IBM

Implementing High Availability Cluster Multi-Processing (HACMP) Cookbook

- Extended case studies with practical disaster recovery examples
- Explore the latest HACMP and HACMP/XD V5.3 features
- Advanced POWER virtualization explained

Octavian Lascu
Shawn Bodily
Maria-Katharina Esser
Michael Herrera
Patrick Pothier
Dusan Prelec
Dino Quintero
Ken Raymond
Viktor Sebesteny
Andrei Socoliuc
Antony (Red) Steel

Redbooks

ibm.com/redbooks

International Technical Support Organization

Implementing High Availability Cluster Multi-Processing (HACMP) Cookbook

December 2005

SG24-6769-00

Note: Before using this information and the product it supports, read the information in “Notices” on page xiii.

First Edition (December 2005)

This edition applies to Version 5, Release 3, of IBM High Availability Cluster Multi-Processing (product number 5765-F62).

Contents

Notices

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 IBM 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.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS 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.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

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.

COPYRIGHT LICENSE:
This information contains sample application programs in source language, which illustrates programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM's application programming interfaces.

Trademarks

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

@server®	Enterprise Storage Server®	Redbooks™
Redbooks (logo)  ™	FlashCopy®	Requisite®
ibm.com®	HACMP™	RS/6000®
pSeries®	IBM®	Seascape®
AIX 5L™	Magstar®	Tivoli®
AIX®	MVS™	TotalStorage®
Cross-Site®	NetView®	WebSphere®
DB2 Universal Database™	POWER4™	
DB2®	POWER5™	

The following terms are trademarks of other companies:

IPC, Java, Solaris, Sun, Ultra, and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Microsoft, Windows, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

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

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

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

Preface

This IBM® Redbook will help you install, tailor and configure the new HACMP™ V5.3, and understand the new and improved features like Dynamic LPAR integration, Virtual I/O, and Disaster Recovery (DR) configurations.

This redbook gives a broad understanding of the HACMP and HACMP Extended Distance (HACMP/XD) architecture. If you plan to install, migrate, or merely administer a high availability cluster, this book is right for you. Disaster recovery elements and how HACMP fulfills these necessities are also presented in detail.

This cookbook helps AIX® professionals that are seeking a comprehensive and task-oriented guide for developing the knowledge and skills required for HACMP cluster design and implementation as well as for daily system administration. It is designed to provide a combination of theory and practical experience.

This book will be especially useful for system administrators currently running both HACMP/ES and HACMP Extended Distance (XD) clusters who may want to consolidate their environment and move to a new HACMP V5.3. There is a detailed description of a node-by-node migration to HACMP/ES 5.3 and a comprehensive discussion about how to prepare for an upgrade or migration.

The team that wrote this redbook

This redbook was produced by a team of specialists from around the world working at the International Technical Support Organization, Austin Center.

Octavian Lascu is a Project Leader at the International Technical Support Organization, Poughkeepsie Center, and has worked with IBM since 1992. He writes extensively and teaches IBM classes worldwide in all areas of pSeries® and Linux® clusters. Before joining the ITSO, Octavian worked in IBM Global Services Romania as a software and hardware Services Manager. He holds a master's degree in Electronic Engineering from the Polytechnical Institute in Bucharest and is also an IBM Certified Advanced Technical Expert in AIX/PSSP/HACMP.

Shawn Bodily is a Senior IT Specialist for ATS Americas in Dallas, Texas. He has worked for IBM for seven years, and has ten years of AIX experience and seven years specializing in HACMP. He is HACMP and ATE certified in both V4 and V5. He has written and presented on high availability and storage. He has co-authored two Redbooks™.

Maria-Katharina Esser is an IT Specialist for pre-sales technical support and works for the IBM System and Technology Group (STG) in Munich, Germany. She has worked for IBM for 17 years, and has six years of experience in AIX, RS/6000®, and IBM @server® pSeries. Her areas of expertise include storage and HACMP.

Michael Herrera is a Staff Software Engineer at the IBM AIX Software Support Center in Dallas, Texas. He has seven years of experience in AIX, RS/6000, SAN and pSeries support. He holds a bachelor's degree in Management Information Systems from the University of Connecticut and an MBA from the University of Dallas. He specializes in AIX/HACMP and SAN environments and is certified by IBM as an Advanced Technical Expert.

Patrick Pothier is an IT Specialist in STG France in FTSS pSeries for five years. He has 13 years experience in the UNIX® field and 11 years experience in HACMP. His areas of experience include operating systems (MVS™, AIX, Solaris™, Linux), high availability (HACMP), backup solutions (ADSM/6000, Tivoli® Storage Manager, and Netbackup), and ERP administration (SAP R/3 BC).

Dusan Prelec is a pSeries Support Specialist in IBM Global Services, Slovenia. He has worked in AIX and HACMP Support for CEE (Central and Eastern Europe) for three years. He is involved in various cluster and storage implementation projects in Slovenia and in CEE countries. He is an IBM Certified AIX and HACMP Specialist, and teaches HACMP classes. Before joining the IBM Global Services team, he worked in IBM CEMA pSeries Technical Pre-sales Support.

Dino Quintero is a Senior Certified Consulting IT Specialist at ITSO in Poughkeepsie, New York. Before joining ITSO, he worked as a Performance Analyst for the Enterprise Systems Group and as a Disaster Recovery Architect for IBM Global Services. His areas of expertise include disaster recovery and pSeries clustering solutions. He is certified in pSeries system administration and pSeries clustering technologies. He is also an IBM Senior Certified Professional in pSeries technologies. Currently, he leads technical teams delivering IBM Redbook solutions in pSeries clustering technologies and delivering technical workshops worldwide.

Ken Raymond is a Senior IT Specialist working for IBM Global Services in Ottawa, Canada. He has over 26 years of experience with IBM, the past 10 years working extensively with AIX customers in the areas of systems management and high availability. His areas of expertise include AIX systems management and technical support, pSeries (logical partitions and virtualization) solution design, AIX implementation and migration support, clustering (HACMP, CSM, PSSP) implementation and support, and project management services.

Viktor Sebesteny is an IT Specialist in Hungary. He has worked for IBM Global Services for nine years. He holds a bachelor's degree in Computer Science from KKMF University. His areas of expertise include pSeries, AIX, HACMP and CSM.

Andrei Socoliuc is a Software Support Engineer in IBM Global Services in Romania. He holds a master's degree in Computer Science from Polytechnic Institute in Bucharest, Romania. He has six years of experience in the AIX, pSeries and clustering. His areas of expertise include AIX, PSSP, HACMP, TSM, and Linux. He has written extensively about pSeries clusters and PSSP.

Antony (Red) Steel is a Senior IT Specialist in ITS Australia. He has 12 years experience in the UNIX field, predominately AIX and Linux. He holds an honors degree in Theoretical Chemistry from the University of Sydney. His areas of expertise include scripting, system customization, performance, networking, high availability and problem solving. He has written and presented about LVM, TCP/IP, and high availability both in Australia and throughout the Asia Pacific region. He has co-authored three IBM Redbooks.

Thanks to the following people for their contributions to this project:

Michael K. Coffey
IBM Poughkeepsie

Paul Moyer
IBM Poughkeepsie

Tomas Weaver
IBM Austin

David Truong
IBM Dallas

Skip Russell
IBM Poughkeepsie

Steve Tovcimak
IBM Poughkeepsie

Elaine Krakower
IBM Poughkeepsie

Geoffrey Mattes
IBM Australia

Gabrielle Velez
International Technical Support Organization

Become a published author

Join us for a two- to six-week residency program! Help write an IBM Redbook dealing with specific products or solutions, while getting hands-on experience with leading-edge technologies. You'll team with IBM technical professionals, Business Partners and/or customers.

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you'll develop a network of contacts in IBM development labs, and increase your productivity and marketability.

Find out more about the residency program, browse the residency index, and apply online at:

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

We want our Redbooks to be as helpful as possible. Send us your comments about this or other Redbooks in one of the following ways:

- Use the online **Contact us** review redbook form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. JN9B Building 905
  11501 Burnet Road
  Austin, Texas 78758-3493

Part 1

Introduction

Part 1 provides an overview of HACMP and describes HACMP components as part of a successful implementation. As HACMP is a mature product, we consider that it is important to present some of the recent HACMP history, which will help for planning future actions, such as migrating existing configurations to the latest version and exploiting the new features in HACMP V5.2 and V5.3.

We also introduce the basic HACMP management concepts, with recommendations and considerations to ease the system administrator's job.

1

Introduction to HACMP

This chapter provides an introduction to high availability in general and discusses the IBM HACMP in particular:

- What High Availability Cluster Multi-Processing (HACMP) is
- Availability solutions
- History and evolution
- High availability terminology and concepts
- High availability versus fault tolerance
- Planning software for an installation
- HACMP installation considerations

1.1 What HACMP is

HACMP stands for High Availability Cluster Multi-Processing. The main components are high availability and multi-processing in a cluster (multi-node) environment.

1.1.1 High availability

In today's complex environments, providing continuous service for applications is a key component of a successful IT implementation. High availability is one of the components that contributes to providing continuous service for the application clients, by masking or eliminating both planned and unplanned systems and application downtime. This is achieved through the elimination of hardware and software single points of failure (SPOF). A high availability solution will ensure that the failure of any component of the solution, either hardware, software, or system management, will not cause the application and its data to become permanently unavailable to the end user.

High availability solutions should eliminate single points of failure through appropriate design, planning, selection of hardware, configuration of software, control of applications and carefully controlled environment and change management discipline.

