspend_appmon This action is used by the Application Monitor to suspend monitoring of an application.

For more information on how resource groups are handled in HACMP, see Appendix B: Resource Group Behavior During Cluster Events in the *Administration Guide*. It contains information on selective fallover for handling resource groups, handling of resource group acquisition failures, handling of resource groups configured with IPAT via IP Aliases, and HACMP/XD resource groups.

Customizing Cluster Event Processing

The Cluster Manager’s ability to recognize a specific series of events and sub_events permits a very flexible customization scheme. The HACMP event customization facility lets you customize cluster event processing to your site. Customizing event processing allows you to provide a highly efficient path to the most critical resources in the event of a failure. However, this efficiency depends on your configuration.

As part of the planning process, you need to decide whether to customize event processing. If the actions taken by the default scripts are sufficient for your purposes, then you do *not* need to do anything further to configure events during the configuration process.

If you do decide to customize event processing to your environment, use the HACMP event customization facility described in this chapter. If you customize event processing, register these user-defined scripts with HACMP during the configuration process.

If necessary, you can modify the default event scripts or write your own. *Modifying the default event scripts or replacing them with your own is strongly discouraged*. This makes maintaining, upgrading, and troubleshooting an HACMP cluster much more difficult. Again, if you write your own event customization scripts, you need to configure the HACMP software to use those scripts.

The event customization facility includes the following functions:

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

Complete customization of an event includes a notification to the system administrator (before and after event processing), and user-defined commands or scripts that run before and after event processing, as shown in the following example:

```
Notify sysadmin of event to be processed
Pre-event script or command
HACMP event script
Post-event script or command
Notify sysadmin that event processing is complete
```

For a discussion of event emulation, which lets you emulate HACMP event scripts without actually affecting the cluster, see the section Event Emulation in Chapter 1: Troubleshooting HACMP Clusters in the *Troubleshooting Guide*.

Event Notification

You can specify a **notify** command that sends mail to indicate that an event is about to happen (or has just occurred), and that an event script succeeded or failed. You configure notification methods for cluster events In SMIT under the **Change/Show a Custom Cluster Events** panel. For example, a site may want to use a network failure notification event to inform system administrators that traffic may have to be re-routed. Afterwards, you can use a **network_up** notification event to tell system administrators that traffic can again be serviced through the restored network.

Event notification in an HACMP cluster can also be done using pre- and post-event scripts. For more information, see Chapter 5: Configuring Cluster Events in the *Administration Guide*.

You can also configure a custom remote notification in response to events. For details, see the section Custom Remote Notification of Events.

Pre- and Post-Event Scripts

You can specify commands or multiple user-defined scripts that execute before and after the Cluster Manager calls an event script.

For example, you can specify one or more pre-event scripts that run before the **node_down** event script is processed. When the Cluster Manager recognizes that a remote node is down, it first processes these user-defined scripts. One such script may designate that a message be sent to all users to indicate that performance may be affected (while adapters are swapped, while application servers are stopped and restarted). Following the **node_down** event script, a post processing event script for **network_up** notification may be included to broadcast a message to all users that a certain system is now available at another network address.

The following scenarios are other examples of where pre- and post-event processing are useful:

- If a **node_down** event occurs, site specific actions may dictate that a pre-event script for the **start_server** subevent script be used. This script could notify users on the server about to takeover for the downed application server that performance may vary, or that they should seek alternate systems for certain applications.
- Due to a network being down, a custom installation may be able to re-route traffic through other machines by creating new IP routes. The **network_up** and **network_up_complete** event scripts could reverse the procedure, ensuring that the proper routes exist after all networks are functioning.
- A site may want to stop cluster services and move resource groups to another node as a post-event if a network has failed on the local node (but otherwise the network is functioning).

Note that when writing your HACMP pre- or post-event, none of the shell environment variables defined in **/etc/environment** are available to your program. If you need to use any of these variables, explicitly source them by including this line in your script: "`. /etc/environment`".

If you plan to create pre- or post-event scripts for your cluster, be aware that your scripts will be passed the same parameters used by the HACMP event script you specify. For pre- and post-event scripts, the arguments passed to the event command are the event name, event exit status, and the trailing arguments passed to the event command.

All HACMP event scripts are maintained in the **/usr/es/sbin/cluster/events** directory. The parameters passed to your script are listed in the event script headers.

WARNING: Be careful *not* to kill any HACMP processes as part of your script. If you are using the output of the **ps** command and using a **grep** to search for a certain pattern, make sure the pattern does *not* match any of the HACMP or RSCT processes.

Pre- and Post-Event Scripts May No Longer Be Needed

If you migrated from a previous version of HACMP, some of your existing pre- and post-event scripts may no longer be needed. HACMP itself handles more situations.

Using the Forced Varyon Attribute Instead of Pre- or Post-Event Scripts

Prior to HACMP 5.1, you could use either pre- or post- event scripts or event recovery routines to achieve a forced activation of volume groups in the case when the activation and acquisition of raw physical volumes and volume groups fails on a node. In HACMP 5.1 and up, you can still use the previously mentioned methods, or you can specify the forced varyon attribute for a volume group. For more information, see Using Forced Varyon.

If the forced varyon attribute is specified for a volume group, special scripts to force a varyon operation are no longer required.

event_error Now Indicates Failure on a Remote Node

Prior to HACMP 5.2, non-recoverable event script failures result in the **event_error** event being run on the cluster node where the failure occurred. The remaining cluster nodes do *not* indicate the failure. With HACMP 5.2 and up, all cluster nodes run the **event_error** event if any node has a fatal error. All nodes log the error and call out the failing node name in the **hacmp.out** log file.

If you have added pre- or post-events for the **event_error** event, be aware that those event methods are called on every node, *not* just the failing node, starting with HACMP 5.2.

A Korn shell environment variable indicates the node where the event script failed: EVENT_FAILED_NODE is set to the name of the node where the event failed. Use this variable in your pre- or post-event script to determine where the failure occurred.

The variable LOCALNODENAME identifies the local node; if LOCALNODENAME is *not* the same as EVENT_FAILED_NODE then the failure occurred on a remote node.

Resource Groups Processed in Parallel and Using Pre- and Post-Event Scripts

Resource groups are processed in parallel by default in HACMP 5.4.1 unless you specify a customized serial processing order for all or some of the resource groups in the cluster.

When resource groups are processed in parallel, fewer cluster events occur in the cluster and appear in the event summaries.

The use of parallel processing reduces the number of particular cluster events for which you can create customized pre- or post-event scripts. If you start using parallel processing for a list of resource groups in your configuration, be aware that some of your existing pre- and post-event scripts may *not* work for these resource groups.

In particular, only the following events take place during parallel processing of resource groups:

node_up

node_down

acquire_svc_addr

acquire_takeover_addr

release_svc_addr

release_takeover_addr

start_server

stop_server

Note: In parallel processing, these events apply to an entire list of resource groups that are being processed in parallel, and *not* to a single resource group, as in serial processing. Prior to HACMP 5.1, if you had pre- and post-event scripts configured for these events, then, after migration, these event scripts are launched *not* for a single resource group but for a list of resource groups, and may *not* work as expected.

The following events *do not occur* in parallel processing of resource groups:

get_disk_vg_fs

release_vg_fs

node_up_ local

node_up_remote

node_down_local

node_down_remote

node_up_local_complete

node_up_remote_complete

node_down_local_complete

node_down_remote_complete

Consider these events that *do not occur* in parallel processing if you have pre- and post-event scripts and plan to upgrade to the current version.

If you want to continue using pre- and post-event scripts, you could have one of the following cases:

Scenario	What You Should Do
You would like to use pre- and post-event scripts for newly added resource groups.	All newly added resource groups are processed in parallel, which results in fewer cluster events. Therefore, there is a limited choice of events for which you can create pre- and post-event scripts. In this case, if you have resources in resource groups that *require* handling by pre- and post-event scripts written for specific cluster events, include these resource groups in the serial processing lists in SMIT, to ensure that specific pre- and post-event scripts can be used for these resources. For information about specifying serial or parallel processing of resource groups, see the section Configuring Processing Order for Resource Groups in Chapter 4: Configuring HACMP Resource Groups (Extended) in the *Administration Guide*.
You upgrade to HACMP 4.5 or higher and choose parallel processing for some of the pre-existing resource groups in your configuration.	If, before migration you had configured customized pre- or post-event scripts in your cluster, then now that these resource groups are processed in parallel after migration, the event scripts for a number of events cannot be utilized for these resource groups, since these events do *not* occur in parallel processing. If you want existing event scripts to *continue* working for the resource groups, include these resource groups in the serial ordering lists in SMIT, to ensure that the pre- and post-event scripts can be used for these resources. For information about specifying serial or parallel processing of resource groups, see the section Configuring Processing Order for Resource Groups in Chapter 4: Configuring HACMP Resource Groups (Extended) in the *Administration Guide*.

Dependent Resource Groups and Pre- and Post-Event Scripts

Prior to HACMP 5.2, to achieve resource group and application sequencing, system administrators had to build the application recovery logic in their pre- and post-event processing scripts. Every cluster would be configured with a pre-event script for all cluster events, and a post-event script for all cluster events.

Such scripts could become all-encompassing case statements. For instance, if you want to take an action for a specific event on a specific node, you need to edit that individual case, add the required code for pre- and post-event scripts, and also ensure that the scripts are the same across all nodes.

To summarize, even though the logic of such scripts captures the desired behavior of the cluster, they can be difficult to customize and even more difficult to maintain later on, when the cluster configuration changes.

If you are using pre-and post-event scripts or other methods, such as resource group processing ordering to establish dependencies between applications that are supported by your cluster, then these methods *may no longer be needed or can be significantly simplified*. Instead, you can specify dependencies between resource groups in a cluster. For more information on planning dependent resource groups, see Resource Group Dependencies in Chapter 6: Planning Resource Groups.

If you have applications included in dependent resource groups and still plan to use pre- and post-event scripts in addition to the dependencies, additional customization of pre- and post-event scripts may be needed. To minimize the chance of data loss during the application stop and restart process, customize your application server scripts to ensure that any uncommitted data is stored to a shared disk temporarily during the application stop process and read back to the application during the application restart process. It is important to use a shared disk as the application may be restarted on a node other than the one on which it was stopped.

For information on how to configure resource group dependencies, see the *Administration Guide.*

Event Recovery and Retry

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

For example, a recovery command could include the retry of unmounting a file system after logging a user off, and making sure no one was currently accessing the file system.

If a condition that affects the processing of a given event on a cluster is identified, such as a timing issue, you can insert a recovery command with a retry count high enough to be sure to cover for the problem.

Custom Remote Notification of Events

You can define a notification method through the SMIT interface to issue a customized page in response to a cluster event. In HACMP 5.4.1, you can send text messaging notification to any number including a cell phone, or mail to an email address.

You can use the verification automatic monitoring **cluster_notify** event to configure an HACMP Remote Notification Method to send out a message in case of detected errors in cluster configuration. The output of this event is logged in **hacmp.out** throughout the cluster on each node that is running cluster services.

You can configure any number of notification methods, for different events and with different text or numeric 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.

After configuring the notification method, you can send a test message to make sure everything is configured correctly and that the expected message will be sent for a given event.

Planning for Custom Remote Notification

Remote notification requires the following conditions:

- A tty port used for paging cannot also be used for heartbeat traffic or for the DBFS function of HAGEO.
- Any tty port specified must be defined to AIX and must be available.
- Each node that may send a page or text messages must have an appropriate modem installed and enabled.

Note: HACMP checks the availability of the tty port when the notification method is configured and before a page is issued. Modem status is *not* checked.

- Each node that may send email messages from the SMIT panel using AIX mail must have a TCP/IP connection to the Internet.
- Each node that may send text messages to a cell phone must have an appropriate Hayes-compatible dialer modem installed and enabled.
- Each node that may transmit an SMS message wirelessly must have a Falcom-compatible GSM modem installed in the RS232 port with the password disabled. Ensure that the modem connects to the cell phone system.

For information about how to configure a remote notification method, see Chapter 5: Configuring Cluster Events in the *Administration Guide*.

Customizing Event Duration Time Until Warning

Depending on cluster configuration, the speed of cluster nodes and the number and types of resources that need to move during cluster events, certain events may take different time intervals to complete. For such events, you may want to customize the time period HACMP waits for an event to complete before issuing the **config_too_long** warning message.

Cluster events that include acquiring and releasing resource groups take a longer time to complete. They are considered *slow* events and include the following:

- **node_up**
- **node_down**
- **reconfig_resource**
- **rg_move**
- **site_up**
- **site_down.**

Customizing event duration time for *slow* cluster events lets you avoid getting unnecessary system warnings during normal cluster operation.

All other cluster events are considered *fast* events. These events typically take a shorter time to complete and do *not* involve acquiring or releasing resources. Examples of fast events include:

- **swap_adapter**
- events that do *not* handle resource groups.

Customizing event duration time before receiving a warning for fast events allows you to take corrective action faster.

Consider customizing **Event Duration Time Until Warning** if, in the case of slow cluster events, HACMP issues warning messages too frequently; or, in the case of fast events, you want to speed up detection of a possible problem event.

Note: Dependencies between resource groups offer a predictable and reliable way of building clusters with multi-tier applications. However, processing of some cluster events (such as **node_up**) in clusters with dependencies could take more time than processing of those events where all resource groups are processed in parallel. Whenever resource group dependencies allow, HACMP processes multiple non-concurrent resource groups in parallel, and processes multiple concurrent resource groups on all nodes at once. However, a resource group that is dependent on other resource groups cannot be started until the others have been started first. The **config_too_long** warning timer for **node_up** events should be set large enough to allow for this.

For information about how to change the event duration before receiving a system warning, see the section Tuning Event Duration Time Until Warning in Chapter 5: Configuring Cluster Events in the *Administration Guide*.

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.

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 IBM RSCT documentation.

Note: The RSCT Event Management subsystem has been replaced by the Resource Monitoring and Control (RMC) subsystem in HACMP 5.2 and up. See the chapter on Upgrading an HACMP Cluster for more information.

An RMC *resource* refers to an instance of a physical or logical entity that provides services to some other component of the system. The term resource is used very broadly to refer to software as well as hardware entities. For example, a resource could be a particular file system or a particular host machine. A *resource class* refers to all resources of the same type, such as processors or host machines.

A *resource manager* (daemon) maps actual entities to RMC's abstractions. Each resource manager represents a specific set of administrative tasks or system functions. The resource manager identifies the key physical or logical entity types related to that set of administrative tasks or system functions, and defines resource classes to represent those entity types.

For example, the Host resource manager contains a set of resource classes for representing aspects of a individual host machine. It defines resource classes to represent:

- Individual machines (IBM.Host)
- Paging devices (IBM.PagingDevice)
- Physical volumes (IBM.PhysicalVolume)
- Processors (IBM.Processor)
- A host's identifier token (IBM.HostPublic)
- Programs running on the host (IBM.Program)
- Each type of adapter supported by the host, including ATM adapters (IBM.ATMDevice), Ethernet adapters (IBM.EthernetDevice), FDDI adapters (IBM.FDDIDevice), and Token Ring adapters (IBM.TokenRingDevice).

The AIX *resource monitor* generates events for OS-related resource conditions such as the percentage of CPU that is idle (`IBM.Host.PctTotalTimeIdle)` or percentage of disk space in use (`IBM.PhysicalVolume.PctBusy`). The *program resource monitor* generates events for process-related occurrences such as the unexpected termination of a process. It uses the resource attribute `IBM.Program.ProgramName.`

Note that you cannot use the Event Emulator to emulate a user-defined event.

Writing Recovery Programs

A recovery program has a sequence of recovery command specifications, possibly interspersed with **barrier** commands.

The format of these specifications follows:

```
:node_set recovery_command expected_status NULL
```

Where:

- *node_set* is a set of nodes on which the recovery program is to run
- *recovery_command* is a quote-delimited string specifying a full path to the executable program. The command cannot include any arguments. Any executable program that requires arguments must be a separate script. The recovery program must be in this path on all nodes in the cluster. The program must specify an exit status.
- *expected_status* is an integer status to be returned when the recovery command completes successfully. The Cluster Manager compares the actual status returned to the expected status. A mismatch indicates unsuccessful recovery. If you specify the character X in the expected status field, the Cluster Manager omits the comparison.
- *NULL* is *not* used now, but is included for future functions.