In short, we can define the high availability as the process of ensuring an application is available for use through the use of duplicated and/or shared hardware resources managed by a specialized software component.

1.1.2 Cluster multi-processing

In addition to the high availability, HACMP also provides the multi-processing component. The multi-processing capability comes from the fact that in a cluster there are multiple hardware and software resources managed by HACMP to provide complex application functionality and better resource utilization.

As a short definition for cluster multi-processing can be: multiple applications running over a number of nodes with shared or concurrent access to the data.

Although desirable, the cluster multi-processing component depends on the application capabilities and system implementation to efficiently use all resources available in a multi-node (cluster) environment. This must be implemented starting with the cluster planning and design phase.

HACMP is only one of the high availability technologies and builds on the increasingly more reliable operating systems, more reliable and hot swappable

hardware, increasingly more resilient applications, by offering monitoring and automated response.

A high availability solution based on HACMP provides automated failure detection, diagnosis, application recovery, and node reintegration. With an appropriate application, HACMP can also provide concurrent access to the data for parallel processing applications, thus offering excellent horizontal, and vertical scalability (with the addition of the dynamic LPAR management capabilities).

IBM has also designed an extended version of HACMP which provides disaster recovery functionality integrated is a solution known as HACMP Extended Distance (HACMP/XD) which supports HACMP functionality between two geographic sites. HACMP/XD supports a number of distinct methods for replicating the data and is discussed in detail in Chapter 15, “HACMP Extended distance concepts and planning” on page 613.

1.2 Availability solutions - overview

There are a range of solutions that provide a wide range of availability options. In Table 1-1, we describe different types of availability solutions and their characteristics.

Table 1-1 Types of availability solutions

Solution	Downtime measured in	Data availability	Observations
Standalone	Days	From last backup	Basic hardware and software costs ($)
Enhanced standalone	Hours	Till last transaction	Double basic hardware cost ($$)
High availability clusters	Minutes	Till last transaction	Double hardware and additional services ($$+)
Fault-tolerant computing	Never stops	No loss of data	Specialized hardware and software, very expensive ($$$$$$)
HACMP/XD	Minutes	Till last transaction	Two-three times the hardware cost + additional communication($$$$)

High availability solutions in general offer the following benefits:

- Standard hardware and networking components (can be used with the existing hardware)
- Works with just about any application (it only depends on the implementor’s ability).

- Works with a wide range of disk and network types
- Excellent availability at reasonable cost

IBM's high available solution for the IBM @server® pSeries offers distinct benefits that include:

- Proven solution (more than 15 years of product development)
- Flexibility (virtually any application running on a standalone AIX system can be protected with HACMP)
- Using "of the shelf" hardware components (pSeries)
- Proven commitment for supporting our customers

When planning to implement an HACMP solution, the following aspects have to be considered:

- Thorough design and detailed planning
- Elimination of single points of failure
- Selection of appropriate hardware
- Correct implementation (do NOT take "shortcuts")
- Disciplined system administration practices and change control
- Documented operational procedures
- Comprehensive test plan and thorough testing

A typical HACMP environment is shown in Figure 1-1 on page 7.

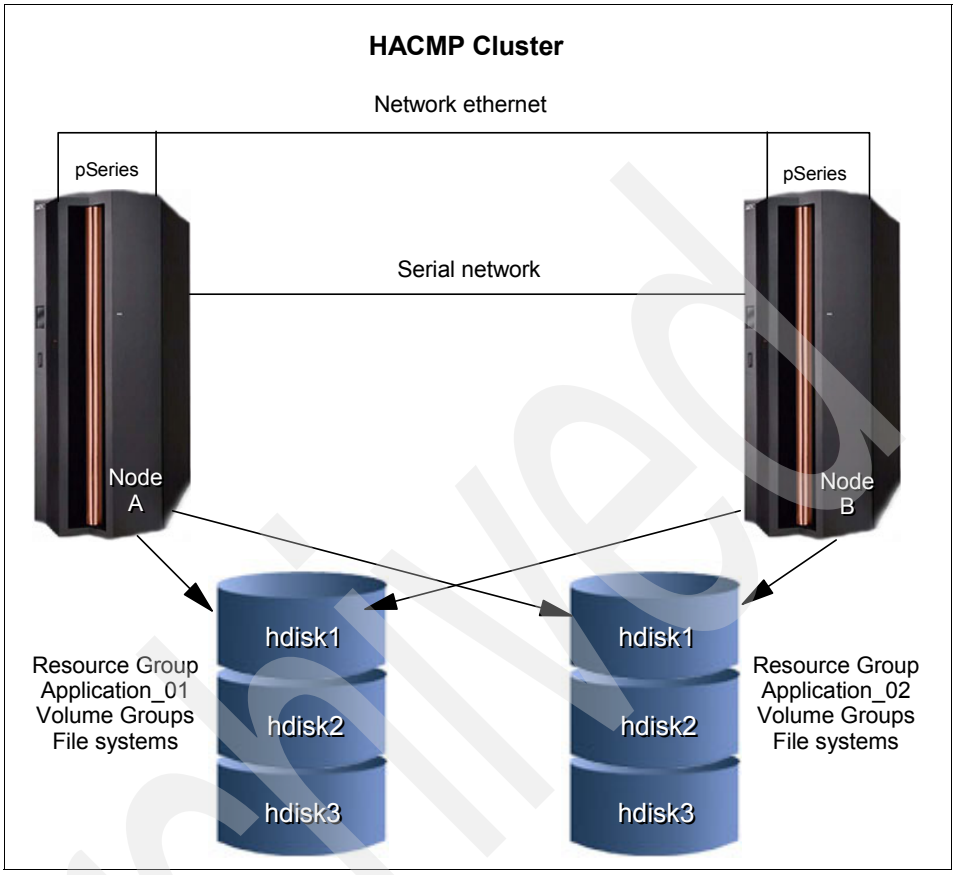

Figure 1-1 HACMP cluster

1.2.1 Downtime

The downtime is the period when an application is not available to serve its clients. We can classify downtime in two categories:

- Planned:
  - Hardware upgrades
  - Repairs
  - Software updates/upgrades
  - Backups (offline backups)
  - Testing (periodic testing is required for cluster validation.)
  - Development

- Unplanned:
  - Administrator errors
  - Application failures
  - Hardware failures
  - Operating system errors
  - Environmental disasters

Thus the role of HACMP is to both maintain application availability through the unplanned outages and normal day to day administrative requirements. HACMP provides monitoring and automatic recovery of the resources on which your application depends.

1.2.2 Single point of failure (SPOF)

A single point of failure (SPOF) is any individual component integrated in a cluster which, in case of failure, renders the application unavailable for end users.

Good design will remove single points of failure in the cluster - nodes, storage, networks. HACMP manages these, as well as managing the resources required by the application (including the application start/restart scripts)

Ultimately, the goal of any IT solution in a critical environment is to provide continuous application availability and data protection. The high availability is just one building block in achieving the continuous operation goal. The high availability is based on the availability of the hardware, software (operating system and its components), application, and network components.

For a avoiding SPOFs you need:

- Redundant servers
- Redundant network paths
- Redundant storage (data) paths
- Redundant (mirrored/RAID) storage
- Monitoring
- Failure detection and diagnosis
- Automated application fallover
- Automated resource reintegration

As previously mentioned, a good design is able to avoid single points of failure, and HACMP will manage the availability of the application through downtimes. Table 1-2 on page 9 lists each Cluster Object, which, if it fails, can result in loss of

availability of the application. Each Cluster Object can be a physical or logical component

Table 1-2 Single point of failure

Cluster Object	Single Point of Failure eliminated by:
Node (Servers)	Multiple Nodes
Power Supply	Multiple circuits and/or power supplies and/or UPS
Network Adapter	Redundant Network Adapters
Network	Multiple networks connected to each nodes, redundant network paths with independent hardware between each node and the clients.
TCP/IP Subsystem	Use point-to-point networks to connect each node to it's neighbor in a ring.
I/O Adapter	Redundant I/O Adapters
Controllers	User redundant controllers
Storage	Redundant hardware, enclosures, disk mirroring / RAID technology, redundant data paths
Application	Configuring application monitoring and backup node(s) to acquire the application engine and data
Sites	Use more than 1 site for disaster recovery
Resource Groups	Use resource groups to control all the resource that an application requires

HACMP also optimizes availability by allowing for dynamic reconfiguration of running clusters. Maintenance tasks such as adding or removing nodes can be performed without stopping and restarting the cluster.

In addition, other management tasks, such as modifying storage, managing users, can be performed on the running cluster using the *Cluster Single Point of Control (C-SPOC)* without interrupting user access to application running on *cluster nodes.* C-SPOC also ensures that changes made on one node are replicated across the cluster in a consistent manner.

1.3 History and evolution

IBM High Availability Cluster Multi-Processing goes back to the early 1990s. HACMP development started in 1990 to provide high availability solution for applications running on RS/6000 servers. We do not provide information about the very early releases, since those releases are nor supported or in use at the time this book was developed, instead, we provide highlights about the most recent versions.

Originally designed as a standalone product (known as HACMP “classic”), once the IBM high availability infrastructure known as Reliable Scalable Clustering Technology (RSCT) became available, HACMP adopted this technology and HACMP Enhanced Scalability (HACMP/ES), as it provides performance and functional advantages over the “classic” version.