You specify node sets by dynamic relationships. HACMP supports the following dynamic relationships:

- *all—t*he recovery command runs on all nodes in the current membership.
- *event*—the node on which the event occurred.
- *other*—all nodes except the one on which the event occurred.

The specified dynamic relationship generates a set of recovery commands identical to the original, except that a node id replaces *node_set* in each set of commands.

The command string for user defined event commands must start with a slash (/). The **clcallev** command runs commands that do *not* start with a slash.

Useful Commands and Reference for RMC Information

To list all persistent attribute definitions for the `IBM.Host` RMC resource (*selection string* field):

```
lsrsrcdef -e -A p IBM.Host
```

To list all dynamic attribute definitions for the `IBM.Host` RMC resource (*Expression* field):

```
lsrsrcdef -e -A d IBM.Host
```

See Chapter 3, Managing and Monitoring Resources Using RMC and Resource Managers in the *IBM RSCT for AIX and Linux Administration Guide* for more information on the SQL like expressions used to configure user defined events selection strings.

Recovery Program Example

A sample program to send a message to **/tmp/r1.out** that paging space is low on the node where the event occurred. For recovery program **r1.rp**, the SMIT fields would be filled in as follows:

Event Name	`E_page_space(User-defined name)`
Recovery program path	`/r1.rp`
Resource name	`IBM.Host (cluster node)`
Selection string	`Name = ?″%″ (name of node)`
Expression	`TotalPgSpFree < 256000` (VMM is within 200 MB of paging space warning level). The resource attribute plus the condition you want to flag.
Rearm expression	`TotalPgSpFree >256000` The resource attribute plus the adjusted condition.

Where recovery program **r1.rp** is as follows:

```
#format:
#relationship  >command to run  >expected status NULL
#
event  "/tmp/checkpagingspace"  0 NULL
```

Note that the recovery program does *not* execute a command with arguments itself. Instead, it points to a shell script, **/tmp/checkpagingspace**, which contains:

```
#!/bin/ksh
/usr/bin/echo “Paging Space LOW!” > /tmp/r1.out
exit 0
```

Recovery Program for node_up Event Example

The following example is a recovery program for the **node_up** event:

```
#format:
#relationship  command to run  expected status NULL
#
other  "node_up"  0 NULL
#
barrier
#
event "node_up" 0 NULL
#
barrier
#
all "node_up_complete" X NULL
```

Barrier Commands

You can put any number of barrier commands in the recovery program. All recovery commands before a barrier start in parallel. Once a node encounters a barrier command, all nodes must reach it before the recovery program continues.

The syntax of the barrier command is **barrier**.

Event Roll-up

If there are multiple events outstanding simultaneously, you only see the highest priority event. Node events are higher priority than network events. But user-defined events, the lowest priority, do *not* roll up at all, so you see all of them.

Event Summaries and Preamble

When events are logged to a node's **hacmp.out** log file, the verbose output contains numerous lines of event details followed by a concise event summary. The event summaries make it easier to scan the log for important cluster events.

You can view a compilation of just the event summary portions of the past seven days' **hacmp.out** files by using the **View Event Summaries** option in the **Problem Determination Tools** SMIT panel. The event summaries can be compiled even if you have redirected the **hacmp.out** file to a non-default location. The **Display Event Summaries** report also includes resource group information generated by the **clRGinfo** command. You can also save the event summaries to a specified file instead of viewing them through SMIT.

When events handle resource groups with dependencies or sites, a preamble is written to the **hacmp.out** log file listing the plan of sub_events for handling the resource groups.

For more information on this file, see the section Event Summaries in Chapter 2: Using Cluster Log Files in the *Troubleshooting Guide*.

Completing the Cluster Event Worksheet

The Cluster Event Worksheet helps you plan the cluster event processing for your cluster. Appendix A: Planning Worksheets includes the worksheet referenced in the following procedure.

For more information about cluster events, see Chapter 5: Configuring Cluster Events in the *Administration Guide*.

For each node in the cluster, repeat the steps in the following procedure on a separate worksheet. To plan the customized processing for a specific cluster event, enter a value in the fields only as necessary:

1. Record the cluster name in the **Cluster Name** field.
2. Record the custom cluster event description in the **Cluster Event Description** field.
3. Record the full pathname of the cluster event method in the **Cluster Event Method** field.
4. Fill in the name of the cluster event in the **Cluster Event Name** field.
5. Fill in the full pathname of the event script in the **Event Command** field.
6. Record the full pathname of the event notification script in the **Notify Command** field.
7. Record the full text of the remote notification method in **Remote Notification Message Text.**

8. Record the full pathname of the file containing the remote notification method text in **Remote Notification Message Location**.
9. Record the name of the pre-event script in the **Pre-Event Command** field.
10. Record the name of the post-event script in the **Post-Event Command** field.
11. Record the full pathname of the event retry script in the **Event Recovery Command** field.
12. Indicate the number of times to retry in the **Recovery Command Retry** field.
13. Record the time to allot to process an event for a resource group before a warning is displayed in the **Time Until Warning** field.

    Repeat steps 3 through 13 for each event you plan to customize.

Where You Go from Here

You have now planned the customized and user-defined event processing for your cluster. You will next address cluster clients issues, described in Chapter 8: Planning for HACMP Clients.

Chapter 8: Planning for HACMP Clients

This chapter discusses planning considerations for HACMP clients. This is the last step before proceeding to installation of your HACMP software. This chapter contains the following sections:

- Prerequisites
- Overview
- Clients Running Clinfo
- Clients Not Running Clinfo
- Network Components
- Where You Go from Here

Prerequisites

By now, you should have completed the planning steps in the previous chapters:

- Chapter 2: Initial Cluster Planning
- Chapter 3: Planning Cluster Network Connectivity
- Chapter 4: Planning Shared Disk and Tape Devices
- Chapter 5: Planning Shared LVM Components
- Chapter 6: Planning Resource Groups
- Chapter 7: Planning for Cluster Events

Overview

HACMP *clients* are end-user devices that can access the nodes in an HACMP cluster. In this step of the planning process, you evaluate the cluster from the point of view of the clients.

HACMP clients may contain hardware and software from a variety of vendors. To maintain connectivity to the HACMP cluster, consider the issues in the following sections.

Client Application Systems

All clients should run Clinfo if possible. If you have hardware other than IBM System p™ nodes in the configuration, you may want to port Clinfo to those platforms. Clinfo source code is provided as part of HACMP.

You need to think about what applications are running on these clients. Who are the users? Is it required or appropriate for users to receive a message when cluster events affect their system?

NFS Servers

For information about NFS-related issues, see the section Using NFS with HACMP in Chapter 5: Planning Shared LVM Components.

Terminal Servers

If you plan to use a terminal server on the local area network, consider the following when choosing the hardware:

- Can you update the terminal server's ARP cache? The terminal server must comply with the TCP/IP protocol, including Telnet.
- Is the terminal server programmable, or does it need manual intervention when a cluster event happens?
- Can you download a control file from the cluster node to the terminal server that updates or handles cluster events' effects on clients?

If your terminal server does *not* meet these operating requirements, choose the hardware address swapping option when configuring the cluster environment.

Clients Running Clinfo

The Clinfo program calls the **/usr/es/sbin/cluster/etc/clinfo.rc** script whenever a network or node event occurs. By default, this action updates the system's ARP cache to reflect changes to network addresses. You can customize this script if further action is desired.

Reconnecting to the Cluster

Clients running the Clinfo daemon will be able to reconnect to the cluster quickly after a cluster event. If you have hardware other than IBM System p™ between the cluster and the clients, make sure that you can update the ARP cache of those network components after a cluster event occurs.

If you configure the cluster to swap hardware addresses as well as IP addresses, you do *not* need to be concerned about updating the ARP cache. However, be aware that this option causes a longer delay for the users.

If you are using IPAT via IP Aliases, make sure all your clients support TCP/IP gratuitous ARP.

Customizing the clinfo.rc Script

For clients running Clinfo, decide whether to customize the **/usr/es/sbin/cluster/etc/clinfo.rc** script to do more than update the ARP cache when a cluster event occurs. For more information, see the chapter on Installing HACMP on Client Nodes in the *Installation Guide*. Also see the *Programming Client Applications* guide for a sample **clinfo.rc** script.

Clients *Not* Running Clinfo

On clients *not* running Clinfo, you may have to update the local ARP cache indirectly by pinging the client from the cluster node. On the cluster nodes, add the name or address of a client host you want to notify to the **PING_CLIENT_LIST** variable in the **clinfo.rc** script. When a cluster event occurs, **clinfo.rc** runs the following command for each host specified in PING_CLIENT_LIST:

```
ping -c1 $host
```

This assumes the client is connected directly to one of the cluster networks.

Network Components

If you configured the network so that clients attach to networks on the other side of a router, bridge, or gateway rather than to the cluster's local networks, be sure that you can update the ARP cache of those network components after a cluster event occurs. If this is *not* possible, then make sure to use hardware address swapping when you configure the cluster environment.

Where You Go from Here

This chapter concludes the planning process. You can now begin to install the HACMP software. For detailed instructions, see the *Installation Guide*.

Chapter 9: Using Online Planning Worksheets

This chapter describes how to use the Online Planning Worksheets (OLPW) application, which creates a *cluster definition file* (also sometimes referred to as a *worksheets* file). This chapter also describes the cluster definition file and how to use it to define your HACMP cluster.

A cluster definition file contains planning information that can be applied to your cluster to configure it for HACMP. There are several ways to create a cluster definition file, depending on the state of your cluster:

- You can use the Online Planning Worksheets to create a new file from scratch or from an existing file as described in the section Using the Online Planning Worksheets Application in this chapter.
- If you prefer to manually edit the cluster definition file, you can edit it using an XML editor or plain text editor. See Understanding the Cluster Definition File in this chapter.
- If you have an existing HACMP snapshot, you can use SMIT to convert the snapshot information into a cluster definition file as described in the section Converting a Snapshot to a Cluster Definition File in this chapter.
- If you have an existing HACMP installation, you can use SMIT to export your configuration as a cluster definition file as described in the section Recovering Planning Worksheets from a Working Cluster: Exporting a Definition File in this chapter. This option could be useful in cases when you are faced with an already configured cluster which you did not set up. Many times, it is necessary to support such clusters, and the planning worksheets based on which the cluster was originally configured do not exist. This function lets you recover the planning worksheets and obtain a readable, online or printed description of the cluster.
- If you have an existing HACMP installation and do *not* want to create an intermediary file, you can bring your cluster configuration information directly into OLPW as described in the section Creating a New Cluster Definition File.

After you finish populating your cluster definition with your specific configuration information, you apply the cluster definition file to your cluster as described in Applying Worksheet Data to Your HACMP Cluster.

Prerequisites

Before you use the Online Planning Worksheets application, you should understand the concepts and terminology relevant to HACMP. The *Concepts and Facilities Guide* provides this information. You should also be familiar with the planning process. While using the application, refer to the chapters earlier in this guide for more information about how to plan each component of your cluster. Also, draw a diagram of your planned configuration before you get started, as suggested in Chapter 2: Initial Cluster Planning.

For information about installation prerequisites, see the section Installing the Online Planning Worksheets Application.

Overview of the Online Planning Worksheets Application

The Online Planning Worksheets application is a Java-based application that enables you to plan your HACMP cluster configuration. Using this application, you can either import HACMP configuration information and edit it as needed, or you can enter in all configuration information manually. The application saves your information as a cluster definition file that you can apply to your cluster. Then HACMP automatically configures your cluster, based on that information. The application also validates your data to ensure that all required information has been entered.

To document your configuration, you can produce an HTML report, which indicates what you have finished planning and what remains to be completed.

Limitations

The Online Planning Worksheets application *only* opens configuration files generated by the following versions of HACMP:

- HACMP 5.4
- HACMP 5.3
- HACMP 5.2
- HACMP 5.1.0.1 (5.1 PTF 1)

The Online Planning Worksheets application *does not* support the following:

- HACMP discovery
- National Language Support
- HACMP/XD
- Custom Resource Recovery
- GPFS integration with HACMP
- Smart Assists for WebSphere, DB2, and Oracle

In addition, the application *does not* let you configure the following:

- Runtime parameters:
  - RSCT maximum log lengths
  - HACMP runtime parameters
  - Advanced performance tuning parameters
- Complex configurations that require significant AIX configuration before the HACMP configuration:
  - Customization of the distribution preferences for service IP label aliases
  - Additions or changes to NIM configuration
  - HA communication links for WAN support, SNA, or X25
  - Custom verification methods
  - User-defined events
- Customized policies for setting the Dynamic Node Priority

- Facilities that require cluster connectivity in order to function, such as functions that require C-SPOC.

After you configure a parent/child relationship for two resource groups and configure a Resource Group location dependency of type **Online on the same node Dependency**, OLPW does *not* show two resource group settings, parent/child relationship and Resource Group location dependency. When you select **Resource Group Runtime Policies**, only the following values are shown in the right box/window:

- Acquisition order / release order
- RG distribution policy
- Custom resource group recovery

Installing the Online Planning Worksheets Application

You can install and run the Online Planning Worksheets application on a Microsoft Windows or an IBM AIX system. You can also run the application directly from the IBM website.

This section contains the following topics:

- Installation Prerequisites
- User Permission Requirements
- Downloading the Application on an AIX System
- Installing the Application on a Windows System

Installation Prerequisites

The Online Planning Worksheets application requires the Java Runtime Environment (JRE) version 1.3.0 or higher.

The Online Planning Worksheets application is supported on the following operating systems:

- IBM AIX versions (which include the JRE by default):
  - AIX v.5.3
- Microsoft Windows 2000

The Online Planning Worksheets application should also run on any other system that has the appropriate JRE installed. Note that the display of fonts in the application depends on the system fonts available.

User Permission Requirements

Typically, you can run the Online Planning Worksheets application from any Windows 2000 or AIX system; however the Java Virtual Machine (JVM) must let users save data to disk from a Java application.

On an AIX system, users also require the following privileges:

- Privileges to copy files to the AIX system—for example, if the configuration file was created on another system, such as Windows
- Root privilege to apply the configuration file

Downloading the Application on an AIX System

You download the Online Planning Worksheets application from the installable image that is located at this URL:

http://www.ibm.com/systems/p/ha/ha_olpw.html

The name of the file to download is:

```
worksheets.jar
```

Once you accept the license agreement, find the online planning worksheets **worksheets.jar** file and click on it or run the following command from the AIX command line:

```
java -jar worksheets.jar
```

Install the application as you install any HACMP component.

The Online Planning Worksheets application is installed in the **/usr/es/sbin/cluster/worksheets** directory.

Note: If you move the **worksheets.jar** file from the **/usr/es/sbin/cluster/worksheets** directory, edit the **/usr/es/sbin/cluster/worksheets/worksheets** file and set the WORKSHEETS variable to the correct full path of the **worksheets.jar** file.

Moving the **/usr/es/sbin/cluster/worksheets/worksheets** file to a different location *does not* require changes to the WORKSHEETS variable.

For information on running the Online Planning Worksheets application, see Running the Application from an AIX Installation.

Installing the Application on a Windows System

To install the Online Planning Worksheets application on a Microsoft Windows system:

1. Download and install the Online Planning Worksheets on an AIX system from this URL:

   http://www.ibm.com/systems/p/ha/ha_olpw.html

   The name of the file to download is:

   ```
   worksheets.jar
   ```
   For information about downloading and installing the application on AIX, see the section Downloading the Application on an AIX System.

2. Copy the **worksheets.bat** and **worksheets.jar** files to a directory of your choice on your Windows system. If you copy the files via FTP, be sure to specify the binary mode.

3. If you do *not* have the **worksheets.bat** and **worksheets.jar** files in the same directory, edit the **worksheets.bat** file and set the WORKSHEETS variable to specify the full path of the **worksheets.jar** file.

Note: You do *not* need to edit the CLASSPATH environment variable to run the Online Planning Worksheets application.

For information on running the Online Planning Worksheets application, see Running the Application from a Windows Installation.

Starting and Stopping the Application

This section contains the following topics:

- Running the Application from an AIX Installation
- Running the Application from a Windows Installation
- Running the Application from the IBM Website

Running the Application from an AIX Installation

To run the application on an AIX system, execute the following command:

```
/usr/es/sbin/cluster/worksheets/worksheets
```

The application verifies that you have an appropriate version of the JRE installed before it runs the application in the background.

Next, see Where You Go from Here.

Running the Application from a Windows Installation

To run the application, execute the **worksheets.bat** command from the command line.

Next, see Where You Go from Here.

Running the Application from the IBM Website

You can run the Online Planning Worksheets Application directly from the following IBM URL:

http://www.ibm.com/systems/p/ha/ha_olpw.html

The name of the file to download is:

```
worksheets.jar
```

Once you accept the license agreement, find the online planning worksheets **worksheets.jar** file and click on it or run the following command from the AIX command line:

```
java -jar worksheets.jar
```

Stopping the Application

To stop the application, select **File** > **Exit**.

Where You Go from Here

If you are new to Online Planning Worksheets and would like an orientation on navigating the user interface and entering data, see the section Using the Online Planning Worksheets Application.

If you are ready to begin using Online Planning Worksheets to fill in your configuration information, see section Planning a Cluster.

If you prefer to fill in your cluster definition using an XML editor, see section Understanding the Cluster Definition File.

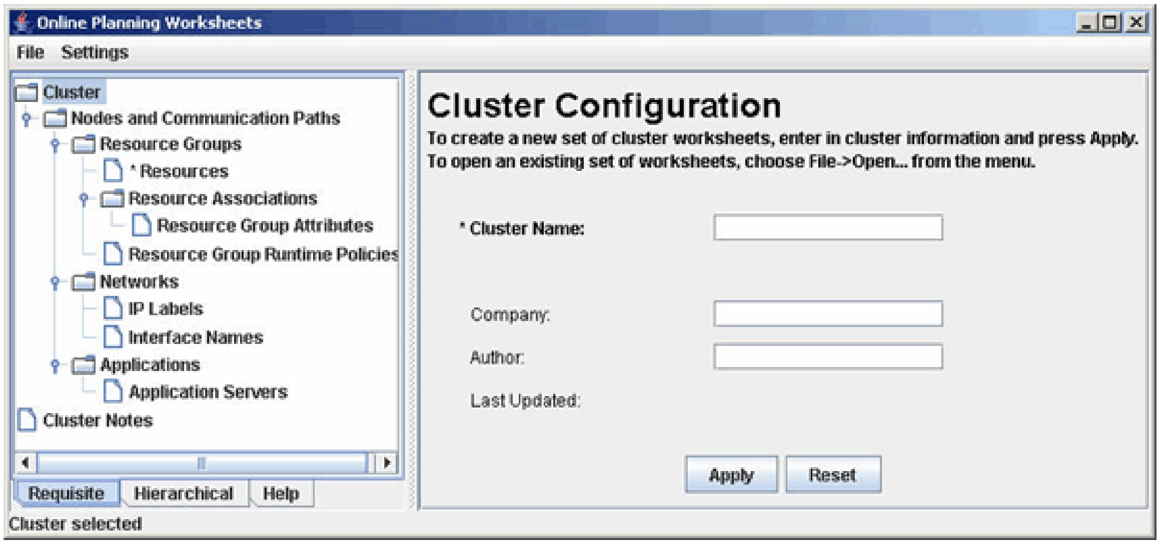
Using the Online Planning Worksheets Application

This section describes the following Online Planning Worksheets tasks:

- Understanding the Main Window
- Navigating Your Cluster Configuration
- Viewing Standard and Extended Configuration Panels
- Entering Data in the Configuration Panels
- Opening an Existing Cluster Definition File
- Creating a New Cluster Definition File
- Adding, Modifying, and Deleting Cluster Information
- Adding Notes about the Cluster Configuration
- Validating a Cluster Definition
- Saving a Cluster Definition File
- Creating an HTML Configuration Report
- Using Online Help

Understanding the Main Window

When the Online Planning Worksheets application opens, its main window displays as follows.



Cluster Configuration

To create a new set of cluster worksheets, enter in cluster information and press Apply.
To open an existing set of worksheets, choose File->Open... from the menu.

* Cluster Name:

Company:

Author:

Last Updated:

Cluster selected

Figure 17. Online Planning Worksheets Main Window

The main window consists of two panes:

- The left pane enables navigation to cluster components
- The right pane displays panels associated with icons selected in the left pane. These panels are where you enter your configuration information.

Navigating Your Cluster Configuration

The left pane provides three tabbed views that enable you to navigate your cluster configuration:

- Requisite. Use this view to enter cluster information. The items in this view are organized in the sequence recommended for configuring cluster components. The items appear in a logical sequence showing dependencies among some configuration items. For example, *Nodes* appears before *Resource Groups*, because you identify the nodes in a cluster before defining a resource group to include them.
- Hierarchical. Use this to view logical groupings of your cluster configuration.
- Help. Use this view to list the Help topics.

Note: Icons in the navigation pane that display an asterisk (*) contain the navigational items beneath them; they do *not* display a configuration panel.

The following figure shows the difference between the organization of items in the Requisite view and the Hierarchical view.

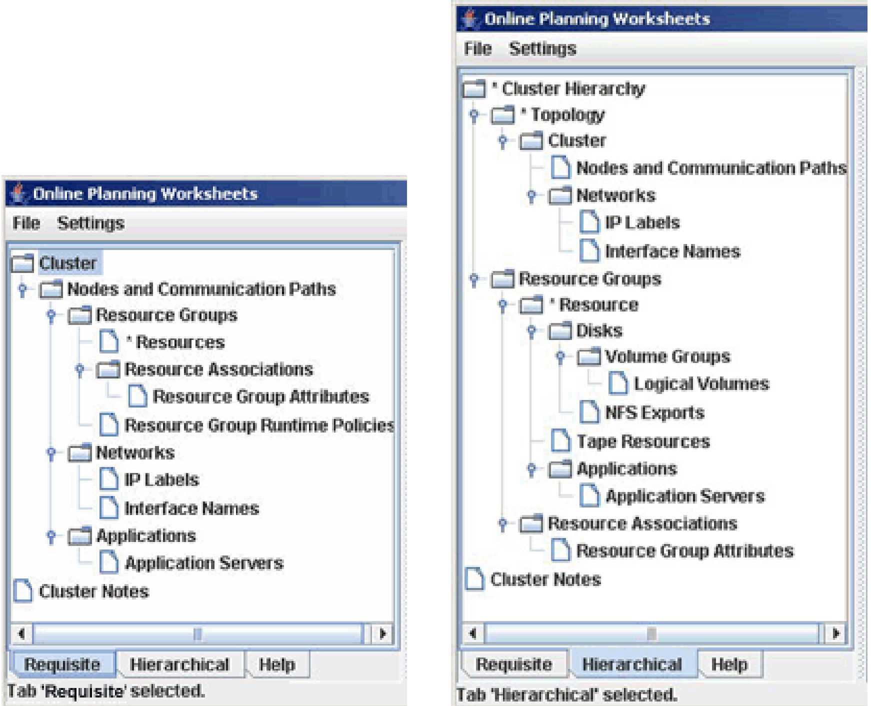


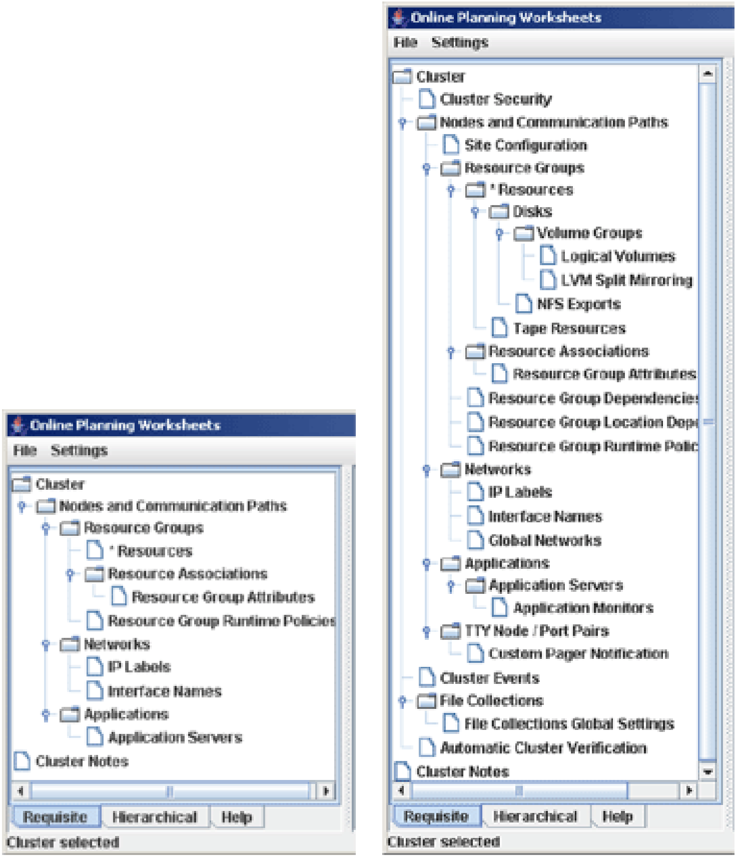
Figure 18. Requisite View and Hierarchical View

When you click a tab at the bottom of the left pane to switch views, the same item remains selected in the left pane. This enables you to easily switch context for a selected item (for example, to view online help).

Viewing Standard and Extended Configuration Panels

You can display the standard set of configuration panels or the extended (full) set of configuration panels. These settings are similar to the SMIT standard and extended configuration options. Select the set of panels to display from the **Settings** menu.

The standard set of configuration panels enables you set up a basic HACMP configuration. The following figure shows the standard and extended Requisite views:



Standard Configuration View

Extended Configuration View

Figure 19. Standard and Extended Views

Entering Data in the Configuration Panels

The configuration panels contain groups of fields for you to enter information about your cluster. You can begin entering data immediately. As you enter data, the application validates the syntax of the information. Note that the lack of cluster connectivity at the planning stage limits verification of the accuracy of the information entered. For example, the application does *not* validate that the value for an IP label assigned to an interface is accurate.

The application also provides a number of panels for entering information about your HACMP cluster for documentation purposes:

- Disks
- Volume Groups
- Logical Volumes
- Network File System Exports
- Applications.

Information that you enter in these panels may be available from other panels. For example if you specify an application before configuring an application server. This information is included in any reports you generate, but is *not* used in the configuration file.

Entering Data into Required and Optional Fields

Fields in the configuration panels are either required or optional:

- *Required fields*. These fields must be filled in before you apply the configuration to your cluster. A required field appears in bold-face type.

  If you leave a required field blank, when you save the file, a message indicates which remaining fields must be completed. You can save this validation message as a log file.
- *Optional fields*. These fields are *not* required to run HACMP but their data are included in the Online Planning Worksheets reports. An optional field appears in regular type.

Using Online Help

While entering information in the configuration panels, refer to the online Help, available from the **Help** tab, to get information about:

- A description of the current panel
- Button actions.

The Help view provides information about the panel selected in the Requisite or Hierarchical view.

You can display the following topics from the index link at the top of each Help panel:

- About Online Planning Worksheets
- Menus
- Navigation.

Creating a New Cluster Definition File

When you start planning your cluster, you can fill in all information by hand, OLPW can read in your cluster configuration information. To import the HACMP definition, run OLPW from the cluster node. You can also create a cluster definition file from an active HACMP cluster.

Note: Importing the HACMP definition is *not* supported from a Windows 2000 machine; the menu option will be disabled.

To create a cluster definition file:

1. Enter all data by hand, following the procedures described in section Planning a Cluster.

   or

2. Read in configuration information directly from your HACMP cluster as follows:

   a. Select **File > Import HACMP Definition**.

      The Import Validation dialog box appears. You can view information about validation errors or receive notification that the validation of the HACMP definition file was successful.

   b. Enter additional data by hand, following the procedures described in section Planning a Cluster.

3. To create a cluster definition file from an active HACMP cluster, see the section Recovering Planning Worksheets from a Working Cluster: Exporting a Definition File and open the file as described in the section Opening an Existing Cluster Definition File.

Opening an Existing Cluster Definition File

Note: The cluster definition file must reside on the same node running the Online Planning Worksheets application.

The Online Planning Worksheets application supports opening cluster definition files with the following file extensions:

.haw This is the preferred extension. It is supported in HACMP 5.4.1.

.xml This extension may be used. It is supported in HACMP 5.4.1.

.ws This file format is supported in HACMP 5.1.0.1. For backwards compatibility, **.ws** files can be opened in HACMP 5.4.1; however, they must be saved with either the **.xml** or **.haw** file extension. When you save a file, the Save dialog box specifies the **.xml** or **.haw** file extensions, as shown in section Saving a Cluster Definition File.

Files with the .**ws** extension created using a version of the Online Planning Worksheets application prior to HACMP 5.1.0.1 are *not* supported.

To open a cluster definition file, select **File > Open**.

Only one cluster definition file can be open at a time. If a cluster definition file is open and you want to start a new one, select **File > New**. You will be prompted to save the current file, whether or *not* you have made any modifications, and the main window appears with no configuration information.

Adding, Modifying, and Deleting Cluster Information

When you enter information and then select **Add**, the information appears in the list at the bottom of the panel. You can then enter information for another item.

You can select items in the list at the bottom of the dialog box, and then modify information for the item or delete the line. Changes that you make may affect entries in other panels that use the information. For example, if you delete a resource group, other panels that reference that resource group no longer provide the name of the deleted resource group.

When you select **Add** or **Modify**, the application validates the syntax of the information entered in a field. A message appears to provide information about any syntax errors.

Adding Notes about the Cluster Configuration

As you plan your configuration, you can add notes to save information that may be helpful to you at another time. To add cluster notes:

1. In the left pane in either the Requisite view or the Hierarchical view, select **Cluster Notes**.
2. In the Cluster Notes panel, enter the information you want to save.
3. Select **Apply**.

Validating a Cluster Definition

Validation checks that the information specified is complete and meets the criteria for the parameter (for example, a name limit of 32 characters). By default, Online Planning Worksheets runs a validation when:

- Importing a cluster definition from an active cluster
- Opening a cluster definition file
- Saving a cluster definition file.

During the cluster planning phase, you may want to turn off validation to avoid being prompted to validate the cluster definition each time you save the file.

To turn off automatic validation of a cluster definition stored in a cluster definition file, select **Settings > Validate When Saving** to clear the check mark next to the menu item.

You can also validate a cluster definition file at any time.

To validate the information in a cluster definition file, select **File > Validate HACMP Definition**.

Saving a Cluster Definition File

To save a cluster definition file:

- Select **File > Save** to use your cluster name as the filename.

  or

- Select **File > Save As** to enter a different filename.

  In the Save dialog box, enter the name and location for your cluster definition file, make sure that the filename has the **.haw** (or **.xml**) extension, and click **Save**. Online Planning Worksheets saves the cluster definition file.

Figure 20. Save Dialog Box

When you save a file, OLPW automatically validates the cluster definition unless automatic validation has been turned off. For information about cluster validation, see the section Validating a Cluster Definition.

When saving a cluster definition file, OLPW does *not* save information about components that are *not* supported. For example, X.25 and SNA links are *not* supported, so any information pertaining to them will *not* be saved.

Creating an HTML Configuration Report

A configuration report enables you to record information about the state of your cluster configuration in an HTML file. To help you evaluate your configuration, fields on empty panels and required fields that do *not* have a value on other panels are marked as `Undefined`.

You can create a report during the planning process. After you apply the configuration definition file, generate a report to record the initial configuration of HACMP.

A report provides summary information that includes:

- The name of the directory that stores images used in the report
- The version of the Online Planning Worksheets application
- The author and company specified on the Cluster Configuration panel
- Cluster notes added from the Cluster Notes panel
- The latest date and time that Online Planning Worksheets saved the cluster definition file.