1.3.1 HACMP Version 4

HACMP V4.2.2	Along with HACMP Classic (HAS), this version introduced the enhanced scalability version (ES) based on RSCT (Reliable Scalable Clustering Technology) topology, group, and event management services, derived from PSSP (Parallel Systems Support Program).
HACMP V4.3.X	This version introduced, among other aspects, 32 node support for HACMP/ES, C-SPOC enhancements, ATM network support, HACMP Task guides (GUI for simplifying cluster configuration), multiple pre- and post-event scripts, FDDI MAC address takeover, monitoring and administration support enhancements, node by node migration, and AIX fast connect support.
HACMP V4.4.X	New items in this version are integration with Tivoli, application monitoring, cascading with out fallback, C-SPOC enhancements, improved migration support, integration of HA-NFS functionality, and soft copy documentation (HTML and PDF).
HACMP V4.5	In this version, AIX 5L is required, and there is an automated configuration discovery feature, multiple service labels on each network adapter (through the use of IP aliasing), persistent IP address support, 64-bit-capable APIs, and monitoring and recovery from loss of volume group quorum.

1.3.2 HACMP Version 5 Release 1

This is the version that introduced major changes, from configuration simplification and performance enhancements to changing HACMP terminology. Some of the important new features in HACMP V5.1 were:

- HACMP “classic” (HAS) has been dropped; only HACMP/ES was available, based on IBM Reliable Scalable Cluster Technology
- SMIT “Standard” and “Extended” configuration paths (procedures)
- Automated configuration discovery
- Custom resource groups
- Non IP networks based on heartbeating over disks
- Fast disk takeover
- Forced varyon of volume groups
- Heartbeating over IP aliases
- Improved security, by using cluster communication daemon (eliminating the need of using AIX “r” commands, thus eliminating the need for the /.rhosts file)
- Improved performance for cluster configuration and synchronization
- Normalization of HACMP terminology (aligning it with other HA products)
- Simplification of configuration and maintenance
- Online Planning Worksheets enhancements
- Forced varyon of volume groups
- Custom resource groups
- Heartbeat monitoring of service IP addresses/labels on takeover node(s)
- Heartbeating over IP aliases
- Heartbeating over disks
- Various C-SPOC enhancements
- GPFS integration
- Fast disk takeover
- Cluster verification enhancements
- Improved resource group management

1.3.3 HACMP Version 5 Release 2

Introduced in July 2004, HACMP V5.2 added more improvements in manage-

ment, configuration simplification, automation, and performance areas. Here is a summary of the improvements in HACMP V5.2:

- Two-Node Configuration Assistant, with both SMIT menus and a Java™ interface (in addition to the SMIT “Standard” and “Extended” configuration paths).
- File collections
- User password management
- Classic resource groups are not used anymore, replaced by custom resource groups
- Automated test procedures
- Automatic cluster verification
- Improved Online Planning Worksheets (OLPW) can now import a configuration from an existing HACMP cluster
- Event management (EM) has been replaced by resource monitoring and a control (RMC) subsystem (standard in AIX)
- Enhanced security
- Resource group dependencies
- Self-healing clusters (correcting certain cluster configuration errors)
- HACMP Smart Assist for WebSphere® Application Server

1.3.4 HACMP Version 5 Release 3

Starting July 2005, the new HACMP V5.3 continued the development of HACMP, by adding further improvements in management, configuration simplification, automation, and performance areas. Here is a summary of the improvements in HACMP V5.3:

- Cluster verification at cluster startup
- Additional corrective actions taken during verification
- **clverify** warns of recognizable single points of failure
- **clverify** integrates HACMP/XD options - PPRC; GeoRM; GLVM
- **clverify** automatically populates the clhosts file
- XML file format for OLPW files and ability to convert existing snapshot files into XML cluster configuration files
- OEM volume and file system support
- Further integration of HACMP with RSCT

- More ‘Smart Assist’ options - DB2® and Oracle Application Server
- Removal of certain site related restrictions from HACMP
- Location dependency added for Resource Groups
- WebSMIT security improved by:
  - client data validation before any HACMP commands are executed
  - Server side validation of parameters
  - WebSMIT authentication tools integrated with the AIX authentication mechanisms
- Cluster manager (**clstrmgrES**) daemon running at all times (regardless of cluster status - up or down) to support further automation of cluster configuration and enhanced administration
- Cluster multi-peer extension daemon (**clsmuxpdES**) and cluster information daemon (**clinfoES**) shared memory removed

Note: At the time of publication, HACMP 5.1, 5.2 and 5.3 were available, but most of testing for this redbook was done using HACMP V5.3

1.4 High availability terminology and concepts

To understand the correct functionality of HACMP and to utilize it effectively, it is necessary to understand some important terms and concepts:

1.4.1 Terminology

Starting in HACMP V5.1, the terminology used to describe HACMP configuration and operation has changed dramatically. The reason for this change is to simplify the overall usage and maintenance of HACMP, and also to align the terminology with the IBM product line.

For example, in previous HACMP versions, the term “Adapter”, depending on the context, could have different meanings, which made configuration confusing and difficult.

The following terms will be used throughout this book:

Cluster Loosely-coupled collection of independent systems (nodes) or Logical Partitions (LPARs) organized into a network for the purpose of sharing resources and communicating with each other.
HACMP defines relationships among cooperating systems where peer cluster nodes provide the services offered by a

cluster node should that node be unable to do so.
These individual nodes are together responsible for maintaining the functionality of one or more applications in case of a failure of any cluster component.

Node An IBM @server pSeries machine (or LPAR) running AIX and HACMP that is defined as part of a cluster. Each node has a collection of resources (disks, file systems, IP addresses, and applications) that can be transferred to another node in the cluster in case the node or a component fails.

Clients A client is a system that can access the application running on the cluster nodes over a local area network. Clients run a client application that connects to the server (node) where the application runs.

1.4.2 Concepts

The basic concepts of HACMP can be classified as follows:

Topology Contains basic cluster components nodes, networks, communication interfaces, communication devices, and communication adapters.

Resources Logical components or entities that are being made highly available (for example, file systems, raw devices, service IP labels, and applications) by being moved from one node to another. All the resources that together form a highly available application or service, are grouped together in resource groups (RG). HACMP keeps the RG highly available as a single entity that can be moved from node to node in the event of a component or node failure.
Resource groups can be available from a single node or, in the case of concurrent applications, available simultaneously from multiple nodes.
A cluster may host more than one resource group, thus allowing for efficient use of the cluster nodes (thus the “Multi-Processing” in HACMP).

Service IP label
A label that matches to a service IP address and is used for communications between clients and the node. A service IP label is part of a resource group, which means that HACMP will monitor it and keep it highly available.

IP address takeover
The process whereby an IP address is moved from one adapter to another adapter on the same logical network. This adapter

may be on the same node, or another node in the cluster. If aliasing is used as the method of assigning addresses to adapters, then more than one address may reside on a single adapter.

Resource takeover
This is the operation of transferring resources between nodes inside the cluster. If one component or node fails due to a hardware or operating system problem, its resource groups will be moved to the another node.

Fallover Represents the movement of a resource group from one active node to another node (backup node) in response to a failure on that active node.

Fallback Represents the movement of a resource group back from the backup node to the previous node, when it becomes available. This movement is typically in response to the reintegration of the previously failed node.

Heartbeat packet
A packet sent between communication interfaces in the cluster, used by the various cluster daemons to monitor the state of the cluster components - nodes, networks, adapters.

RSCT daemons
This consists of two processes (topology and group services) that monitor the state of the cluster and each node. The cluster manager receives event information generated by these daemons and takes corresponding (response) actions in case of failure(s).

Group Leader
The node with the highest IP as defined in one of the HACMP networks (the first network available), that acts as the central repository for all topology and group data coming from the RSCT daemons concerning the state of the cluster.

Group leader backup
This is the node with the next highest IP address on the same arbitrarily chosen network, that acts as a backup for the group Leader - and will take over in the event that the group Leader leaves the cluster.

Mayor A node chosen by the RSCT Group Leader (the node with the next highest IP address after the GL Backup), if such exists, else it is the GL Backup itself. It is the mayor's responsibility to inform other nodes of any changes in the cluster as determined by the group leader (GL)

Note: Earlier versions of HACMP used to refer to a RG move in response to a failure of a component on one node as a ‘failover’.

These concepts are described in detail in Chapter 2, “High availability components” on page 37.

1.5 High availability versus fault tolerance

Based on the response time and response action to system detected failures, the clusters and systems can be classified as:

- Fault-tolerant
- High availability

1.5.1 Fault-tolerant systems

The systems provided with fault tolerance are designed to operate virtually without interruption, regardless of the failure that may occur (except perhaps for a complete site down due to a natural disaster). In such systems, ALL components are at least duplicated for both software or hardware.

This all components, CPUs, memory, and disks have a special design and provide continuous service, even if one sub-component fails. Only special software solutions will run on fault tolerant hardware.

Such systems are very expensive and extremely specialized. Implementing a fault tolerant solution requires a lot of effort and a high degree of customization for all system components.

For environments where *no* downtime is acceptable (life critical systems), fault-tolerant equipment and solutions are required.

1.5.2 High availability systems

The systems configured for high availability are a combination of hardware and software components configured to work together to ensure automated recovery in case of failure with a minimal acceptable downtime.

In such systems, the software involved detects problems in the environment, and manages application survivability by restarting it on the same or on another available machine (taking over the identity of the original machine - node).

Thus, it is very important to eliminate all single points of failure (SPOF) in the environment. For example, if the machine has only one network interface (connection), a second network interface (connection) should be provided in the same node to take over in case the primary interface providing the service fails.

Another important issue is to protect the data by mirroring and placing it on shared disk areas accessible from any machine in the cluster.