The report also provides a section for each of the following:

- Nodes and communication paths
- Networks
- IP labels
- Applications
- NFS exports
- Application servers

- Nodes and communication paths
- Global network
- Sites
- Disks
- Resource groups
- Volume groups
- Logical volumes
- File collections
- Cross-site LVM Mirroring
- Applications
- Application monitors
- Pagers or cell phones
- Remote notifications
- Tape resources
- Resource group runtime policies
- Node summary
- Cluster verification

Note: For Disks, Volume Groups, Logical Volumes, and NFS Exports resources, the only items listed will be ones entered using OLPW in these sections. The other resources can come either from OLPW or from exporting the cluster definition file.

To create a configuration definition report:

1. Select **File > Create Report**.
2. In the Save dialog box, enter a name and location for the report file.

   When a report is generated, a directory named **olpwimages** is created in the same directory that stores the report. For example, if you save your report file to the directory **/home/pat/reports**, the graphics directory is **/home/pat/reports/olpwimages**. The **olpwimages** directory contains graphics files associated with the report.

   Each time you generate a report, the report and files in the images directory are replaced.

Note: The generated report files and their associated **olpwimages** files remain on your system. You must manually manage these files.

The following illustration shows part of a report for a four-node cluster:

Figure 21. Sample Report

Planning a Cluster

When you plan your cluster, use the Requisite view. This view guides you through the planning panels in a logical sequence. After you enter data in a panel, fields in subsequent panels may display the previously entered information.

Overview of the Planning Process

To plan an HACMP configuration:

1. From the Requisite view, select **Cluster**.
2. In the Cluster Configuration panel, specify the **Cluster Name** and any other values, and press **Apply**.

3. Select **Nodes and Communication Paths** and enter configuration information in the corresponding panel.
4. Continue selecting items in the Requisite view, and providing configuration information in the associated right panel.
5. Save the planning and configuration information periodically.

When you complete the planning process:

1. Save the configuration file. See the section Saving a Cluster Definition File.
2. Apply it to a cluster. See the section Applying Worksheet Data to Your HACMP Cluster.
3. Create a report to save information about the initial cluster configuration. See the section Creating an HTML Configuration Report.

The order of the following sections is the same as the items listed in the Requisite view. These sections briefly describe the type of configuration information you enter and provide references to other sections in this book that supply more information about the configuration options. Text identifies which panels are available in the extended configuration view.

Defining the Cluster

In the **Cluster Configuration** panel, provide a name for the cluster. It is a good idea to provide the author's name. The application displays the date the cluster definition file was last updated.

Defining Cluster Security

In the **Cluster Security** panel, select the **Connection Authentication Mode** and **Message Authentication Mode and Key Management** for communications within the HACMP cluster. If the cluster includes SP systems that use Kerberos security, consider selecting **Extended Security**. For information about standard and extended security for an HACMP cluster, see Planning Cluster Security in Chapter 2: Initial Cluster Planning.

Adding Nodes to the Cluster

In the **Nodes and Communication Paths** panel, specify the nodes to add to your HACMP cluster and the communication path to connect to that node. The communication path can be an IP label/address or a fully-qualified domain name.

Planning Site Configuration

In the **Site Configuration** panel, specify information for a multiple site configuration. Configurations that will support HACMP/XD require more than one HACMP site.

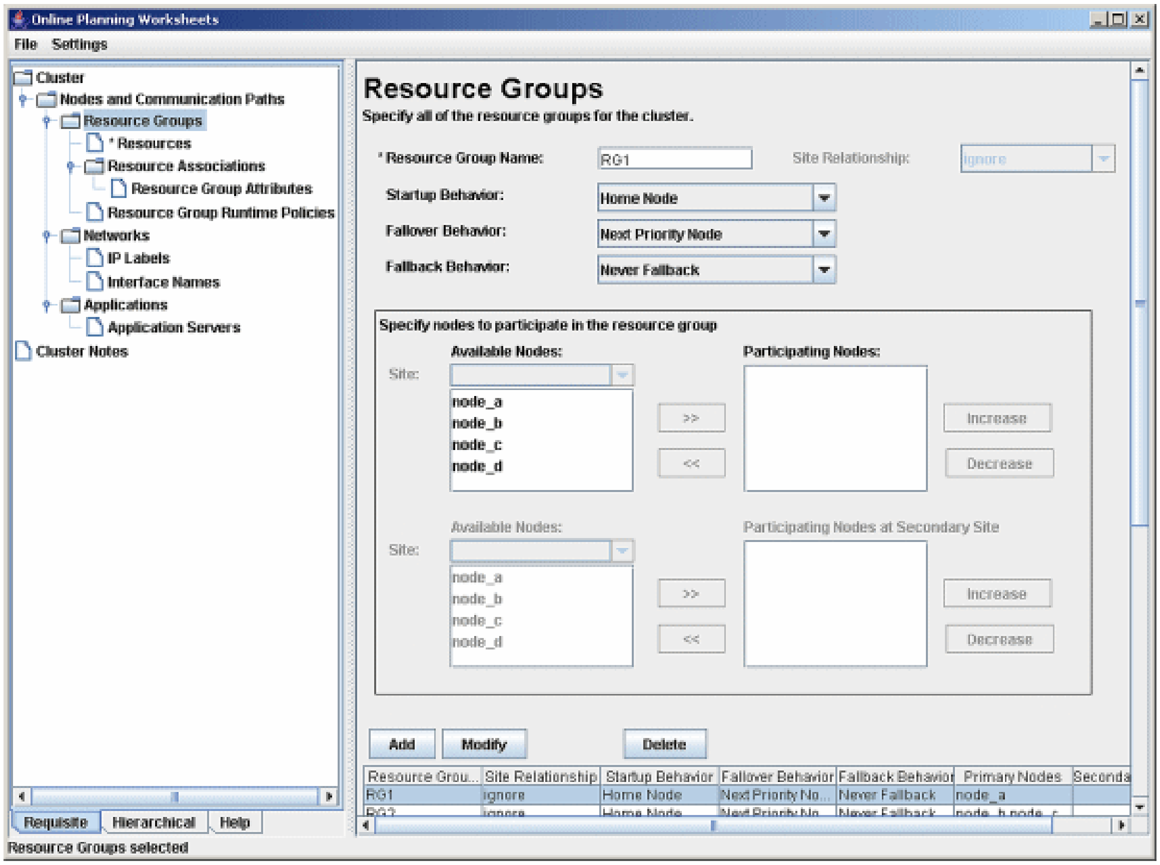
For information about planning for sites, see Chapter 2: Initial Cluster Planning.

Creating Resource Groups

In the **Resource Groups** panel, assign names to your resource groups and specify the management policies for each resource group.

You can define a **Site Relationship** for resource groups and assign nodes to a site after you complete the Site Configuration panel (Display Extended Config panels). A node may be assigned to only one site.

The following figure shows configuration in progress for a new resource group:



Resource Groups

Specify all of the resource groups for the cluster.

* Resource Group Name: RG1
Site Relationship: ignore
Startup Behavior: Home Node
Failover Behavior: Next Priority Node
Fallback Behavior: Never Fallback

Specify nodes to participate in the resource group

Available Nodes: node_a node_b node_c node_d
Participating Nodes:

Available Nodes: node_a node_b node_c node_d
Participating Nodes at Secondary Site

Resource Grou...	Site Relationship	Startup Behavior	Failover Behavior	Fallback Behavior	Primary Nodes	Seconda
RG1	ignore	Home Node	Next Priority No...	Never Fallback	node_a	
RG2	ignore	Home Node	Next Priority No	Never Fallback	node_b node_c	

Resource Groups selected

Figure 22. Resource Groups Panel

For information about resource groups, see Chapter 6: Planning Resource Groups.

From the Settings menu, you can select either **Standard RG Config Options** or **Extended RG Config Options**. On the Resource Associations panel, each view displays different types of resources listed in **Resource Type**. The resource types listed from the **Extended RG Config Options** show all of the types of resources that are visible from the extended configuration panels. Otherwise, the two views are the same.

Defining Disk Resources

In the set of panels under Resources, specify information about disks and file systems to be included in your cluster. These panels are available in the Display Extended Config Panels.

Disks

(*Display Extended Config Panels*) In the Disks panel, enter information about the disks you want to use in your cluster.

For information about planning for disks, see Chapter 4: Planning Shared Disk and Tape Devices.

Volume Groups

(*Display Extended Config Panels*) In the Volume Groups panel, specify the volume groups for your cluster.

For information about planning for volume groups, see Chapter 5: Planning Shared LVM Components.

Logical Volumes

(*Display Extended Config Panels*) In the Logical Volumes panel, specify logical volumes and their associated volume groups for your cluster.

For information about planning for logical volume groups, see Chapter 5: Planning Shared LVM Components. For additional information, see the *AIX System Management Guide: Operating System and Devices*.

Cross-Site LVM Mirroring

(*Display Extended Config Panels*) In the Cross-Site LVM Mirroring panel, define which hdisks associated with a cluster node to assign to an HAMCP site. Before using this panel, define nodes in the Nodes and Communication Paths panel and define sites in the Site Configuration panel.

For more information about cross-site LVM mirroring, see Chapter 5: Planning Shared LVM Components.

NFS Exports

(*Display Extended Config Panels*) In the NFS Exports panel, specify which file systems, if any, are to be exported for NFS mounting by other nodes.

For information about planning for NFS exports, see Chapter 5: Planning Shared LVM Components.

Adding Tape Resources

(*Display Extended Config Panels*) In the Tape Resources panel, specify the tape resources to include in the resource group and identify the start and stop scripts for the device. Defining a tape resource here makes it available in the Resource Associations panel.

Note: A tape resource may only be included in a two-node resource group.
A resource group may only contain one tape resource.

For information about planning for tape resources, see Chapter 4: Planning Shared Disk and Tape Devices.

Adding Resource Associations

In the Resource Associations panel, specify which individual resources—such as file systems, service IP labels, volume groups, application servers, and hdisks—are to be part of each resource group. For more information, see the section Creating Resource Groups.

Create a resource group in the Resource Groups panel before you specify resource associations for the resource group. If you want to delete a resource group, you need to delete the related resource associations first.

For information about configuring associations, see Chapter 6: Planning Resource Groups.

Resource Group Attributes

In the Resource Group Attributes panel, specify information about the file system, Workload Manager, dynamic node priority, and general attributes (such as forced varyon). Create a resource group in the Resource Groups panel before you specify Resource Group Attributes for the resource group.

For information about configuring resource group attributes, see Chapter 6: Planning Resource Groups and Chapter 5: Planning Shared LVM Components.

Resource Group Dependencies

(*Display Extended Configuration Panels*) In the Resource Group Dependencies panel, specify the parent/child dependencies and click the **Add** button after the pair is entered.

For information about resource group runtime policies, see section Resource Group Location Dependencies in Chapter 6: Planning Resource Groups.

Resource Group Location Dependencies

(*Display Extended Configuration Panels*) In the Resource Group Location Dependencies panel, specify the resource group dependency type and then choose the resource groups.

For information about resource group runtime policies, see section Resource Group Location Dependencies in Chapter 6: Planning Resource Groups.

Resource Group Runtime Policies

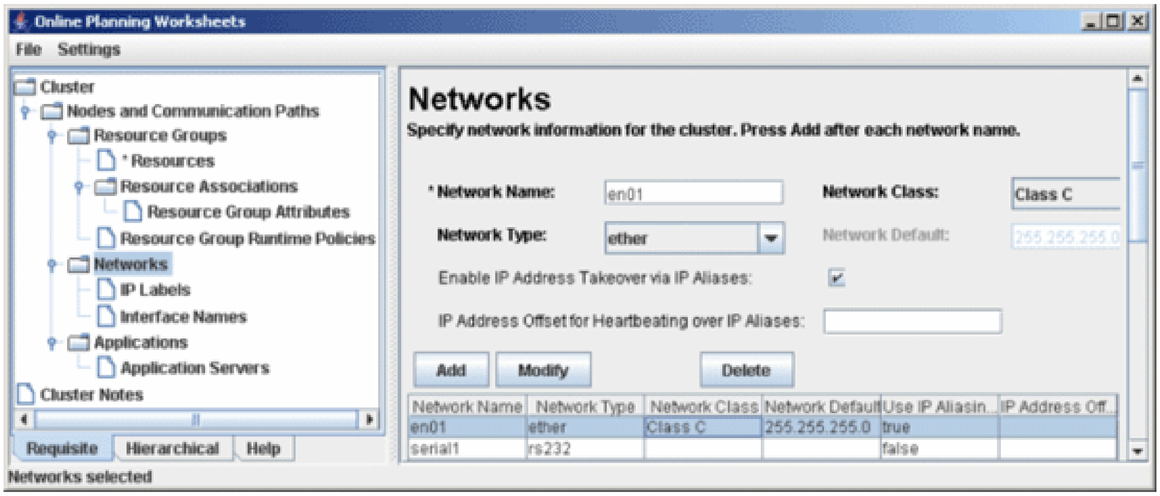
(*Display Extended Configuration Panels*) In the Resource Group Runtime Policies panel, specify the resource group processing order and the settling time, and resource group node distribution policy.

For information about resource group runtime policies, see Chapter 6: Planning Resource Groups.

Defining Networks

In the Networks panel, define the networks for your cluster including the network name and type. Information that you enter about the network is specific to the network type you select. For disk heartbeating networks, also plan which disk will be used for heartbeating. The Networks section also includes configuration panels for IP Labels, Interface Names, and in the *Display Extended Config* panels, *Global Networks*.

The following figure shows the type of information you enter in the Networks panel:



Networks

Specify network information for the cluster. Press Add after each network name.

* Network Name: en01
Network Class: Class C
Network Type: ether
Network Default: 255.255.255.0
Enable IP Address Takeover via IP Aliases: ☑
IP Address Offset for Heartbeating over IP Aliases:

Network Name	Network Type	Network Class	Network Default	Use IP Aliasin...	IP Address Off...
en01	ether	Class C	255.255.255.0	true	
serial1	rs232			false	

Networks selected

Figure 23. Networks Panel

Note: Before you can delete a node or network that includes an IP label or interface, delete the IP label, the interface, or both from the respective panel before deleting the associated node or network.

For information about planning networks, see Chapter 3: Planning Cluster Network Connectivity.

IP Labels

In the IP Labels panel, specify service IP labels and persistent labels for an interface.

For more information about assigning labels, see Chapter 3: Planning Cluster Network Connectivity and the *Concepts and Facilities Guide*.

Interface Names

In the Interface Names panel, specify names for each interface on a node.

For more information about assigning interface names, see Chapter 3: Planning Cluster Network Connectivity.

Global Networks

(*Display Extended Config Panels*) In the Global Networks panel, specify information about a set of networks that comprise a global network.

For information about global networks for HACMP, see Chapter 3: Planning Cluster Network Connectivity.

Adding Applications

In the Application panel, identify an application that is to be highly available. The Applications section also includes configuration panels for Application Servers, and in the Extended Config panels, *Application Monitors*.

For information about adding applications, see Chapter 2: Initial Cluster Planning.

Application Servers

In the Application Servers panel, specify information about the application, its start and stop scripts, and application monitors. You must configure an application before you can configure an application server.

For information about application servers, see Chapter 2: Initial Cluster Planning.

Application Monitors

(*Display Extended Configuration Panels*) In the Application Monitors panel, specify the name for the application monitor, which may be the same as the application server name, and define the actions to take should the application fail. Configure at least one application server before setting up an application monitor. You can assign more than one application monitor to an application server.

Note: After you create application monitors, associate them with an application server from the Application Server panel.

For information about monitoring applications, see Chapter 2: Initial Cluster Planning.

Specifying TTY Node/Port Pairs

(*Display Extended Configuration Panels*) In the Configure a Node/Port Pair panel, specify information about TTY connections that notification methods can use to issue a customized page in response to a cluster event.

For information about planning for TTY connections, see Chapter 3: Planning Cluster Network Connectivity.

Custom Remote Notification

(*Display Extended Configuration Panels*) In the Configure Remote Notification panel, specify information about notification methods to issue a customized message in response to a cluster event. Specify a TTY Node in the TTY Node/Port Pairs panel before you set up a custom pager notification. For information about custom pager notification, see Chapter 7: Planning for Cluster Events.

Specifying Cluster Events

(*Display Extended Configuration Panels*) In this panel, specify pre- and post-event scripts for HACMP events as necessary. For information about planning for cluster events, see Chapter 7: Planning for Cluster Events.