The HACMP (High Availability Cluster Multi-Processing) software provides the framework and a set of tools for integrating applications in a highly available system.

Applications to be integrated in a HACMP cluster require a fair amount of customization, not at the application level, but rather at the HACMP and AIX platform level.

HACMP is a flexible platform that allows integration of generic applications running on AIX platform, providing for high available systems at a reasonable cost.

It is important to remember that HACMP is not a fault tolerant solution and should never be implemented as such.

1.6 Software planning

In the process of planning a HACMP cluster, one of the most important steps is to chose the software levels that will be running on the cluster nodes.

The decision factors in node software planning are:

- Operation system requirements: AIX version and recommended levels.
- Application compatibility: Ensure that all requirements for the applications are met, and supported in cluster environments.
- Resources: Types of resources that may be used (IP addresses, storage configuration, if NFS is required, and so on).

1.6.1 AIX level and related requirements

Before you install the HACMP, you must check the other software level requirements.

Table 1-3 on page 18 shows the recommended HACMP and other software levels at the time this redbook was written.

Table 1-3 OS level requirements for HACMP V5.1 and V5.2

HACMP Version	AIX OS Level and other software	AIX APARs	RSCT Level
HACMP V5.1	5100-05	IY50579, IY48331	2.2.1.30 or higher
HACMP V5.1	5200-02	IY48180, IY44290	2.3.1.0 or higher
HACMP V5.2	5100-06	IY54018, IY53707, IY54140, IY55017	2.2.1.30 or higher
HACMP V5.2	5200-03	IY56213	2.3.3.0 or higher
HACMP V5.3	5200-04	IY72082, IY72946, IY72928	2.3.6 or higher
HACMP V5.3	5300-02	IY71500, 72852, IY72916, IY72928	2.4.2 or higher
HACMP 5.3, CBU, DLPAR, CuOD	5.2 / 5.3	IY73050, IY73051	
HACMP/XD:HAGeo -		no additional requisites	
HACMP/XD:GLVM	5200-04	IY66555	
HACMP/XD:GLVM	5300-02	IY68029, IY68300	

For the latest list of recommended maintenance levels for HACMP V5.1, 5.2 and V5.3, access the IBM Web site at:

`http://www-912.ibm.com/eserver/support/fixes/fcgui.jsp`

Note:

- To use C-SPOC with VPATH disks, Subsystem Device Driver (SDD) 1.3.1.3 or later is required.
- To use HACMP Online Planning Worksheets, AIX 5L Java Runtime Environment 1.3.1 or later and a graphics display (local or remote) are required.
- HACMP V5.1 and V5.2 support the use of AIX 5L V5.2 Multi-path I/O (MPIO) device drivers for accessing disk subsystems.

For HACMP/XD using ESS/PPRC

- AIX 5L Java 1.3.0.13 or later

- ESS microcode 2.1.1 or later
- 2105 command line interface (ibm2105cli.rte.32.6.200.13 or ibm2105esscli.rte.2.1.0.15)
- IBM 2105 subsystem device driver (ibmSdd_510nchacmp.rte 1.3.3.6 or above)
- ESS eRCMF V2.0 for HACMP/XD for eRCMF

For HACMP/XD using SVC/PPRC

- openssh version 3.6.1 or later
- IBM 2145 subsystem device driver -devices.fcp.disk.ibm.rte (1.0.0.0), devices.sdd.5.2.rte, devices.fcp.disk.ibm2145.rte

The following AIX base operating system (BOS) components are prerequisites for HACMP:

- bos.adt.lib
- bos.adt.libm
- bos.adt.syscalls
- bos.net.tcp.client
- bos.net.tcp.server
- bos.rte.SRC
- bos.rte.libc
- bos.rte.libcfg
- bos.rte.libcur
- bos.rte.libpthreads
- bos.rte.odm
- bos.data

When using the (enhanced) concurrent resource manager access, the following components are also required:

- bos.rte.lvm.5.1.0.25 or higher (for AIX 5L V5.1)
- bos.clvm.enh (as required by the LVM)

For the complete list of recommended maintenance levels for AIX 5L V5.1 and V5.2, see the following IBM Web page:

`http://www-912.ibm.com/eserver/support/fixes/fcgui.jsp`

1.6.2 Licensing

Most software vendors require that you have a unique license for each application for each physical machine or per processor in a multi-processor (SMP) machine. Usually, the license activation code is entered at installation time.

However, in a HACMP environment, in a takeover situation, if the application is restarted on a different node, you must make sure that you have the necessary activation codes (licenses) for the new machine; otherwise the application may not start properly.

The application may also require a unique node-bound license (a separate license file on each node).

Some applications also have restrictions with the number of floating licenses available within the cluster for that application. To avoid this problem, be sure that you have enough licenses for each cluster node machine, so the application can run simultaneously on multiple nodes (especially for concurrent applications).

1.7 HACMP software installation

The HACMP software provides a series of facilities that you can use to make your applications highly available. You must keep in mind that not all system or application components are protected by HACMP.

For example, if all the data for a critical application resides on a single disk, and that specific disk fails, then that disk is a single point of failure for the entire cluster, and is *not* protected by HACMP. AIX logical volume manager or storage subsystems protection must be used in this case. HACMP only provides takeover for the disk on the backup node, to make the data available for use.

This is why HACMP planning is so important, because 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.

Also keep in mind that a well-planned cluster is easy to install, provides higher application availability, performs as expected, and requires less maintenance than a poorly planned cluster.

1.7.1 Checking for prerequisites

Once you have finished your planning working sheets, verify that your system meets the requirements that are required by HACMP; many potential errors can be eliminated if you make this extra effort.

HACMP V5.1 requires one of the following operating system components:

- AIX 5L V5.1 ML5 with RSCT V2.2.1.30 or higher.

- AIX 5L V5.2 ML2 with RSCT V2.3.1.0 or higher (recommended 2.3.1.1).
- AIX 5L V5.3 ML2 with RSCT V2. or higher (recommended 2.).
- C-SPOC vpath support requires SDD 1.3.1.3 or higher.

For the latest information about prerequisites and APARs, refer to the README file that comes with the product and the following IBM Web page:

HACMP V5.2 requires one of the following operating system components:

- AIX 5L V5.1 ML5 with RSCT V2. or higher.
- AIX 5L V5.2 ML2 with RSCT V2 or higher (recommended 2.3).
- AIX 5L V5.3 ML2 with RSCT V2. or higher (recommended 2.).
- C-SPOC vpath support requires SDD 1. or higher.

HACMP V5.3 requires one of the following operating system components:

- AIX 5L V5.3 ML2 with RSCT V2 or higher (recommended 2.3).
- C-SPOC vpath support requires SDD 1. or higher.

http://techsupport.services.ibm.com/server/cluster/

1.7.2 New installation

HACMP can be installed using the AIX Network Installation Management (NIM) program, including the Alternate Disk Migration option. You must install the HACMP filesets on each cluster node. You can install HACMP filesets either by using NIM or from a local software repository.

Installation via a NIM server

We recommend using NIM, simply because it allows you to load the HACMP software onto other nodes faster from the server than from other media. Furthermore, it is a flexible way of distributing, updating, and administrating your nodes. It allows you to install multiple nodes in parallel and provide an environment for maintaining software updates. This is very useful and a time saver in large environments; for smaller environments a local repository might sufficient.

If you choose NIM, you need to copy all the HACMP filesets onto the NIM server and define a lpp_source resource before proceeding with the installation.

Installation from CD-ROM or hard disk

If your environment has only a few nodes, or if the use of NIM is more than you need, you can use CD-ROM installation or make a local repository by copying the

HACMP filesets locally and then use the **exportfs** command; this allows other nodes to access the data using NFS.

For other installation examples, such as installations on SP systems, and for instructions on how to create an installation server, refer to Part 3, “Network Installation”, in the *AIX 5L Version 5.2 Installation Guide and Reference*, SC23-4389.

1.7.3 Installing HACMP

Before installing HACMP, make sure you read the HACMP V5.1 release notes in the /usr/es/lpp/cluster/doc directory for the latest information about requirements or known issues.

To install the HACMP software on a server node, do the following steps:

1. If you are installing directly from the installation media, such as a CD-ROM or from a local repository, enter the **smitty install_all** fast path. SMIT displays the Install and Update from ALL Available Software screen.
2. Enter the device name of the installation medium or install directory in the INPUT device/directory for software field and press Enter.
3. Enter the corresponding field values.

   To select the software to install, press F4 for a software listing, or enter `all` to install all server and client images. Select the packages you want to install according to your cluster configuration. Some of the packages may require prerequisites that are not available in you environment (for example, Tivoli Monitoring).

   The cluster.es and cluster.cspoc images (which contain the HACMP run-time executable) are required and must be installed on all servers.

   Note: If you are installing the Concurrent Resource Manager feature, you must install the cluster.es.clvm LPPs, and if you choose cluster.es and cluster.cspoc, you must also select the associated message packages.

   Make sure you select **Yes** in the Accept new license agreements field. You must choose Yes for this item to proceed with installation. If you choose No, the installation may stop with a warning that one or more filesets require the software license agreements. You accept the license agreement only once for each node.
4. Press Enter to start the installation process.

Post-installation steps

To complete the installation after the HACMP software is installed, do the following steps:

1. Verify the software installation by using the AIX command **lppchk**, and check the installed directories to see if the expected files are present.
2. Run the commands **lppchk -v** and **lppchk -c cluster***. Both commands run clean if the installation is OK; if not, use the proper problem determination techniques to fix any problems.
3. Each cluster node should be rebooted.

1.7.4 Migration paths and options

If you are in the process of upgrading or converting your HACMP cluster, the following options are available: node-by-node migration and snapshot conversion.

Node-by-node migration

The node-by-node migration path is used if you need to maintain the application available during the migration process. The steps for a node-by-node migration are:

1. Stop the cluster services on one cluster node.
2. Upgrade the HACMP software.
3. Reintegrate the node into the cluster again.

This process has also been referred to as “rolling migration”. This migration option has certain restrictions; for more details, see 1.7.6, “Node-by-node migration” on page 27.

If you can afford a maintenance window for the application, the steps for migration are:

1. Stop cluster services on all cluster nodes.
2. Upgrade the HACMP software on each node.
3. Start cluster services on one node at a time.

Snapshot migration

You can also convert the entire cluster to HACMP V5.1 by using a cluster snapshot facility. However, the cluster will be unavailable during the entire process, and all nodes *must* be upgraded before the cluster is activated again. For more details, see 1.7.5, “Converting a cluster snapshot” on page 24.

1.7.5 Converting a cluster snapshot

This migration method has been provided for cases where both AIX and HACMP must be upgraded/migrated at once (for example, AIX V4.3.3 and HACMP V4.4.1 to AIX 5L™ V5.1 and HACMP V5.1).

Important: It is very important that you do not leave your cluster in a mixed versions state for longer periods of time, since high availability cannot be guaranteed.

If you are migrating from an earlier supported version of HACMP (HAS) to HACMP V5.X, you can migrate the cluster without taking a snapshot. Save the planning worksheet and configuration files from the current configuration for future reference if you want to configure the HACMP cluster in the same way as it was configured in the previous installation. Uninstall the HACMP software components, reinstall them with the latest HACMP version, and configure them according to the saved planning and configuration files.

Note: You should be aware that after a migration or upgrade, none of the new HACMP V5.X features are active. To activate the new features (enhancements), you need to configure the options and synchronize the cluster.

To convert from a supported version of HAS to HACMP, do the following steps:

1. Make sure that the current software is committed (not in applied status).
2. Save your HAS cluster configuration in a snapshot and save any customized event scripts you want to retain.
3. Remove the HAS software on all nodes in the cluster.
4. Install the HACMP V5.1 software.
5. Verify the installed software.
6. Convert and apply the saved snapshot.

The cluster snapshot utility allows you to save the cluster configuration to a file by doing the following steps:

1. Reinstall any saved customized event scripts, if needed.
2. Reboot each node.
3. Synchronize and verify the HACMP V5.1 configuration.

The following sections explain each of these steps.

Check for previous HACMP versions

To see if HACMP Classic (HAS) software exists on your system, enter the following command:

```
# lslpp -h "cluster*"
```

If the output of the **lslpp** command reveals that HACMP is installed, but is less than V4.5, you must upgrade to V4.5 at a minimum before continuing with the snapshot conversion utility. For more information, refer to the *HACMP for AIX 5L V5.1 Administration and Troubleshooting Guide*, SC23-4862-02.

Saving your cluster configuration and customized event scripts

To save your HACMP (HAS) (V4.5 or greater) cluster configuration, create a snapshot in HACMP (HAS). If you have customized event scripts, they must also be saved.

Attention: Do *not* save your cluster configuration or customized event scripts in any of the following directory paths /usr/sbin/cluster, /usr/es/sbin/cluster, or /usr/lpp/cluster. These directories are deleted and recreated during the installation of new HACMP packages.

How to remove the HACMP classic (HAS) software

To remove the HACMP software and your cluster configuration on cluster nodes and clients, do the following steps:

1. Enter the **smitty install_remove** fast path. You should get the screen shown in Example 1-1.

Example 1-1 Remove installed software

```
                        Remove Installed Software

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* SOFTWARE name                                      [cluster*]              +
  PREVIEW only? (remove operation will NOT occur)     yes                     +
  REMOVE dependent software?                          no                      +
  EXTEND file systems if space needed?                no                      +
  DETAILED output?                                    no                      +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Installing HACMP V5.1

Follow the instructions for installing the HACMP software in 1.7.3, “Installing HACMP” on page 22.

Note: Do *not* reboot until you have converted and applied the saved snapshot.

Verify the installed software

After installing HACMP, verify that the expected files are there using **`lppchk`**. For more information, see “Post-installation steps” on page 23.

Convert and apply the saved snapshot

After you have installed HACMP V5.1 on the cluster nodes, you need to convert and apply the snapshot you saved from your previous configuration.

Important: Converting the snapshot must be performed before rebooting the cluster nodes.

To convert and apply the saved snapshot:

1. Use the **`clconvert_snapshot`** utility, specifying the HACMP (HAS) version number and snapshot file name to be converted. The -C flag converts an HACMP (HAS) snapshot to an HACMP V5.1 snapshot format:

   ```
   clconvert_snapshot -C -v version -s <filename>
   ```

2. Apply the snapshot.

Reinstall saved customized event scripts

Reinstall any customized event scripts that you saved from your previous configuration.

Note: Some pre- and post-event scripts used in previous versions may not be useful in HACMP V5.1, especially in resource groups using parallel processing.

Reboot cluster nodes

Rebooting the cluster nodes is necessary to activate the new cluster communication daemon (clcomdES).

Verify and synchronize the cluster configuration

After applying the HACMP software and rebooting each node, you must verify and synchronize the cluster topology. Verification provides errors and/or

warnings to ensure that the cluster definition is the same on all nodes. In the following section, we briefly go through the cluster verification process.

Run **smitty hacmp** and select **Extended Configuration → Extended Verification and Synchronization**, select **Verify changes only**, and press Enter (see Example 1-2 on page 27).

Example 1-2 HACMP verification and synchronization

```
   HACMP Verification and Synchronization (Active Cluster on a Local Node)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Emulate or Actual                                   [Actual]                +
  Force synchronization if verification fails?        [No]                    +
* Verify changes only?                                [No]                    +
* Logging                                             [Standard]              +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Important: You cannot synchronize the configuration in a mixed-version cluster. While upgrading, you should not leave the cluster with mixed versions of HACMP for long periods of time. New functionality supplied with V5.1 is only available when all nodes have been upgraded and the cluster has been synchronized.

1.7.6 Node-by-node migration

You must consider the following items in order to perform a node-by-node ("rolling") migration:

- All nodes in the cluster must have HACMP V4.5 installed and committed.
- Node-by-node migration functions only for HACMP (HAS) V4.5 to HACMP V5.1 migrations.
- All nodes in the cluster must be up and running the HAS V4.5 software.
- The cluster must be in a stable state.
- There must be enough disk space to hold both HAS and HACMP software during the migration process:
  - Approximately 120 MB in the /usr directory

  - Approximately 1.2 MB in the / (root) directory
- When the migration is complete, the space requirements are reduced to the normal amount necessary for HACMP V5.1 alone.
- Nodes must have enough memory to run both HACMP (HAS) and HACMP daemons simultaneously. This is a minimum of 64 MB of RAM. 128 MB of RAM is recommended.
- Check that you do not have network types unsupported in HACMP. You cannot make configuration changes once migration is started.You must remove or change unsupported types beforehand. See Chapter 3, “Planning Cluster Network Connectivity“, of the *HACMP for AIX 5L V5.1 Planning and Installation Guide*, SC23-4861-02 for details.

Important: As in any migration, once you have started the migration process, do *not* attempt to make any changes to the cluster topology or resources.

- If any nodes in the cluster are currently set to start cluster services automatically on reboot, change this setting before beginning the migration process. The following procedures describe how to turn off automatic startup for a cluster.
  - Use C-SPOC to disable automatic starting of cluster services on system restart.
  - Use the SMIT fastpath **`smitty clstop`**, and select the options shown in Example 1-3.

Example 1-3 Stop cluster services

```
                        Stop Cluster Services

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Stop now, on system restart or both                 on system restart      +
  Stop Cluster Services on these nodes                [p630n01]              +
  BROADCAST cluster shutdown?                          true                  +
* Shutdown mode                                       graceful               +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

If you do not use C-SPOC, you must change the setting on each cluster node individually.

How to perform a node-by-node migration

To perform a node-by-node migration from HACMP V4.5 to HACMP V5.1, do the following steps:

1. Save the current configuration in a snapshot (as a precautionary measure). Place it in a safe directory (one that is not touched by the installation procedures). Do *not* use /usr/sbin/cluster.
2. Stop cluster services on one of the nodes running HAS V4.5 using the graceful with takeover method. To stop cluster services from the command line, run:

   ```
   # /usr/es/sbin/cluster/utilities/clstop -gr
   ```

3. Verify that the cluster services are stopped on the node and that its cluster resources have been transferred to take over nodes before proceeding.
4. Install HACMP V5.1 on the node. For instructions, see 1.7, “HACMP software installation” on page 20.
5. Check the installed software using the AIX command **lppchk**. See “Post-installation steps” on page 23.
6. Reboot the node.
7. Restart the HACMP software:

   a. Enter the fast path **smitty hacmp**.

   b. Go to **System Management (C-SPOC)**.

   c. Select **Manage HACMP Services**.

   d. Select **Start Cluster Services**.

   When you restart Cluster Services:

   – The HACMP software is also started.

   – HACMP cluster services run on the node and the node rejoins the cluster.

   – The node reacquires the cascading resources for which it is the primary node (depending on your Inactive Takeover set).

   Both the old and new versions of HACMP (that is, HACMP V4.5 and Enhanced Scalability HACMP V5.1) are now running on the node, but only HACMP Classic (HAS) controls the cluster events and resources. If you list the daemons controlled by the system resource controller (SRC), you will see the following daemons listed on this hybrid node (see Table 1-4 on page 30).

Table 1-4 List of daemons used by HACMP

HACMP	HACMP/ES	RSCT
clstmgr	clstmgrES	grpsvcs
cllockd (optional)	cllockdES (optional)	topsvcs
clsmuxpd	clsmuxpES	emsvcs
clinfo (optional)	clinfoES (optional)	grpglsm
	clcomdES	emaixos

8. Repeat steps 2 through 6 for all the other nodes in the cluster.

   Attention: Starting the cluster services on the last node is the point of no return.

   Once you have restarted HACMP (which restarts both versions of HACMP) on the last node, and the migration has commenced, you *cannot* reverse the migration.

   If you want to return to the HACMP configuration after this point, you will have to reinstall the HACMP software and apply the saved snapshot. Up to this point, you can back out of the installation of HACMP and return to your running HACMP cluster. If you need to do this, see “Backout procedure” on page 32.

   During the installation and migration process, when you restart each node, the node is running both products, with the HACMP clstrmgr in control of handling cluster events and the clstrmgrES in passive mode.

   After you start the cluster services on the last node, the migration to HACMP proceeds automatically. Full control of the cluster transfers automatically to the HACMP V5.1 daemons.

   Messages documenting the migration process are logged to the /tmp/hacmp.out file as well as to the /tmp/cm.log and /tmp/clstrmgr.debug log files.

   When the migration is complete, and all cluster nodes are up and running HACMP V5.1, the HACMP (HAS) software is uninstalled.

9. After all nodes have been upgraded and rebooted, and the cluster is stable, synchronize and verify the configuration.

   You should also test the cluster’s proper fall-over and recovery behavior after any migration.

Note: The process of node-by-node migration from HAS 4.5 to HACMP V5.1, you will see the following warnings:

```
sysck: 3001-036 WARNING: File /etc/cluster/lunreset.lst is also owned by
fileset cluster.base.server.events.
sysck: 3001-036 WARNING: File /etc/cluster/disktype.lst is also owned by
fileset cluster.base.server.events.
```

You may safely ignore these warnings and proceed with the installation.

config_too_long message

When the migration process has completed and the HACMP filesets are being deinstalled, you may see a *config_too_long message*.

This message appears when the cluster manager detects that an event has been processing for more than the specified time. The config_too_long messages continue to be appended to the hacmp.out log until the event completes. If you observe these messages, you should periodically check that the event is indeed still running and has not failed.

You can avoid these messages by increasing the time to wait before HACMP calls the config_too_long event (use SMIT). To change the interval allocated for an event to process, do the following steps:

1. Enter the fast path **`smitty hacmp`**.
2. Go to **Extended Configuration**.
3. Select **Extended Event Configuration**.
4. Select **Change/Show Time Until Warning**.

You must do this on every node. It takes effect after restarting cluster services.

How the node-by-node migration process works

When you have installed HACMP on all cluster nodes (all nodes are now in a hybrid state), starting Cluster Services on the last cluster node automatically triggers the transfer of control to HACMP V5.1 as follows:

1. Installing HACMP V5.1 installs a recovery file called firstboot in a holding directory on the cluster node, and creates a migration file (.mig) to be used as a flag during the migration process.
2. The HACMP recovery driver sends a message to the HACMP Cluster Manager telling it to run the waiting and waiting_complete events.
   - HACMP uses the RSCT Group Services to verify cluster stability and membership.

- The firstboot file on each cluster node is moved to an active directory (/etc).
- The migration flag (.mig file) created during installation is transferred from the HACMP V5.1 directory to the HACMP V4.5 directory on all nodes.

When the firstboot file is moved to the active directory and the.mig file transfer is complete on all nodes, transfer of control to HACMP continues with the HACMP migrate event.

3. The HACMP recovery driver issues the migrate event.
   - HACMP V5.1 stops the HACMP V4.5 daemons using the forced option.
   - The HACMP V5.1 clinfoES and clsmuxpdES daemons are all activated, reusing the ports previously used by the HACMP V4.5 versions of those daemons.
4. HACMP V5.1 recovery driver runs the migrate_complete event.
   - HACMP V4.5 is deinstalled. Configuration files common to both products are left untouched.
   - Base directories are relinked.
   - The /etc/firstboot files are removed.
   - The migration flag (.mig file) in the HACMP /usr/sbin/cluster directory is removed.
5. Migration is now complete.

Cluster snapshots saved during migration

Pre-existing HACMP snapshots are saved in the /usr/es/sbin/cluster/snapshots directory.

Handling node failure during the migration process

If a node fails during the migration process after its firstboot file moved to an active directory, it completes the migration process during node reboot. However, the failed node may have an HACMP ODM that is not in sync when it reintegrates into the cluster. In this case, synchronize the topology and resources of the cluster before reintegrating the failed node into the cluster. To synchronize the cluster.

Backout procedure

If for some reason you decide not to complete the migration process, you can uninstall the HACMP V5.1 software on the nodes where you have installed it at any point in the process before starting HACMP on the last node.

Note: Deinstall the HACMP software only on the local node. During a migration, do not select the option to deinstall the software from multiple nodes.

To deinstall the HACMP software:

1. On each node, one by one, stop cluster services:

   To stop cluster services, see Example 1-3 on page 28.

   Check that the cluster services are stopped on the node and that its cluster resources have been transferred to takeover nodes before proceeding.
2. When you are sure the resources on the node have been properly transferred to a takeover node, remove the HACMP V5.1 software. See "How to remove the HACMP classic (HAS) software" on page 25.
3. Start HACMP on this node. When you are certain the resources have transferred properly (if necessary) back to this node, repeat these steps on the next node.
4. Continue this process until HACMP has been removed from all nodes in the cluster.

Handling synchronization failures during node-by-node migration

If you try to make a change to the cluster topology or resources when migration is incomplete, the synchronization process will fail. You will receive the following message:

```
cldare: Migration from HACMP V4.5 to HACMP V5.1 Detected. cldare cannot be run
until migration has completed.
```

To back out from the change, you must restore the active ODM. Do the following steps:

1. Enter **smitty hacmp**.
2. Go to **Problem Determination Tools**.
3. Select **Restore HACMP Configuration Database from Active Configuration**.

1.7.7 Upgrade options

Here we discuss upgrades to HACMP.

Supported upgrades to HACMP V5.1

HACMP conversion utilities provide an easy upgrade path from the versions listed here to V5.1:

- HACMP/ES V4.4.1 to HACMP V5.1
- HACMP/ES V4.5 to HACMP V5.1

If you want to convert to HACMP V5.1 from versions earlier than those listed here, you must first upgrade to one of the supported versions. You will then be able to convert to HACMP V5.1. For example, to convert from HACMP/ES 4.2.2 to HACMP V5.1, you must first perform an installation upgrade to HACMP/ES 4.4.1 or higher and then upgrade to HACMP V5.1.

To upgrade to HACMP V5.1, do the following steps:

1. Upgrade to AIX 5L V5.1 Maintenance Level 5 or higher if needed.
2. Check and verify the AIX installation, if needed.
3. Commit your current HACMP software on all nodes.
4. Stop HACMP/ES on one node (gracefully with takeover) using the **clstop** command.
5. After the resources have moved successfully from the stopped node to a takeover node, install the new HACMP software.

   For instructions on installing the HACMP V5.1 software, see 1.7, “HACMP software installation” on page 20.

   Verify the software installation by using the AIX command **lppchk**, and check the installed directories to see that expected files are present:

   ```
   lppchk -v or lppchk -c “cluster.*”
   ```

   Both commands should run clean if the installation is OK.
6. Reboot the first node.
7. Start the HACMP software on the first node using **smitty clstart** and verify that the first node successfully joins the cluster.
8. Repeat the preceding steps on remaining cluster nodes, one at a time.
9. Check that the tty device is configured as a serial network.
10. Check that all external disks are available on the first node (use **lspv** to check the PVIDs for each disk). If PVIDs are not displayed for the disks, you may need to remove the disk and reconfigure them.
11. After all nodes have been upgraded, synchronize the node configuration and the cluster topology from Node A to all nodes, as described in “cl_convert and clconvert_snapshot” on page 35. Do not skip verification during synchronization.

Important: When upgrading, never synchronize the cluster definition from an upgraded node, when a node that has not been upgraded remains in a mixed-version cluster. The **cl_convert** utility assigns node IDs that are consistent across all nodes in the cluster. These new IDs may conflict with the already existing ones.

12. Restore the HACMP event ODM object class to save any pre- and post-events you have configured for your cluster.
13. Make additional changes to the cluster if needed.
14. Complete a test phase on the cluster before putting it into production.

cl_convert and clconvert_snapshot

The HACMP conversion utilities are **cl_convert** and **clconvert_snapshot**.

Upgrading HACMP/ES software to the newest version of HACMP involves converting the ODM from a previous release to that of the current release. When you install HACMP, **cl_convert** is run automatically. However, if installation fails, you must run **cl_convert** from the command line.

In a failed conversion, run **cl_convert** using the -F flag. For example, to convert from HACMP/ES V4.5 to HACMP V5.1, use the -F and -v (version) flags as follows (note the "0" added for V4.5):