Defining File Collections

(*Display Extended Configuration Panels*) In the File Collections panel, specify which files that are located on each cluster node that need be synchronized for HACMP to behave correctly. You can add or remove files from a default list of files. For information about planning for file collections, see the chapter on Configuring AIX for HACMP in the *Installation Guide*.

File Collections Global Settings

(*Display Extended Configuration Panels*) In this panel, specify how often to check whether or *not* files to be automatically synchronized are the same. For information about planning for file collections, see the chapter on Configuring AIX for HACMP in the *Installation Guide*.

Specifying Verification Options

In the Automatic Cluster Verification panel, define settings for cluster verification monitoring. You select the one cluster node and the time at which to run verification for that node.

Understanding the Cluster Definition File

The cluster definition file stores planning information about your cluster, enabling you to first define your cluster and then import the cluster definition file into your cluster definition to come a working cluster configuration.

A cluster definition file can contain information about the following cluster components, but this information is saved only for informational purposes—*not* for HACMP configuration. Other AIX system components manage the configuration data for these components:

- Applications
- Disks
- Volume groups
- Logical volumes
- NFS mounted file systems.

The Online Planning Worksheets application stores definition information about a cluster in an XML file. The structured format of an XML file, allows you to easily edit it by hand using a text or XML editor. This section explains concepts relevant to the HACMP cluster definition XML files; it does *not* explain XML concepts.

Cluster Definition File Format

The cluster definition file is a structured XML template of elements that contain data about your cluster configuration. All elements in the XML file must occur in the same sequence specified within the associated DTD and XSD schema files.

Note: In XML files, the use of shorthand beginning and ending tags for elements with no data, such as <VALIDATED/> is *not* allowed, both the beginning and ending tags must be used. Ensure that you write constructions such as <tag/>, as <tag></tag>; this affects only a few entities, such as <VALIDATED/>.

The following elements are required; all other elements are optional.

- CONFIG
- VERSION
- NODES
- NETWORKS
- INTERFACES.

The following example shows the contents of a cluster definition file that uses only the required elements.

```
<?xml version="1.0" encoding="ISO-8859-1" standalone="yes"?>
<CONFIG>
      <VERSION>5300</VERSION>
      <CLUSTER>
              <CLUSTERNAME>cluster</CLUSTERNAME>
              <AUTHOR></AUTHOR>
              <COMPANY></COMPANY>
              <DATE>Sun Sep 12 18:08:52 EDT 2004</DATE>
      </CLUSTER>
      <NODES>
              <NODE>
                      <NODENAME>node1</NODENAME>
              </NODE>
      </NODES>
      <NETWORKS>
              <NETWORK>
                      <NETWORKNAME>network1</NETWORKNAME>
                      <NETWORKTYPE>ether</NETWORKTYPE>
                      <IPALIASING>yes</IPALIASING>
                      <NETMASK>255.255.255.0</NETMASK>
                      </NETWORK>
      </NETWORKS>
      <INTERFACES>
              <INTERFACE>
                      <INTERFACETYPE>ether</INTERFACETYPE>
                      <NETWORKNAME>network1</NETWORKNAME>
                      <NODENAME>node1</NODENAME>
                      <IPLABELNAME>ip_label1</IPLABELNAME>
              </INTERFACE>
      </INTERFACES>
</CONFIG>
```

Sample Cluster Definition File

To view or modify a sample cluster definition file of a two-node cluster, see:

```
/usr/es/sbin/cluster/worksheets/cluster-sample.haw
```

You can edit this file using an XML editor. When you save the cluster definition file, you can use any filename, but use **.haw** (or **.xml**) as the extension.

Cluster Definition Schema Files

The cluster definition file is accompanied by two schema files (XSD and DTD). These files define the format to follow when creating an XML configuration file by hand. The XSD and DTD files enable you to validate your configuration file. The DTD file validates only the document structure. The XSD schema provides further validation by checking *not* only the document structure but also the type, sequence, and the composition of data. Use whichever one your prefer.

By default, the XSD and DTD files are world and group readable; only the root user has read and write permissions (`644`). These files reside in the following locations:

- **/usr/es/sbin/cluster/worksheets/hacmp-v5300.xsd**
- **/usr/es/sbin/cluster/worksheets/hacmp-v5300.dtd**.

Converting an HACMP Cluster Configuration into OLPW

In addition to creating cluster information, you can use the following means to convert your existing HACMP cluster information into a format Online Planning Worksheets can read:

- Recovering Planning Worksheets from a Working Cluster: Exporting a Definition File
- Converting a Snapshot to a Cluster Definition File.

Note: Although you can import a cluster definition and save it, some of the data is informational only. For information about informational components in a cluster definition file, see the section Entering Data in the Configuration Panels.

Recovering Planning Worksheets from a Working Cluster: Exporting a Definition File

Using SMIT, you can create a cluster definition file from an active HACMP cluster. Then you can open this file in the Online Planning Worksheets application.

This option is vitally useful in cases when you are faced with an already configured cluster which you did not set up. Many times, it is necessary to support such clusters, and the planning worksheets based on which the cluster was originally configured do not exist. This function lets you recover the planning worksheets and obtain a readable, online or printed description of the cluster.

To create a cluster definition file (or to recover planning worksheets) from an active HACMP cluster:

1. Enter `smit hacmp`

2. In SMIT, select **Extended Configuration > Export Definition File for Online Planning Worksheets** and press Enter.
3. Enter field values as follows and press Enter:

   File Name The complete pathname of the cluster definition file.

   The default pathname is /**var/hacmp/log/config.haw**.

   Cluster Notes Any additional comments that pertain to your cluster. The information that you enter here will display in the Cluster Notes panel within Online Planning Worksheets.

4. The *Import Validation* dialog box appears, which indicates whether the HACMP definition file validated successfully during the export.
5. Open the cluster definition file in Online Planning Worksheets.

   For information about opening a cluster definition file, see the section Opening an Existing Cluster Definition File.

Converting a Snapshot to a Cluster Definition File

Converting a cluster snapshot to a cluster definition file enables the Online Planning Worksheets application or the **cl_opsconfig** utility to read in your snapshot information.

When a snapshot is converted to a cluster definition file, all cluster information specified by the schema is converted—even information that Online Planning Worksheets does *not* support.

To create a cluster snapshot:

1. Enter `smit hacmp`
2. In SMIT, select **Extended Configuration > Snapshot Configuration > Convert Existing Snapshot for Online Planning Worksheets** and press Enter.
3. Enter field values as follows and press Enter:

   Cluster Snapshot Name The name of the Cluster Snapshot to convert.

   File Name The path where the cluster definition file will be written. If you provide a relative path name, the file is relative to the **/var/hacmp/log** directory. Maximum length is 128 characters.

   Description Any additional comments that pertain to your cluster. The information that you enter here will display in the Cluster Notes panel within Online Planning Worksheets.

4. Open the cluster definition file in Online Planning Worksheets.

   For information about opening a cluster definition file, see the section Opening an Existing Cluster Definition File.

Applying Worksheet Data to Your HACMP Cluster

After you complete the configuration panels in the application, you save the file, and then apply it to a cluster node. If you use the Online Planning Worksheets application on a Windows system, you first copy the configuration file to a cluster node before applying it.

Prerequisites

Before applying your cluster definition file to a cluster, ensure the following conditions are met:

- The HACMP software is installed on all cluster nodes.
- All hardware devices that you specified for your cluster configuration are in place.
- If you are replacing an existing configuration, any current cluster information in the HACMP configuration database was retained in a snapshot.

  For information about creating, restoring, and applying cluster snapshots in a cluster environment, see Chapter 18: Saving and Restoring Cluster Configurations in the *Administration Guide*.
- Cluster services are stopped on all nodes.
- A valid **/usr/es/sbin/cluster/etc/rhosts** file resides on all cluster nodes. This is required for running the **cl_opsconfig** utility.

Applying Your Cluster Configuration File

To apply your cluster definition file:

1. From the Online Planning Worksheets application, validate your cluster definition file as described in section Validating a Cluster Definition.
2. Create a report to document your cluster configuration as described in section Creating an HTML Configuration Report.
3. Save the file and exit the application. If your cluster configuration file resides on a Windows system, copy the file to an HACMP node.
4. From the cluster node, run the **cl_opsconfig** command as follows:

   ```
   /usr/es/sbin/cluster/utilities/cl_opsconfig your_config_file
   ```

   where *your_config_file* is the name of the configuration file on the node.

   The **cl_opsconfig** utility validates the file (if the Online Planning Worksheets application is installed locally), applies the information to your cluster, and then verifies it. During verification, onscreen messages appear, indicating the events taking place and any warnings or errors.

You can view the **cl_opsconfig** error messages on screen, or redirect them to a log file. You can redirect the standard error output as in the follows for the korn shell (other shells may vary):

```
/usr/sbin/cluster/utilities/cl_opsconfig your_config_file 2> output_file
```

Do *not* redirect *all* output (standard output and standard error).

Troubleshooting Cluster Definition File Problems

Validation first determines whether the <VALIDATED> element exists. If it does *not* exist and Online Planning Worksheets has been installed, OLPW validates the XML configuration file and processing continues. If OLPW has *not* been installed, an error message notifies you that Online Planning Worksheets is *not* installed and file verification terminates.

If any error messages appear, fix the problem(s) using the Online Planning Worksheets application, and then repeat the steps described in Applying Worksheet Data to Your HACMP Cluster. If a prior cluster definition file exists, **cl_opsconfig** prompts you for confirmation before deleting it.

Alternatively, you can address errors in HACMP SMIT. In this case, the final report generated from the Online Planning Worksheets application does *not* provide an accurate picture of your initial configuration.

Where You Go from Here

After you apply the configuration to your cluster, you have at a minimum a basic working HACMP configuration.

Next proceed with additional configuration and customization tasks—for example, configuring AIX Error Notification, runtime parameters, custom scripts, and so forth. For information about configuring and customizing HACMP, see the appropriate chapters in this guide or in the *Administration Guide.*

Appendix A: Planning Worksheets

Print and use the paper planning worksheets from the PDF version of this guide. In the PDF version, each new worksheet is aligned properly to start at the top of a page. You may need more than one copy of some worksheets.

This appendix contains the following worksheets:

Worksheet	Purpose
Two-Node Cluster Configuration Worksheet	Use this worksheet to record the information required to complete the entries in the Two-Node Cluster Configuration Assistant.
TCP/IP Networks Worksheet	Use this worksheet to record the TCP/IP network topology for a cluster. Complete one worksheet per cluster.
TCP/IP Network Interface Worksheet	Use this worksheet to record the TCP/IP network interface cards connected to each node. You need a separate worksheet for each node defined in the cluster, print a worksheet for each node and fill in a node name on each worksheet.
Point-to-Point Networks Worksheet	Use this worksheet to record the point-to-point network topology for a cluster. Complete one worksheet per cluster.
Serial Network Interface Worksheet	Use this worksheet to record the serial network interface cards connected to each node. You need a separate worksheet for each node defined in the cluster, print a worksheet for each node and fill in the node name on each worksheet.
Fibre Channel Disks Worksheet	Use this worksheet to record information about Fibre Channel disks to be included in the cluster. Complete a separate worksheet for each cluster node.
Shared SCSI Disk Worksheet	Use this worksheet to record the shared SCSI disk configuration for the cluster. Complete a separate worksheet for each shared bus.
Shared IBM SCSI Disk Arrays Worksheet	Use this worksheet to record the shared IBM SCSI disk array configurations for the cluster. Complete a separate worksheet for each shared SCSI bus.
Shared IBM SCSI Tape Drive Worksheet	Use this worksheet to record the shared IBM SCSI tape drive configurations for the cluster. Complete a separate worksheet for each shared tape drive.
Shared IBM Fibre Tape Drive Worksheet	Use this worksheet to record the shared IBM Fibre tape drive configurations for the cluster. Complete a separate worksheet for each shared tape drive.
Shared IBM Serial Storage Architecture Disk Subsystems Worksheet	Use this worksheet to record the IBM 7131-405 or 7133 SSA shared disk configuration for the cluster.

Worksheet	Purpose
Non-Shared Volume Group Worksheet (Non-Concurrent Access)	Use this worksheet to record the volume groups and file systems that reside on a node's internal disks in a non-concurrent access configuration. You need a separate worksheet for each volume group, print a worksheet for each volume group and fill in a node name on each worksheet.
Shared Volume Group and File System Worksheet (Non-Concurrent Access)	Use this worksheet to record the shared volume groups and file systems in a non-concurrent access configuration. You need a separate worksheet for each shared volume group, print a worksheet for each volume group and fill in the names of the nodes sharing the volume group on each worksheet.
NFS-Exported File System or Directory Worksheet (Non-Concurrent Access)	Use this worksheet to record the file systems and directories NFS-exported by a node in a non-concurrent access configuration. You need a separate worksheet for each node defined in the cluster, print a worksheet for each node and fill in a node name on each worksheet.
Non-Shared Volume Group Worksheet (Concurrent Access)	Use this worksheet to record the volume groups and file systems that reside on a node's internal disks in a concurrent access configuration. You need a separate worksheet for each volume group, print a worksheet for each volume group and fill in a node name on each worksheet.
Shared Volume Group and File System Worksheet (Concurrent Access)	Use this worksheet to record the shared volume groups and file systems in a concurrent access configuration. You need a separate worksheet for each shared volume group, print a worksheet for each volume group and fill in the names of the nodes sharing the volume group on each worksheet.
Application Worksheet	Use these worksheets to record information about applications in the cluster.
Fast Connect Worksheet	Use this worksheet to record Fast Connect resources
Communication Links (SNA-Over-LAN) Worksheet	Use this worksheet to record information about SNA-over-LAN communications links in the cluster.
Communication Links (X.25) Worksheet	Use this worksheet to record information about X.25 communications links in the cluster.
Communication Links (SNA-Over-X.25) Worksheet	Use this worksheet to record information about SNA-over-X.25 communications links in the cluster.
Application Server Worksheet	Use these worksheets to record information about application servers in the cluster.
Application Monitor Worksheet (Process Monitor)	Use this worksheet to record information for configuring a process monitor for an application.
Application Monitor Worksheet (Custom Monitor)	Use this worksheet to record information for configuring a custom (user-defined) monitor method for an application.

Worksheet	Purpose
Resource Group Worksheet	Use this worksheet to record the resource groups for a cluster.
Cluster Event Worksheet	Use this worksheet to record the planned customization for an HACMP cluster event.
Sample Cluster Site Worksheet	Use this worksheet to record planned cluster sites.
HACMP File Collection Worksheet	Use this worksheet to record planned HACMP cluster file collections.

Two-Node Cluster Configuration Worksheet

Local Node	
Takeover (Remote) Node	
Communication Path to Takeover Node	
Application Server	
Application Start Script	
Application Stop Script	
Service IP Label	

Sample Two-Node Cluster Configuration Worksheet

Local Node	nodea
Takeover (Remote) Node	nodeb
Communication Path to Takeover Node	10.11.12.13
Application Server	appsrv1
Application Start Script	/usr/es/sbin/cluster/utils/start_app1
Application Stop Script	/usr/es/sbin/cluster/utils/stop_app1
Service IP Label	app1_svc

TCP/IP Networks Worksheet

Cluster Name					
Network Name	**Network Type**	**Netmask**	**Node Names**	**IPAT via IP Aliases**	**IP Address Offset for Heartbeating over IP Aliases**

Sample TCP/IP Networks Worksheet

Cluster Name					
Network Name	**Network Type**	**Netmask**	**Node Names**	**IPAT via IP Aliases**	**IP Address Offset for Heartbeating over IP Aliases**
ether1	Ethernet	255.255.255.0	clam, mussel, oyster	enable	
token1	Token-Ring	255.255.255.0	clam, mussel, oyster	enable	
fddi1	FDDI	255.255.255.0	clam, mussel	disable	
atm1	ATM	255.255.255.0	clam, mussel	unsupported	

TCP/IP Network Interface Worksheet

Node Name							
IP Label Address	**IP Alias Distribution Preference**	**Network Interface**	**Network Name**	**Interface Function**	**IP Address**	**Netmask**	**Hardware Address**

Sample TCP/IP Network Interface Worksheet

Node Name		nodea					
IP Label Address	**IP Alias Distribution Preference**	**Network Interface**	**Network Name**	**Interface Function**	**IP Address**	**Netmask**	**Hardware Address**
nodea-en0	Anti-collocation (default)	len0	ether1	service	100.10.1.10	255.255.255.0	0x08005a7a7610
nodea-nsvc1	Anti-collocation (default)	en0	ether1	non-service	100.10.1.74	255.255.255.0	
nodea-en1	Anti-collocation (default)	en1	ether1	non-service	100.10.11.11	255.255.255.0	
nodea-tr0	collocation	tr0	token1	service	100.10.2.20	255.255.255.0	0x42005aa8b57b
nodea-nsvc2	Anti-collocation (default)	tr0	token1	non-service	100.10.2.84	255.255.255.0	
nodea-fi0	collocation	fi0	fddi1	service	100.10.3.30	255.255.255.0	
nodea-svc	collocation	css0	hps1	service			
nodea-nsvc3	Anti-collocation (default)	css0	hps1	non-service			
nodea-at0	collocation	at0	atm1	service	100.10.7.10	255.255.255.0	0x0020481a396500
nodea-nsvc1	Anti-collocation (default)	at0	atm1	non-service	100.10.7.74	255.255.255.0	

Point-to-Point Networks Worksheet

Cluster Name			
Network Name	**Network Type**	**Node Names**	**Hdisk** **(for diskhb networks)**

RS232, target mode SCSI, and target mode SSA, and disk heartbeating links do *not* use the TCP/IP protocol and do *not* require a netmask or an IP address.

Miscellaneous Data

Record any extra information about devices used to extend point-to-point links (for example, modem number or extender information).

____

____

____

____

Sample Point-to-Point Networks Worksheet

Cluster Name	clus1		
Network Name	**Network Type**	**Node Names**	**Hdisk** **(for diskhb networks)**
diskhb1	diskhb	nodeb, nodec	hdisk2
tmscsi1	Target Mode SCSI	nodea, nodeb	--
tmssa1	Target Mode SSA	nodea, nodeb	--

RS232, target mode SCSI, and target mode SSA, and disk heartbeating links do *not* use the TCP/IP protocol and do *not* require a netmask or an IP address.

Miscellaneous Data

Record any extra information about devices used to extend serial links (for example modem number or extender information).

Serial Network Interface Worksheet

Node Name					
Slot Number	**Interface Name**	**Adapter Label**	**Network Name**	**Network Attribute**	**Adapter Function**
				serial	service
				serial	service
				serial	service
				serial	service
				serial	service
				serial	service
				serial	service
				serial	service
				serial	service

Non-IP networks do *not* carry TCP/IP traffic. As a result, no non-service addresses, identifiers (IP addresses), or interface hardware addresses are required to maintain keepalives and control messages between nodes.

Sample Serial Network Interface Worksheet

Node Name	**nodea**				
Slot Number	**Interface Name**	**Adapter Label**	**Network Name**	**Network Attribute**	**Adapter Function**
SS2	/dev/tty1	nodea_tty1	rs232a	serial	service
08	scsi2	nodea_tmscsi2	tmscsi1	serial	service
01	tmssa1	nodea_tmssa1	tmssa1	serial	service

Fibre Channel Disks Worksheet

Node Name	
–	
Fibre Channel Adapter	**Disks Associated with Adapter**

Sample Fibre Channel Disks Worksheet

Node Name	nodea
Fibre Channel Adapter	**Disks Associated with Adapter**
fcs0	hdisk2
	hdisk3
	hdisk4
	hdisk5
	hdisk6

Shared SCSI Disk Worksheet

Complete a separate worksheet for each shared SCSI bus.

Cluster Name				
Host and Adapter Information				
	Node A	**Node B**	**Node C**	**Node D**
Node Name				
Slot Number				
Logical Name				
SCSI Device IDs on Shared Bus				
	Node A	**Node B**	**Node C**	**Node D**
Adapter				
First Shared Drive				
Second Shared Drive				
Third Shared Drive				
Fourth Shared Drive				
Fifth Shared Drive				
Sixth Shared Drive				

Shared Drives

Disk	**Size**	**Logical Device Name**			
		Node A	**Node B**	**Node C**	**Node D**
First					
Second					
Third					
Fourth					
Fifth					
Sixth					

Sample Shared SCSI Disk Worksheet

Complete a separate worksheet for each shared SCSI bus.

Cluster Name				
Host and Adapter Information				
	Node A	**Node B**	**Node C**	**Node D**
Node Name	nodea	nodeb		
Slot Number	7	7		
Logical Name	scsi1	scsi1		
SCSI Device IDs on Shared Bus				
	Node A	**Node B**	**Node C**	**Node D**
Adapter	6	5		
First Shared Drive	3			
Second Shared Drive	4			
Third Shared Drive	5			
Fourth Shared Drive				
Fifth Shared Drive				
Sixth Shared Drive				

Shared Drives

Disk	**Size**	**Logical Device Name**			
		Node A	**Node B**	**Node C**	**Node D**
First	670	hdisk2	hdisk2		
Second	670	hdisk3	hdisk3		
Third	670	hdisk4	hdisk4		
Fourth					
Fifth					
Sixth					

Shared IBM SCSI Disk Arrays Worksheet

Complete a separate worksheet for each shared SCSI disk array.

Host and Adapter Information

	Node A	Node B	Node C	Node D
Node Name				
Slot Number				
Logical Name				

SCSI Device IDs on Shared Bus

Adapter	Node A	Node B	Node C	Node D

Shared Drives	
Size	**RAID Level**

Sample Shared IBM SCSI Disk Arrays Worksheet

Complete a separate worksheet for each shared SCSI disk array.

Host and Adapter Information				
	Node A	**Node B**	**Node C**	**Node D**
Node Name	nodea	nodeb		
Slot Number	2	2		
Logical Name	scsi1	scsi1		
SCSI Device IDs on Shared Bus				
Adapter	**Node A**	**Node B**	**Node C**	**Node D**
	14	15		
Shared Drives				
Size	**RAID Level**			
2GB	5			
2GB	3			
2GB	5			
2GB	5			

Shared IBM SCSI Tape Drive Worksheet

Note: Complete a separate worksheet for each shared SCSI tape drive.

Host and Adapter Information		
	Node A	**Node B**
Node Name		
Slot Number		
Logical Name		
SCSI Tape Drive IDs on Shared Bus		
	Node A	**Node B**
Adapter		
Tape Drive		
Shared Drives	**Logical Device Name**	
	Node A	**Node B**

Sample Shared IBM SCSI Tape Drive Worksheet

This sample worksheet shows a shared SCSI tape drive configuration.

Host and Adapter Information		
	Node A	**Node B**
Node Name	nodea	nodeb
Slot Number	2	2
Logical Name	scsi1	scsi1
SCSI Tape Drive IDs on Shared Bus		
	Node A	**Node B**
Adapter	5	6
Tape Drive	2	
Shared Drives	**Logical Device Name**	
	Node A	**Node B**
5.0GB 8mm tape drive 10GB	/dev/rmt0	/dev/rmt0

Shared IBM Fibre Tape Drive Worksheet

Host and Adapter Information		
	Node A	**Node B**
Node Name		
Slot Number		
Logical Name		
Fibre Device IDs on Shared Bus		
	Node A	**Node B**
SCSI ID		
LUN ID		
World Wide Name		
Shared Drives	**Logical Device Name**	
Tape Drive Name	**Node A**	**Node B**

Sample Shared IBM Fibre Tape Drive Worksheet

This sample worksheet shows a shared Fibre tape drive configuration (recorded after the tape drive had been configured).

Host and Adapter Information		
	Node A	**Node B**
Node Name	nodea	nodeb
Slot Number	1P-18	04-02
Logical Name	fcs0	fcs0
Fibre Device IDs on Shared Bus		
	Node A	**Node B**
SCSI ID	scsi_id 0x26	scsi_id 0x26
LUN ID	lun_id 0x0	lun_id 0x0
World Wide Name	ww_name 0x5005076300404576	ww_name 0x5005076300404576
Shared Drives	**Logical Device Name**	
Tape Drive Name	**Node A**	**Node B**
IBM 3590	/dev/rmt0	/dev/rmt0

Shared IBM Serial Storage Architecture Disk Subsystems Worksheet

Host and Adapter Information				
	Node A	**Node B**	**Node C**	**Node D**
Node Name				
SSA Adapter Label				
Slot Number				
Dual-Port Number				
SSA Logical Disk Drive				
Logical Device Name				
Node A	**Node B**	**Node C**	**Node D**	**Node A**
SSA Logical Disk Drive				
Logical Device Name				
Node A	**Node B**	**Node C**	**Node D**	**Node A**

Sample Shared IBM Serial Storage Architecture Disk Subsystems Worksheet

Host and Adapter Information				
	Node A	**Node B**	**Node C**	**Node D**
Node Name	clam	mussel		
SSA Adapter Label	ha1, ha2	ha1, ha2		
Slot Number	2, 4	2, 4		
Dual-Port Number	a1, a2	a1, a2		
SSA Logical Disk Drive				
Logical Device Name				
Node A	**Node B**	**Node C**	**Node D**	**Node A**
hdisk2	hdisk2			
hdisk3	hdisk3			
hdisk4	hdisk4			
hdisk5	hdisk5			
SSA Logical Disk Drive				
Logical Device Name				
Node A	**Node B**	**Node C**	**Node D**	**Node A**
hdisk2	hdisk2			
hdisk3	hdisk3			
hdisk4	hdisk4			
hdisk5	hdisk5			

Non-Shared Volume Group Worksheet (Non-Concurrent Access)

Node Name	
Volume Group Name	
Physical Volumes	
Logical Volume Name	
Number of Copies of Logical Partition	
On Separate Physical Volumes?	
File System Mount Point	
Size (in 512-byte blocks)	
Logical Volume Name	
Number of Copies of Logical Partition	
On Separate Physical Volumes?	
File System Mount Point	
Size (in 512-byte blocks)	

Sample Non-Shared Volume Group Worksheet (Non-Concurrent Access)

Node Name	clam
Volume Group Name	localvg
Physical Volumes	hdisk1
Logical Volume Name	locallv
Number of Copies of Logical Partition	1
On Separate Physical Volumes?	no
File System Mount Point	localfs
Size (in 512-byte blocks)	100000
Logical Volume Name	
Number of Copies of Logical Partition	
On Separate Physical Volumes?	
File System Mount Point	
Size (in 512-byte blocks)	

Shared Volume Group and File System Worksheet (Non-Concurrent Access)

	Node A	**Node B**	**Node C**	**Node D**
Node Names				
Shared Volume Group Name				
Major Number				
Log Logical Volume Name				
Physical Volumes				
Cross-site LVM Mirror				
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
File System Mount Point				
Size (in 512-byte blocks)				
Cross-site LVM Mirroring enabled				
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
File System Mount Point				
Size (in 512-byte blocks)				
Cross-site LVM Mirroring enabled				

Sample Shared Volume Group and File System Worksheet (Non-Concurrent Access)

	Node A	**Node B**	**Node C**	**Node D**
Node Names	trout	guppy		
Shared Volume Group Name	bassvg			
Major Number	24	24		
Log Logical Volume Name	bassloglv			
Physical Volumes	hdisk6	hdisk6		
	hdisk7	hdisk7		
	hdisk13	hdisk16		
Cross-site LVM Mirror	site1	site2		
Logical Volume Name		basslv		
Number of Copies of Logical Partition		3		
On Separate Physical Volumes?		yes		
File System Mount Point		/bassfs		
Size (in 512-byte blocks)		200000		
Cross-site LVM Mirroring enabled		yes		
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
File System Mount Point				
Size (in 512-byte blocks)				
Cross-site LVM Mirroring enabled				

NFS-Exported File System or Directory Worksheet (Non-Concurrent Access)

Resource Group		
Network for NFS Mount		
File System Mounted before IP Configured?		
Note: Export options include read-only, root access, and so on. For a full list of export options, see the **exports** man page		
File System or Directory to Export (NFSv2/3)		
Export Options		
File System or Directory to Export (NFSv4)		
Export Options		
File System or Directory to Export (NFSv2/3)		
Export Options		
File System or Directory to Export (NFSv4)		
Export Options		
Stable Storage Path (NFSv4)		

Sample NFS-Exported File System or Directory Worksheet (Non-Concurrent Access

Resource Group	rg1	
Network for NFS Mount	tr1	
File System Mounted before IP Configured?	true	
Note: Export options include read-only, root access, and so on. For a full list of export options, see the **exports** man page		
File System or Directory to Export (NFSv2/3)	/fs1 /fs5	
Export Options		
client access:client1		
root access: node 1, node 2		
mode: read/write		
File System or Directory to Export (NFSv4)	/fs5 /fs6	
Export Options		
client access: client 2		
root access: node 1, node 2		
mode: read only		
Stable Storage Path (NFSv4): /mnt_ss/		

Non-Shared Volume Group Worksheet (Concurrent Access)

Node Name	
Volume Group Name	
Physical Volumes	
Logical Volume Name	
Number of Copies of Logical Partition	
On Separate Physical Volumes?	
File System Mount Point	
Size (in 512-byte blocks)	
Logical Volume Name	
Number of Copies of Logical Partition	
On Separate Physical Volumes?	
File System Mount Point	
Size (in 512-byte blocks)	

Sample Non-Shared Volume Group Worksheet (Concurrent Access)

Node Name	clam
Volume Group Name	localvg
Physical Volumes	hdisk1
Logical Volume Name	locallv
Number of Copies of Logical Partition	1
On Separate Physical Volumes?	no
File System Mount Point	/localfs
Size (in 512-byte blocks)	100000
Logical Volume Name	
Number of Copies of Logical Partition	
On Separate Physical Volumes?	
File System Mount Point	
Size (in 512-byte blocks)	

Shared Volume Group and File System Worksheet (Concurrent Access)

	Node A	Node B	Node C	Node D
Node Names				
Shared Volume Group Name				
Physical Volumes				
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
Size (in 512-byte blocks)				
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
Size (in 512-byte blocks)				
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
Size (in 512-byte blocks)				

Sample Shared Volume Group and File System Worksheet (Concurrent Access)

	Node A	**Node B**	**Node C**	**Node D**
Node Names	trout	guppy		
Shared Volume Group Name	bassvg			
Physical Volumes	hdisk6	hdisk6		
	hdisk7	hdisk7		
	hdisk13	hdisk16		
Logical Volume Name		basslv		
Number of Copies of Logical Partition		3		
On Separate Physical Volumes?		yes		
Size (in 512-byte blocks)		/bassfs		
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
Size (in 512-byte blocks)				
Logical Volume Name				
Number of Copies of Logical Partition				
On Separate Physical Volumes?				
Size (in 512-byte blocks)				

Application Worksheet

<table>
<tr><td><b>Application Name</b></td><td colspan="8">______________________</td></tr>
<tr><td></td><td colspan="8"><b>Key Application Files</b></td></tr>
<tr><td></td><td colspan="2"><b>Directory or Path</b></td><td colspan="2"><b>File System</b></td><td colspan="2"><b>Location</b></td><td colspan="2"><b>Sharing</b></td></tr>
<tr><td><b>Executable Files</b></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td></tr>
<tr><td><b>Configuration Files</b></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td></tr>
<tr><td><b>Data Files or Devices</b></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td></tr>
<tr><td><b>Log Files or Devices</b></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td><td colspan="2"></td></tr>
<tr><td><b>Cluster Name</b></td><td colspan="8"></td></tr>
<tr><td colspan="9"><b>Fallover Strategy (P = primary; T = takeover)</b></td></tr>
<tr><td><b>Node</b></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td><b>Strategy</b></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td colspan="9"><b>Normal Start Commands and Procedures</b></td></tr>
<tr><td></td><td colspan="8"></td></tr>
<tr><td></td><td colspan="8"></td></tr>
<tr><td></td><td colspan="8"></td></tr>
<tr><td colspan="9"><b>Verification Commands and Procedures</b></td></tr>
<tr><td></td><td colspan="8"></td></tr>
<tr><td></td><td colspan="8"></td></tr>
<tr><td></td><td colspan="8"></td></tr>
<tr><td colspan="9"><b>Node Reintegration and Takeover Caveats</b></td></tr>
<tr><td><b>Node</b></td><td colspan="8"></td></tr>
<tr><td>One</td><td colspan="8"></td></tr>
<tr><td>Two</td><td colspan="8"></td></tr>
<tr><td>Three</td><td colspan="8"></td></tr>
</table>