```
# /usr/es/sbin/cluster/conversion/cl_convert -F -v 4.5.0
```

To run a conversion utility requires:

- Root user privileges
- The HACMP version from which you are converting

The **cl_convert** utility logs the conversion progress to the /tmp/clconvert.log file so that you can gauge conversion success. This log file is generated (overwritten) each time **cl_convert** or **clconvert_snapshot** is executed.

The **clconvert_snapshot** utility is not run automatically during installation, and must be run from the command line. Run **clconvert_snapshot** to upgrade cluster snapshots when migrating from HACMP (HAS) to HACMP, as described in "cl_convert and clconvert_snapshot" on page 35.

Upgrading the concurrent resource manager

To install the concurrent access feature on cluster nodes, install the Concurrent Resource Manager (CRM) using the procedure outlined in 1.7, "HACMP software installation" on page 20.

AIX 5L V5.1 supports enhanced concurrent mode (ECM). If you are installing HACMP with the Concurrent Resource Manager feature, see Chapter 2, “Initial Cluster Planning”, in the *HACMP for AIX 5L V5.1 Planning and Installation Guide*, SC23-4861-02.

See Chapter 5, “Planning Shared LVM Components“, in the *HACMP for AIX 5L V5.1 Planning and Installation Guide*, SC23-4861-02 for information about enhanced concurrent mode and on supported IBM shared disk devices. In addition, if you want to use disks from other manufacturers, see Appendix D, “OEM Disk Accommodation”, in the *HACMP for AIX 5L V5.1 Planning and Installation Guide*, SC23-4861-02.

Problems during the installation

If you experience problems during the installation, the installation program automatically performs a cleanup process. If, for some reason, the cleanup is not performed after an unsuccessful installation, do the following steps:

1. Enter `smitty install`.
2. Select **Software Maintenance and Utilities**.
3. Select **Clean Up After a Interrupted Installation**.
4. Review the SMIT output (or examine the /smit.log file) for the interruption’s cause.
5. Fix any problems by using AIX problem determination techniques and repeat the installation process.

2

High availability components

In this chapter we discuss the following topics:

- HACMP configuration data
- Software components
- Cluster topology
- Resources and resource groups
- HACMP plugins
- Features (HACMP 5.1, 5.2 and 5.3)
- Limitations
- Storage considerations
- Shared storage configuration

2.1 HACMP configuration data

There are two main components to the cluster configuration:

- Cluster topology - describes the underlying framework - the nodes, the networks and the storage. HACMP uses this framework to keep the other main component - the resources highly available.
- Cluster resources - are those components that HACMP can move from node to node, for example service IP labels, file systems and applications.

When the cluster is configured, The cluster topology and resource information is entered on one node, A verification process is then run, and the data synchronized out to the other nodes defined in the cluster. HACMP keeps this data in it's own Object Data Manager (ODM) classes on each node in the cluster.

While HACMP can be configured / modified from any node in the cluster, it is good practice to perform administrative operations from one node to ensure that HACMP definitions are kept consistent across the cluster, thus preventing a cluster configuration update from multiple nodes which may result in inconsistent data.

We recommend the following basic steps for configuring your cluster:

- Define the cluster and the nodes
- Discover the additional information (networks, disks)
- Define the topology
- Verify and synchronize the topology then start the cluster services
- Define the resources and resource groups
- Verify and synchronize the resources

AIX configuration

You should be aware that HACMP makes some changes to the system when it is installed and/or started:

Installation changes

- Files modified:
  - /etc/inittab
  - /etc/rc.net
  - /etc/services
  - /etc/snmpd.conf
  - /etc/snmpd.peers
  - /etc/syslog.conf

- /etc/trcfmt
- /var/spool/cron/crontabs/root

- Adding the *hacmp* group.
- Also, using cluster configuration and verification the /etc/hosts may also be modified by adding or modifying entries.
- The following network options values are changed:
  - **routerevalidate**: Set to “1” - Each connection’s cached route should be revalidated each time a new route is added to the routing table. This will ensure that applications that keep the same connection open for a long time, will use the correct route after a change to the routing table.
  - **nonlocsrcroute**: Set to “1” - Allows source routed packets to be addressed to hosts outside the local network.
  - **ipsrcrouterecv**: Set to “1” - Allows the system to accept source routed packets

Tuning operating system parameters

In the past tuning AIX for HACMP was encouraged, however we adopt the philosophy that the system should be tune for the application, not for HACMP. For example if the system hangs for a period and HACMP reacts, the system should be tuned so the application is unlikely to hang. While HACMP can be tune to be less sensitive, there is no general AIX tuning rules for HACMP.

2.2 Software components

The following layered model describes the software components of an HACMP cluster:

- **Application layer**: Any application that is made highly available through the services provided by HACMP
- **HACMP layer**: Software that responds to changes within the cluster to ensure that the controlled applications remain highly available
- **RSCT layer**: The daemons that monitor node membership, communication interface and device health and advises HACMP accordingly
- **AIX layer**: Provides support for HACMP through the LVM which manages the storage and TCP/IP layer which provides communication.
- **LVM layer**: Provides access to storage and status information back to HACMP
- **TCP/IP layer**: Provides reliable communication, both node to node and node to client

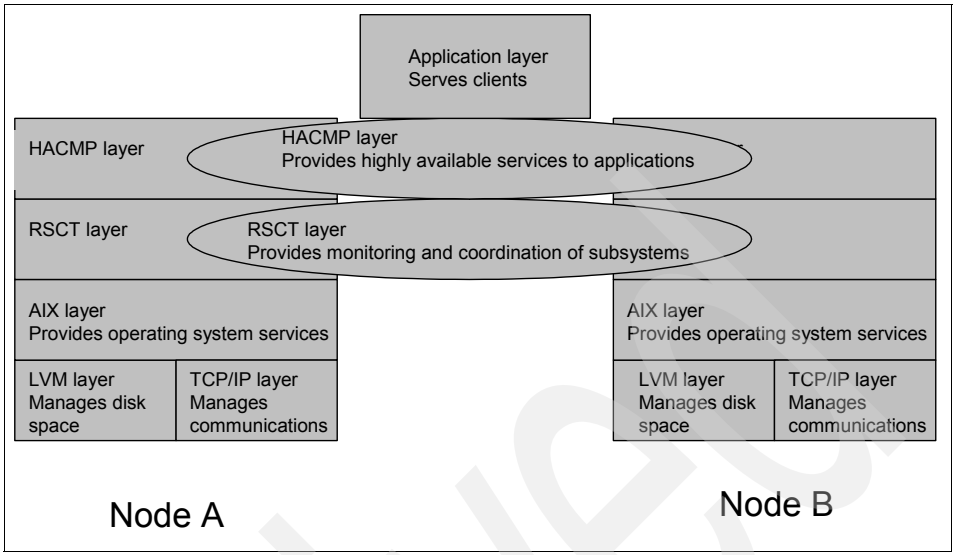

Figure 2-1 Software model of an HACMP cluster

- The application layer may consist of:
  - Application code (programs, daemons, kernel extensions etc.)
  - Application configuration data (files or binaries)
  - Application (customer) data (files or raw devices)
- HACMP layer consists of:
  - HACMP code (binaries - daemons and executable commands, libraries, scripts)
  - HACMP configuration (ODM, ASCII files)
  - HACMP log files
  - Services:
    - Cluster communication daemon (clcomdES)
    - Cluster manager (clstrmgrES)
    - Cluster information daemon (clinfoES)
    - etc.
- RSCT layer consists of:
  - RSCT code (binaries - daemons and commands, libraries, scripts)
  - Configuration files (binary registry and ASCII files)
  - Services:
    - Topology and group (topsvcs and grpsvcs)

- Resource monitoring and control (RMC)

- AIX layer consists of
  - Kernel, daemons and libraries
  - Device drivers
  - Networking and TCP/IP layer
  - Logical volume manager (LVM)
  - Configuration files (ODM, ASCII)

2.3 Cluster topology

The cluster topology represents the physical view of the cluster and how hardware cluster components are connected via networks (IP and non-IP). To understand the operation of HACMP, you need to understand the underlying topology of the cluster - the role each component plays and how HACMP interacts. In this section we describe:

- HACMP cluster
- Nodes
- Sites
- Networks
- Communication interfaces / devices
- Persistent node IP labels / addresses
- Network modules (NIMs)
- Topology and group services
- Clients

Figure 2-2 on page 42 shows typical cluster topology with:

- Three nodes
- Two IP networks (HACMP logical networks) with redundant interfaces on each node.
- Shared storage
- Point-to-point non-IP connections (serial) between nodes configured as independent physical networks, but connecting nodes in a ring configuration.

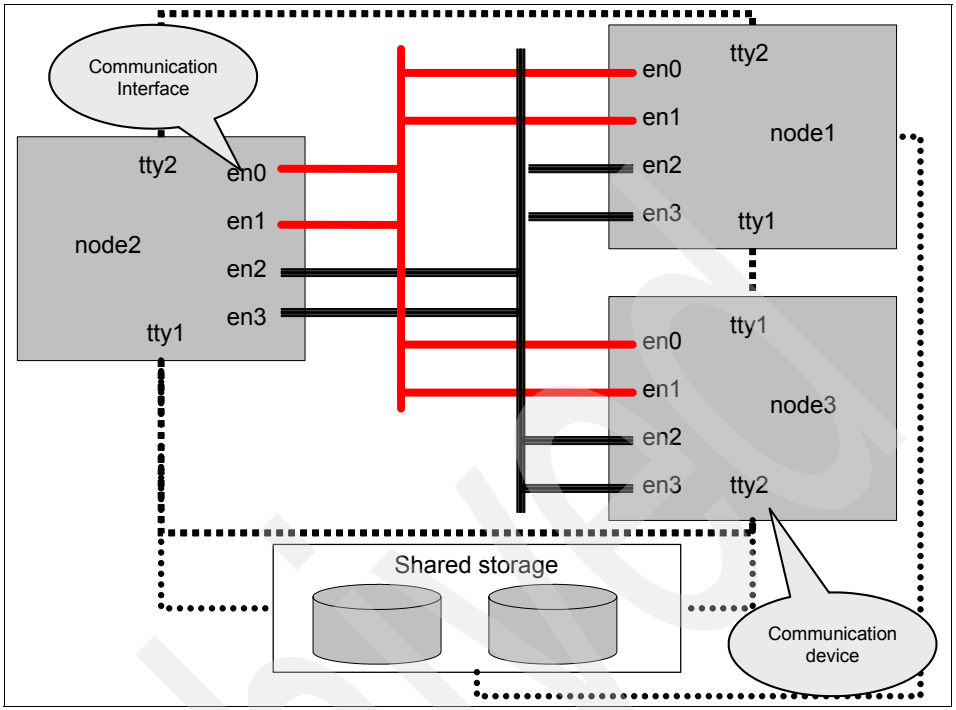

Figure 2-2 Example of cluster topology

HACMP cluster

A name (up to 32 characters, [a-z],[A-Z],[0-9] or "_", starting with an alpha) is assigned to the cluster. A cluster ID (number) is also associated with the cluster. HACMP 4.5 and above automatically generates a unique ID for the cluster. All heartbeat packets contain this ID, so two clusters on the same network should never have the same ID.

Cluster nodes

Nodes form the core of an HACMP cluster. A node is a server running an image of the AIX operating system (standalone or a partition), HACMP code and application software. The maximum number of nodes supported in a HACMP cluster is 32.

When defining the cluster node, a unique name must assigned and a communication path to that node must be supplied (IP address or a resolvable IP label associated with one of the interfaces on that node). Starting HACMP 5.1, the node name can be the hostname (short), a fully qualified name (hostname.domain.name) or any name up to 32 characters ([a-z],[A-Z],[0-9]or"_" and must start with an alpha).

The communication path is first used by HACMP to confirm that the node can be reached, then used to populate the ODM on each node in the cluster once secure communications have been established between the nodes. However, once the cluster topology has been configured, HACMP can use any interface to attempt to communicate between nodes in the cluster.