Application Worksheet (continued)

Normal Stop Commands and Procedures	
Verification Commands and Procedures	
Node Reintegration and Takeover Caveats	
Node	
One	
Two	
Three	

Sample Application Worksheet

Application Name				
	Key Application Files			
	Directory or Path	**File System**	**Location**	**Sharing**
Executable Files	/app1/bin	/app1	internal	non-shared
Configuration Files	/app1/config/one	/app1/config/one	external	shared
Data Files or Devices	/app1lv1	NA	external	shared
Log Files or Devices	/app1loglv1	NA	external	shared
Cluster Name	tetra			
Fallover Strategy (P = primary; T = takeover)				
Node	One	Two	Three	Four
Strategy	P	NA	T1	T2
Normal Start Commands and Procedures				
	• Ensure that the app1 server group is running • If the app1 server group is *not* running, as user app1_adm, execute app1 start -Ione • Ensure that the app1 server is running • If Node Two is up, start (restart) app1_client on Node Two			
Verification Commands and Procedures				
	• Run the following command: lssrc -g app1 • Ensure from the output that daemon1, daemon2, and daemon3 are “Active” • Send notification if *not* “Active”			
Node Reintegration and Takeover Caveats				
Node	NA			
One	NA			
Two	Must restart the current instance of app1 with app1start -Ione -Ithree			
Three	Must restart the current instance of app1 with app1start -Ione -Ifour			

Sample Application Worksheet (continued)

Normal Stop Commands and Procedures	
	• Ensure that the app1 server group is running • If the app1 server group is running, stop by app1stop as user app1_adm • Ensure that the app1 server is stopped • If the app1 server is still up, stop individual daemons with the kill command
Verification Commands and Procedures	
	• Run the following command: lssrc -g app1 • Ensure from the output that daemon1, daemon2, and daemon3 are Inoperative
Node Reintegration and Takeover Caveats	
Node	NA
One	May want to notify app1_client users to log off
Two	Must restart the current instance of app1 with app1start -Ithree
Three	Must restart the current instance of app1 with app1start -Ifour

In this sample worksheet, the server portion of the application, app1, normally runs on three of the four cluster nodes: nodes One, Three, and Four. Each of the three nodes is running its own app1 instance: one, three, or four. When a node takes over an app1 instance, the takeover node must restart the application server using flags for multiple instances. Also, because Node Two within this configuration runs the client portion associated with this instance of app1, the takeover node must restart the client when the client’s server instance is restarted.

Fast Connect Worksheet

Cluster Name		
_		
Resource Group	**Nodes**	**Fast Connect Resources**
Resource Group	**Nodes**	**Fast Connect Resources**

Sample Fast Connect Worksheet

Cluster Name	cluster2	
–		
Resource Group	**Nodes**	**Fast Connect Resources**
rg1	NodeA, NodeC	FS1%f%/smbtest/fs1
		LPT1%p%printq
Resource Group	**Nodes**	**Fast Connect Resources**
rg2	Node B, Node D	FS2%f%/smbtest/fs2
		LPT2%p%printq

Communication Links (SNA-Over-LAN) Worksheet

Cluster Name	
Resource Group	
Nodes	
Communication Link Name	
DLC Name	
Port	
Link Station	
Application Service File	
–	
Resource Group	
Nodes	
Communication Link Name	
DLC Name	
Port	
Link Station	
Application Service File	

Sample Communication Links (SNA-Over-LAN) Worksheet

Cluster Name	cluster1
–	
Resource Group	rg1
Nodes	nodeA, nodeB
Communication Link Name	snalink1
DLC Name	snaprofile1
Port	snaport1
Link Station	snastation1
Application Service File	/tmp/service1.sh
Resource Group	
Nodes	
Communication Link Name	
DLC Name	
Port	
Link Station	
Application Service File	

Communication Links (X.25) Worksheet

Cluster Name	
Resource Group	
Nodes	
Communication Link Name	
Port	
Address or NUA	
Network ID	
Country Code	
Adapter Names(s)	
Application Service File	

Sample Communication Links (X.25) Worksheet

Cluster Name	mycluster
Resource Group	cascrg1
Nodes	nodeA, nodeB
Communication Link Name	x25link1
Port	sx25a2
Address or NUA	241
Network ID	5 *(can be left blank)*
Country Code	*(system default automatically used for local country code)*
Adapter Names(s)	adapterAsx25a2, adapterBsx25a2
Application Service File	/tmp/startx25.sh

Communication Links (SNA-Over-X.25) Worksheet

Cluster Name	
Resource Group	
Nodes	
Communication Link Name	
X.25 Port	
X.25 Address or NUA	
X.25 Network ID	
X.25 Country Code	
X.25 Adapter Names(s)	
SNA DLC	
SNA Port(s)	
SNA Link Station(s)	
Application Service File	

Sample Communication Links (SNA-Over-X.25) Worksheet

Cluster Name	mycluster
Resource Group	casc_rg2
Nodes	nodeA, nodeB, nodeC
Communication Link Name	snax25link1
X.25 Port	sx25a0
X.25 Address or NUA	241
X.25 Network ID	5 *(can be left blank)*
X.25 Country Code	*(system default automatically used for local country code)*
X.25 Adapter Names(s)	adapterAsx25a0, adapterBsx25a0, adapterCsx25a0
SNA DLC	dlcprofile2
SNA Port(s)	snaport2
SNA Link Station(s)	snastation2
Application Service File	/tmp/snax25start.sh

Application Server Worksheet

Cluster Name	
Note: Use full pathnames for all user-defined scripts.	
Server Name	
Start Script	
Stop Script	
Server Name	
Start Script	
Stop Script	
Server Name	
Start Script	
Stop Script	
Server Name	
Start Script	
Stop Script	

Sample Application Server Worksheet

Cluster Name	cluster1
Note: Use full pathnames for all user-defined scripts.	
Server Name	mydemo
Start Script	/usr/es/sbin/cluster/utils/start_mydemo
Stop Script	/usr/es/sbin/cluster/utils/stop_mydemo
Server Name	
Start Script	
Stop Script	
Server Name	
Start Script	
Stop Script	
Server Name	
Start Script	
Stop Script	

Application Monitor Worksheet (Process Monitor)

Cluster Name	
Application Server Name	
Can Application Be Monitored with Process Monitor?*	**Yes / No** *(If **No**, go to Custom Worksheet)*
Processes to Monitor	
Process Owner	
Instance Count	
Stabilization Interval	
Restart Count	
Restart Interval	
Action on Application Failure	
Notify Method	
Cleanup Method	
Restart Method	

Sample Application Monitor Worksheet (Process Monitor)

Cluster Name	cluster1
Application Server Name	mydemo
Can Application Be Monitored with Process Monitor?*	yes
Processes to Monitor	demo
Process Owner	root
Instance Count	1
Stabilization Interval	30
Restart Count	3
Restart Interval	95
Action on Application Failure	fallover
Notify Method	/usr/es/sbin/cluster/events/notify_demo
Cleanup Method	/usr/es/sbin/cluster/utils//events/stop_demo
Restart Method	/usr/es/sbin/cluster/utils/events/start_demo

Application Monitor Worksheet (Custom Monitor)

Cluster Name	
Application Server Name	
Monitor Method	
Monitor Interval	
Hung Monitor Signal	
Stabilization Interval	
Restart Count	
Restart Interval	
Action on Application Failure	
Notify Method	
Cleanup Method	
Restart Method	

Sample Application Monitor Worksheet (Custom Monitor)

Cluster Name	cluster1
Application Server Name	mydemo
Monitor Method	/usr/es/sbin/cluster/events/utils/monitor_mydemo
Monitor Interval	60
Hung Monitor Signal	9
Stabilization Interval	30
Restart Count	3
Restart Interval	280
Action on Application Failure	notify
Notify Method	/usr/es/sbin/cluster/events/utils/notify_mydemo
Cleanup Method	/usr/es/sbin/cluster/events/utils/stop_mydemo
Restart Method	/usr/es/sbin/cluster/events/utils/start_mydemo

Resource Group Worksheet

Cluster Name	
Resource Group Name	
Participating Node Names	
Inter-Site Management Policy	
Startup Policy	
Fallover Policy	
Fallback Policy	
Delayed Fallback Timer	
Settling Time	
Runtime Policies	
Dynamic Node Priority Policy	
Processing Order (Parallel, Serial, or Customized)	
Service IP Label	
File Systems	
File Systems Consistency Check	
File Systems Recovery Method	
File Systems or Directories to Export	
File Systems or Directories to NFS Mount (NFSv2/3)	
File Systems or Directories to NFS Mount (NFSv4)	
Stable Storage Path (NFSv4)	
Network for NFS Mount	
Volume Groups	
Concurrent Volume Groups	
Raw Disk PVIDs	
Fast Connect Services	
Tape Resources	

Application Servers	
Highly Available Communication Links	
Primary Workload Manager Class	
Secondary WLM Class (only non-concurrent resource groups that do *not* have the Online Using Node Distribution Policy startup)	
Miscellaneous Data	
Auto Import Volume Groups	
Disk Fencing Activated	
File Systems Mounted before IP Configured	

Sample Resource Group Worksheet

Cluster Name	clus1
Resource Group Name	rotgrp1
Participating Node Names	ignore
Inter-Site Management Policy	clam, mussel, oyster
Startup Policy	Online Using Node Distribution Policy
Fallover Policy	Fallover to Next Priority Available Node
Fallback Policy	Never Fallback
Delayed Fallback Timer	
Settling Time	
Runtime Policies	
Dynamic Node Priority Policy	
Processing Order (Parallel, Serial, or Customized)	
Service IP Label	myname_svc
File Systems	/sharedfs1 /sharedvg2 /myfs
File Systems Consistency Check	fsck
File Systems Recovery Method	sequential
File Systems or Directories to Export	
File Systems or Directories to NFS Mount (NFSv2/3)	/sharedvg1
File Systems or Directories to NFS Mount (NFSv4)	/sharedvg1 /sharedvg2
Stable Storage Path (NFSv4)	/myfs/stable_storage/
Network for NFS Mount	ether1
Volume Groups	sharedvg
Concurrent Volume Groups	
Raw Disk PVIDs	
Fast Connect Services	
Tape Resources	

Application Servers	mydemo
Highly Available Communication Links	
Primary Workload Manager Class	
Secondary WLM Class (only non-concurrent resource groups that do *not* have the Online Using Node Distribution Policy startup)	
Miscellaneous Data	
Auto Import Volume Groups	false
Disk Fencing Activated	false
File Systems Mounted before IP Configured	false

Cluster Event Worksheet

Use full pathnames for all Cluster Event Methods, Notify Commands, and Recovery commands.

Cluster Name	
Cluster Event Description	
Cluster Event Method	
Cluster Event Name	
Event Command	
Notify Command	
Remote Notification Message Text	
Remote Notification Message Location	
Pre-Event Command	
Post-Event Command	
Event Recovery Command	
Recovery Counter	
Time Until Warning	
Cluster Event Name	
Event Command	
Notify Command	
Remote Notification Message Text	
Remote Notification Message Location	
Pre-Event Command	
Post-Event Command	
Event Recovery Command	
Recovery Counter	
Time Until Warning	

Sample Cluster Event Worksheet

Use full pathnames for all user-defined scripts.

Cluster Name	bivalves
Cluster Event Description	
Cluster Event Method	
Cluster Event Name	node_down_complete
Event Command	
Notify Command	
Remote Notification Message Text	
Remote Notification Message Location	
Pre-Event Command	
Post-Event Command	/usr/local/wakeup
Event Recovery Command	
Recovery Counter	
Time Until Warning	
Cluster Event Name	
Event Command	
Notify Command	
Remote Notification Message Text	
Remote Notification Message Location	
Pre-Event Command	
Post-Event Command	
Event Recovery Command	
Recovery Counter	
Time Until Warning	

Cluster Site Worksheet

Site Name	
_	
Cluster Nodes in Site	
For HAGEO Sites	
Site Dominance	
Site Backup Communication Method	

Sample Cluster Site Worksheet

Site Name	Site_1
Cluster Nodes in Site	
nodea	
nodeb	
For HAGEO Sites	
Site Dominance	
Site Backup Communication Method	

HACMP File Collection Worksheet

Cluster Name	
File Collection name	
File Collection description	
Propagate Files before verification	
Propagate file automatically	
Files to include in this collection	
Automatic check time limit	

Sample HACMP File Collection Worksheet

Cluster Name	MyCluster
File Collection name	Apache_Files
File Collection description	Apache configuration files
Propagate Files before verification	Yes
Propagate file automatically	Yes
Files to include in this collection	/usr/local/apache/conf/httpd.conf /usr/local/apache/conf/ss/.conf /usr/local/apache/conf/mime_types
Automatic check time limit	30 minutes

Appendix B: Applications and HACMP

This appendix addresses some of the key issues to consider when making your applications highly available under HACMP.

HACMP allows you to configure clusters with multi-tiered applications by establishing dependencies between resource groups containing different applications. This appendix describes resource group dependencies and how they can help with keeping dependent applications highly available.

For details on the planning for and configuring of applications, see Chapter 2: Initial Cluster Planning and the *Administration Guide*.

For general information about keeping a cluster running on a 7x24 basis, see Appendix A: 7x24 Maintenance in the *Administration Guide*.

Overview

Besides understanding the hardware and software needed to make a cluster highly available, you will need to spend some time on application considerations when planning your HACMP environment. The goal of clustering is to keep your important applications available despite any single point of failure. To achieve this goal, it is important to consider the aspects of an application that make it recoverable under HACMP.

There are few hard and fast requirements that an application must meet to recover well under HACMP. For the most part, there are simply good practices that can head off potential problems. Some required characteristics, as well as a number of suggestions, are discussed here. These are grouped according to key points that should be addressed in all HACMP environments. This appendix covers the following application considerations:

- *Automation*—making sure your applications start and stop without user intervention
- *Dependencies*—knowing what factors outside HACMP affect the applications
- *Interference*—knowing that applications themselves can hinder HACMP functioning
- *Robustness*—choosing strong, stable applications
- *Implementation*—using appropriate scripts, file locations, and cron schedules.

You should add an application monitor to detect a problem with application startup. An application monitor in startup monitoring mode checks an application server's successful startup within the specified stabilization interval and exits after the stabilization period expires.

In HACMP 5.4.1, you can start the HACMP cluster services on the node(s) without stopping your applications, by selecting an option from the SMIT panel Manage HACMP Services > Start Cluster Services. When starting, HACMP relies on the application startup scripts and configured application monitors, to ensure that HACMP knows about the running application and does *not* start a second instance of the application.

For more information on configuring application monitoring and steps needed to start cluster services without stopping the applications, see the Administration Guide.

Similarly, you can stop HACMP cluster services and leave the applications running on the nodes. When the node that has been stopped and placed in an unmanaged state rejoins the cluster, the state of the resources is assumed to be the same unless a user initiates an HACMP resource group command to bring the resource group into another state (for example, online on an active node).

At the end of this appendix, you will find two examples of popular applications—Oracle Database™ and SAP R/3™—and some issues to consider when implementing these applications in an HACMP environment.

Application Automation: Minimizing Manual Intervention

One key requirement for an application to function successfully under HACMP is that the application be able to start and stop without any manual intervention.

Application Start Scripts

Create a start script that starts the application. The start script should perform any "clean-up" or "preparation" necessary to ensure proper startup of the application, and also properly manage the number of instances of the application that need to be started. When the Application Server is add to a Resource Group. HACMP calls this script to bring the application online as part of processing the Resource Group. Since the cluster daemons call the start script, there is no option for interaction. Additionally, upon an HACMP fallover, the recovery process calls this script to bring the application online on a standby node. This allows for a fully automated recovery, and is why any necessary cleanup and/or preparation should be included in this script.

HACMP calls the start script as the "root" user. It may be necessary to change to a different user in order to start the application. The **su** command can accomplish this. Also, it may be necessary to run **nohup** on commands that are started in the background and have the potential to be terminated upon exit of the shell.

For example, an HACMP cluster node may be a client in a Network Information Service (NIS) environment. If this is the case and you need to use the su command to change user id, there must be a route to the NIS master at all times. In the event that a route doesn't exist and the su is attempted, the application script hangs. You can avoid this by enabling the HACMP cluster node to be an NIS slave. That way, a cluster node has the ability to access its own NIS map files to validate a user ID.

The start script should also check for the presence of required resources or processes. This will ensure an application can start successfully. If the necessary resources are *not* available, a message can be sent to the administration team to correct this and restart the application.

Start scripts should be written so that they check if one instance of the application is already running and do *not* start another instance unless multiple instances are desired. Keep in mind that the start script may be run after a primary node has failed. There may be recovery actions necessary on the backup node in order to restart an application. This is common in database applications. Again, the recovery must be able to run without any interaction from administrators.

Also see the notes in the section Writing Effective Scripts.

Application Stop Scripts

The most important aspect of an application stop script is that it completely stop an application. Failure to do so many prevent HACMP from successfully completing a takeover of resources by the backup nodes. In stopping, the script may need to address some of the same concerns the start script addresses, such as NIS and the **su** command.

The application stop script should use a phased approach. The first phase should be an attempt to stop the cluster services and bring resource groups offline. If processes refuse to terminate, the second phase should be used to forcefully ensure all processing is stopped. Finally, a third phase can use a loop to repeat any steps necessary to ensure that the application has terminated completely.

Be sure that your application stop script exits with the value 0 (zero) when the application has been successfully stopped. In particular, examine what happens if you run your stop script when the application is already stopped. Your script must exit with zero in this case as well. If your stop script exits with a different value, this tells HACMP that the application is still running, although possibly in a damaged state. The event_error event will be run and the cluster will enter an error state. This check alerts administrators that the cluster is *not* functioning properly.

Keep in mind that HACMP allows 360 seconds by default for events to complete processing. A message indicating the cluster has been in reconfiguration too long appears until the cluster completes its reconfiguration and returns to a stable state. This warning may be an indication that a script is hung and requires manual intervention. If this is a possibility, you may wish to consider stopping an application manually before stopping HACMP.

You can change the time period before the **config_too_long** event is called. For more information about how to change this setting, see the section Tuning Event Duration Time Until Warning in Chapter 6: Configuring Cluster Events in the *Administration Guide*.

Application Start and Stop Scripts and Dependent Resource Groups

In HACMP 5.2 and up support for dependent resource groups lets you configure the following:

- Three levels of dependencies between resource groups, for example a configuration in which node A depends on node B, and node B depends on node C. HACMP prevents you from configuring circular dependencies.
- A type of dependency in which 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.

Note: If two applications must run on the same node, both applications must reside in the same resource group.

If a child resource group contains an application that depends on resources in the parent resource group, then upon fallover conditions, if the parent resource group falls over to another node, the child resource group is temporarily stopped and automatically restarted. Similarly, if the child resource group is concurrent, HACMP takes it offline temporarily on all nodes, and brings it back online on all available nodes. If the fallover of the parent resource group is *not* successful, both the parent and the child resource groups go into an ERROR state.

Note that when the child resource group is temporarily stopped and restarted, the application that belongs to it is also stopped and restarted. Therefore, to minimize the chance of data loss during the application stop and restart process, customize your application server scripts to ensure that any uncommitted data is stored to a shared disk temporarily during the application stop process and read back to the application during the application restart process. It is important to use a shared disk as the application may be restarted on a node other than the one on which it was stopped.

Application Tier Issues

Often, applications have a multi-tiered architecture (for example, a database tier, an application tier, and a client tier). Consider all tiers of an architecture if one or more is made highly available through the use of HACMP.

For example, if the database is made highly available, and a fallover occurs, consider whether actions should be taken at the higher tiers in order to automatically return the application to service. If so, it may be necessary to stop and restart application or client tiers. This can be facilitated in one of two ways. One way is to run clinfo on the tiers, the other is to use a remote execution command such as **rsh**, **rexec**, or **ssh**.

Note: Certain methods, such as the use of ~/**.rhosts** files, pose a security risk.

Using Dependent Resource Groups

To configure complex clusters with multi-tiered applications, you can use parent/child dependent resource groups. You may also want to consider using location dependencies. For more information, see the section Application Dependencies in this chapter, Chapter 2: Initial Cluster Planning, and Chapter 6: Planning Resource Groups.

Using the Clinfo API

clinfo is the cluster information daemon. You can write a program using the Clinfo API to run on any tiers that would stop and restart an application after a fallover has completed successfully. In this sense, the tier, or application, becomes "cluster aware," responding to events that take place in the cluster. See the manual *Programming Client Applications* for more detail on the Clinfo API.

Using Pre- and Post-Event Scripts

Another way to address the issue of multi-tiered architectures is to use pre- and post-event scripts around a cluster event. These scripts would call a remote execution command such as **rsh**, **rexec**, or **ssh** to stop and restart the application.

Application Dependencies

Prior to HACMP 5.2, to achieve resource group and application sequencing, system administrators had to build the application recovery logic in their pre- and post-event processing scripts. Every cluster would be configured with a pre-event script for all cluster events, and a post-event script for all cluster events.

Such scripts could become all-encompassing case statements. For example, if you want to take an action for a specific event on a specific node, you need to edit that individual case, add the required code for pre- and post-event scripts, and also ensure that the scripts are the same across all nodes.

To summarize, even though the logic of such scripts captures the desired behavior of the cluster, they can be difficult to customize and even more difficult to maintain later on, when the cluster configuration changes.

If you are using pre-and post-event scripts or other methods, such as resource group processing ordering to establish dependencies between applications that are supported by your cluster, then these methods may no longer be needed or can be significantly simplified. Instead, you can specify dependencies between resource groups in a cluster. This is especially true for HACMP 5.3, with improvements to default parallel processing and the addition of location dependencies.

For an overview and figures representing both parent/child and location dependencies, see the *Concepts and Facilities Guide.*

For planning multi-tiered applications, see Planning Considerations for Multi-Tiered Applications in Chapter 2: Initial Cluster Planning.

For information on how to configure resource group dependencies, see the *Administration Guide*.

Note: In many cases, applications depend on more than data and an IP address. For the success of any application under HACMP, it is important to know what the application should *not* depend upon in order to function properly. This section outlines many of the major dependency issues. Keep in mind that these dependencies may come from outside the HACMP and application environment. They may be incompatible products or external resource conflicts. Look beyond the application itself to potential problems within the enterprise.

Locally Attached Devices

Locally attached devices can pose a clear dependency problem. In the event of a fallover, if these devices are *not* attached and accessible to the standby node, an application may fail to run properly. These may include a CD-ROM device, a tape device, or an optical juke box. Consider whether your application depends on any of these and if they can be shared between cluster nodes.

Hard Coding

Hard coding an application to a particular device in a particular location, creates a potential dependency issue. For example, the console is typically assigned as /dev/tty0. Although this is common, it is by no means guaranteed. If your application assumes this, ensure that all possible standby nodes have the same configuration.

Hostname Dependencies

Some applications are written to be dependent on the AIX hostname. They issue a command in order to validate licenses or name file systems. The hostname is *not* an IP address label. The hostname is specific to a node and is *not* failed over by HACMP. It is possible to manipulate the hostname, or use hostname aliases, in order to trick your application, but this can become cumbersome when other applications, *not* controlled by HACMP, also depend on the hostname.

Software Licensing

Another possible problem is software licensing. Software can be licensed to a particular CPU ID. If this is the case with your application, it is important to realize that a fallover of the software will *not* successfully restart. You may be able to avoid this problem by having a copy of the software resident on all cluster nodes. Know whether your application uses software that is licensed to a particular CPU ID.

Application Interference

Sometimes an application or an application environment may interfere with the proper functioning of HACMP. An application may execute properly on both the primary and standby nodes. However, when HACMP is started, a conflict with the application or environment could arise that prevents HACMP from functioning successfully.

Software Using IPX/SPX Protocol

A conflict may arise between HACMP and any software that binds a socket over a network interface. An example is the IPX/SPX protocol. When active, it binds an interface and prevents HACMP from properly managing the interface. Specifically, for ethernet and token ring, it inhibits the Hardware Address Takeover from completing successfully. A “device busy” message appears in the HACMP logs. The software using IPX/SPX must be either completely stopped or *not* used in order for Hardware Address Takeover to work.

Products Manipulating Network Routes

Additionally, products that manipulate network routes can keep HACMP from functioning as it was designed. These products can find a secondary path through a network that has had an initial failure. This may prevent HACMP from properly diagnosing a failure and taking appropriate recovery actions.

AIX Fast Connect

You can reduce the problem of conflict with certain protocols, and the need for manual intervention, if you are using AIX Fast Connect to share resources. The protocols handled by this application can easily be made highly available because of their integration with HACMP.

AIX Fast Connect software is integrated with HACMP in a similar way, so that it can be configured as a highly available resource. AIX Fast Connect allows you to share resources between AIX workstations and PCs running Windows, DOS, and OS/2 operating systems. Fast Connect supports the NetBIOS protocol over TCP/IP.

For more information about configuring the application for HACMP, see the following resources:

- Chapter 2: Initial Cluster Planning
- The section Configuring AIX Fast Connect in Chapter 4: Configuring HACMP Cluster Topology and Resources (Extended) in the *Administration Guide*
- The appropriate planning worksheets in Appendix A: Planning Worksheets.

Robustness of Application

Of primary importance to the success of any application is the health, or robustness, of the application. If the application is unstable or crashing intermittently, resolve these issues before placing it in a high availability environment.

Beyond basic stability, an application under HACMP should meet other robustness characteristics, such as described in the following sections Successful Start after Hardware Failure and Survival of Real Memory Loss.

Successful Start after Hardware Failure

A good application candidate for HACMP should be able to restart successfully after a hardware failure. Run a test on an application before putting it under HACMP. Run the application under a heavy load and fail the node. What does it take to recover once the node is back online? Can this recovery be completely automated? If *not*, the application may *not* be a good candidate for high availability.

Survival of Real Memory Loss

For an application to function well under HACMP it should be able to survive a loss of the contents of real memory. It should be able to survive the loss of the kernel or processor state. When a node failure occurs, these are lost. Applications should also regularly check-point the data to disk. In the event that a failure occurs, the application will be able to pick up where it last check-pointed data, rather than starting completely over.

Application Implementation Strategies

There are a number of aspects of an application to consider as you plan for implementing it under HACMP. Consider characteristics such as time to start, time to restart after failure, and time to stop. Your decisions in a number of areas, including those discussed in this section—script writing, file storage, **/etc/inittab** file and cron schedule issues—can improve the probability of successful application implementation.

Writing Effective Scripts

Writing smart application start scripts can also help reduce the likelihood of problems when bringing applications online.

A good practice for start scripts is to check prerequisite conditions before starting an application. These may include access to a file system, adequate paging space and free file system space. The start script should exit and run a command to notify system administrators if the requirements are *not* met.

When starting a database it is important to consider whether there are multiple instances within the same cluster. If this is the case, start only the instances applicable for each node. Certain database startup commands read a configuration file and start all known databases at the same time. This may *not* be a desired configuration for all environments.

WARNING: Be careful *not* to kill any HACMP processes as part of your script. If you are using the output of the ps command and using a grep to search for a certain pattern, make sure the pattern does *not* match any of the HACMP or RSCT processes.

Considering File Storage Locations

Give thought to where the configuration files reside. They could either be on shared disk, and therefore potentially accessed by whichever node has the volume group varied on, or on each node's internal disks. This holds true for all aspects of an application. Certain files must be on shared drives. These include data, logs, and anything that could be updated by the execution of the application. Files such as configuration files or application binaries could reside in either location.

There are advantages and disadvantages to storing optional files in either location. Having files stored on each node's internal disks implies that you have multiple copies of, and potentially multiple licenses for, the application. This could require additional cost as well as maintenance in keeping these files synchronized. However, in the event that an application needs to be upgraded, the entire cluster need *not* be taken out of production. One node could be upgraded while the other remains in production. The "best" solution is the one that works best for a particular environment.

Considering /etc/inittab and cron Table Issues

Also give thought to applications, or resources needed by an application, that either start out of the **/etc/inittab** file or out of the cron table. The **inittab** starts applications upon boot up of the system. If cluster resources are needed for an application to function, they will *not* become available until after HACMP is started. It is better to use the HACMP application server facility that allows the application to be a resource that is started only after all dependent resources are online.

Note: It is very important that the following be correct in **/etc/inittab**:

```
hacmp:2:once:/usr/es/sbin/cluster/etc/re.init "a"
```

- The **clinit** and **pst_clinit** entries must be the last entries of run level "2".
- The **clinit** entry must be before the **pst_clinit** entry.
- Any **tty** entries must *not* be "on" or "respawn".

An incorrect entry for these prevents HACMP from starting.

In the cron table, jobs are started according to a schedule set in the table and the date setting on a node. This information is maintained on internal disks and thus cannot be shared by a standby node. Synchronize these cron tables so that a standby node can perform the necessary action at the appropriate time. Also, ensure the date is set the same on the primary node and any of its standby nodes.

Examples: Oracle Database™ and SAP R/3™

Here are two examples illustrating issues to consider in order to make the applications Oracle Database and SAP R/3 function well under HACMP.

Example 1: Oracle Database

The Oracle Database, like many databases, functions very well under HACMP. It is a robust application that handles failures well. It can roll back uncommitted transactions after a fallover and return to service in a timely manner. However, there are a few things to keep in mind when using Oracle Database under HACMP.

Starting Oracle

Oracle must be started by the Oracle user ID. Thus, the start script should contain an su - oracleuser. The dash (-) is important since the su needs to take on all characteristics of the Oracle user and reside in the Oracle user's home directory. The command would look something like this:

```
su - oracleuser -c “/apps/oracle/startup/dbstart”
```

Commands like **dbstart** and **dbshut** read the **/etc/oratabs** file for instructions on which database instances are known and should be started. In certain cases it is inappropriate to start all of the instances, because they may be owned by another node. This would be the case in the mutual takeover of two Oracle instances. The **oratabs** file typically resides on the internal disk and thus cannot be shared. If appropriate, consider other ways of starting different Oracle instances.

Stopping Oracle

The stopping of Oracle is a process of special interest. There are several different ways to ensure Oracle has completely stopped. The suggested sequence is this: first, implement a graceful shutdown; second, call a shutdown immediate, which is a bit more forceful method; finally, create a loop to check the process table to ensure all Oracle processes have exited.

Oracle File Storage

The Oracle product database contains several files as well as data. It is necessary that the data and redo logs be stored on shared disk so that both nodes may have access to the information. However, the Oracle binaries and configuration files could reside on either internal or shared disks. Consider what solution is best for your environment.

Example 2: SAP R/3, a Multi-Tiered Application

SAP R/3 is an example of a three-tiered application. It has a database tier, an application tier, and a client tier. Most frequently, it is the database tier that is made highly available. In such a case, when a fallover occurs and the database is restarted, it is necessary to stop and restart the SAP application tier. You can do this in one of two ways:

- Using a remote execution command such as **rsh**, **rexec**, or **ssh**

  Note: Certain methods, such as the use of **~/.rhosts** files, pose a security risk.

- Making the application tier nodes “cluster aware.”

Using a Remote Execution Command

The first way to stop and start the SAP application tier is to create a script that performs remote command execution on the application nodes. The application tier of SAP is stopped and then restarted. This is done for every node in the application tier. Using a remote execution command requires a method of allowing the database node access to the application node.

Note: Certain methods, such as the use of ~/**.rhosts** files, pose a security risk.

Making Application Tier Nodes “Cluster Aware”

A second method for stopping and starting the application tier is to make the application tier nodes “cluster aware.” This means that the application tier nodes are aware of the clustered database and know when a fallover occurs. You can implement this by making the application tier nodes either HACMP servers or clients. If the application node is a server, it runs the same cluster events as the database nodes to indicate a failure; pre- and post-event scripts could then be written to stop and restart the SAP application tier. If the application node is an HACMP client, it is notified of the database fallover via SNMP through the cluster information daemon (clinfo). A program could be written using the Clinfo API to stop and restart the SAP application tier.

Consult the manual *Programming Client Applications* for more detail on the Clinfo API.

Notices for HACMP Planning 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