HACMP no long requires the hostname to be an resolvable IP label - i.e., an address on one of the IP interfaces. For consistency, we recommend to use the hostname which also resolves to the persistent IP address associated with each node - however this is not mandatory.

Attention: At time of publication, when configuring HACMP with CUoD or DLPAR, the LPAR names (as defined on HMC) must match the HACMP node names and the AIX hostnames.

Sites

The use of sites is optional. They are designed for use in cross-site mirroring and/or HACMP/XD configurations. A site consists of one or more nodes grouped together at a given location. HACMP supports a cluster divided into two sites. Site relationships also may exist as part of a resource group's definition, but should be set to ignore if sites are not defined/used.

It is possible to use sites outside HACMP/XD and cross site mirroring, but appropriate methods or customization must be provided to handle site operations. If sites are defined, site events are run during node_up and node_down events.

Sites also have two characteristics that need to be defined:

- **Dominance**: Which site is the dominant site.
- **Site backup communications**: Can be none, *dbfs* (dial back fail safe) or *sgn* (for geo_secondary network).

The backup communication methods are used in the case of main IP communication network between the two sites fails, in order to avoid site "split brain" situations. The non dominant site will attempt to contact the dominant site using the site backup communications network, and if the dominant site is still operational, it will halt.

Networks

In HACMP, the term "network" is used to define a logical entity that groups the communication interfaces and devices used for communication between the nodes in the cluster, and for client access. The networks in HACMP can be defined as IP networks and non-IP networks.

The following terms are used to describe HACMP networking:

- IP address: the doted decimal IP address
- IP label: the label that is associated with a particular IP address as defined by the name resolution method (DNS or static - i.e., /etc/hosts).
- Base IP label / address: The default IP label / address that is set on the interface by AIX on startup. The base address of the interface.
- Service IP label / address: An IP label / address over which a service is provided and it may be bound to a single node or shared by multiple nodes. Although not part of the topology, these are the addresses that HACMP will keep highly available.
- Boot interface: earlier versions of HACMP have used the terms “boot adapter” and “standby adapter” depending on the function. These have been collapsed into the one term to describe any IP network interface that can be used by HACMP to host a service IP label / address.
- IP aliases: An IP alias is an IP address that is added to an interface, rather than replacing it’s base IP address. This is an AIX function that is supported by HACMP, although HACMP still requires that there be only one subnet mask used for all the addresses associated with the adapter.
- Logical network interface: the name to which AIX resolves a port (for example, en0) of a physical network adapter.

It is good practise to have all the above IP addresses defined in /etc/hosts file and this file the same on all nodes in the cluster. There is certainly no requirement to use fully qualified names. While HACMP is processing network changes, the NSORDER variable is set to local (i.e., pointing to /etc/hosts), however it is also good practice to set this in /etc/netsvc.conf.

HACMP communication interfaces

A “communication interface”, or just “interface”, refers to the physical adapter that supports the TCP/IP protocol. and is represented by an IP address. The network interfaces that are connected to a common physical network are combined into logical networks that are used by HACMP.

Each interface is capable of hosting several TCP/IP addresses. When configuring a cluster, you define the IP addresses that HACMP will monitor via RSCT (base or boot IP addresses) and the IP addresses that HACMP itself will keep highly available (the service IP addresses and persistent aliases).

HACMP communication devices

HACMP topology also includes point-to-point non-IP networks such as serial RS232, Target mode SCSI, target mode SSA, and disk heartbeat connections.

Both ends of a point-to-point network are AIX devices (as defined in /dev directory), such as /dev/tty1, /dev/tmssa1, /dev/tmscsi1, and /dev/hdisk1.

For example, a heartbeat over disk uses the disk device name (for example, /dev/hdisk2) as the device configured to HACMP at each end of the connection.

These non-IP networks are point-to-point connections between two cluster nodes, and are used by RSCT for control messages and heartbeat traffic. These networks provide an additional protection level for the HACMP cluster, in case the IP networks or the TCP/IP subsystem on the nodes fails.

Communication adapters and links

Is an X.25 adapter used to provide a highly available communication link and the following can be defined as resources in HACMP:

- SNA configured over LAN network adapters
- SNA configured over X.25 adapter
- Native X.25 links

HACMP managed these links as part of resource groups, thus ensuring high availability communication links. In the event of a physical network interface failure, an X.25 link failure, or a node failure, the highly available communication link is migrated over to another available adapter on the same node, or on the takeover node (together with all the resources in the same resource group).

Physical and logical networks

A *physical* network connects two or more physical network interfaces. There are many types of physical networks, and HACMP broadly categorizes them as IP-based and non-IP networks:

- TCP/IP-based, such as Ethernet, or Token Ring
- Device-based, such as RS-232, target mode SCSI (tmscsi), target mode SSA (tmssa), or disk heartbeat.

HACMP, like AIX has the concept of logical networks. Two or more network interfaces on one physical network can be grouped together to form a logical network. These logical networks are known by a unique name (for example net_ether_01 if assigned by HACMP) and may consist of one or more subnets. A logical network can be viewed as the group of interfaces used by HACMP to host one or more service IP labels / addresses. RSCT forms it's own networks connecting interfaces on the same subnet, and if needed can provide temporary routing between the subnets.

Networks definitions can be added using the HACMP smit screens, however we recommend that you use the discovery process before starting to configure your

networks. Running the discovery process will populate pull down lists that can be used in the configuration process. The discovery process will harvest information from the **/etc/hosts** file, defined interfaces, defined adapters, target mode devices and existing enhanced concurrent mode disks and create the following files:

- clip_config: Contains details of the discovered interfaces, used in the <f4> smit lists
- clvg_config: Contains the details of each physical volume (PVID, VG name, status, major number etc.) and a list of free major numbers.

Running discovery may also reveal any inconsistency in the network at your site.

Global network

A g*lobal network* is a collection of multiple HACMP networks of the same type, for example Ethernet. As discussed above, the HACMP logical networks may be composed of any combination of physically different networks, and / or different subnets. It is important for HACMP to know where a single network on a node has failed - or if there is a global network failure - as in a global failure, there is nothing to be gained by moving a resource group to another node.

2.3.1 RSCT and HACMP heartbeating

The HACMP cluster manager uses a variety of sources to get information about possible failures:

- RSCT monitors the state of the network interfaces and devices
- AIX LVM monitors the state of the disks, logical volumes and volume groups
- HACMP Application monitors monitor the state of the applications

HACMP, like many other types clusters, uses heartbeat (“keep alive” - KA) packets to monitor the availability of network interfaces, communication devices and IP labels (service, non-service, and persistent). HACMP can use both IP and non-IP networks to exchange heartbeat packets or messages between the nodes. Through heartbeating, HACMP maintains information about the status of the interfaces, devices and adapters, and through them, the availability of the cluster nodes.

Starting with HACMP V5.1, heartbeating is exclusively based on RSCT topology services. Prior to this, HACMP Classic (up to HACMP V4.5) used it’s own code for Network Interface Modules (NIMs). The RSCT daemons use UDP for the heartbeat packets between nodes. When HACMP is started on a node, HACMP passes the network topology stored in the HACMP ODM configuration to RSCT.

RSCT uses this information to construct it's communication groups ("heartbeat rings") and in turn provides failure notifications back to HACMP.

RSCT was originally developed in the early 1990's for the IBM SP systems and then became the infrastructure for HACMP/ES (enhanced scalability).

RSCT consists of the following components:

- **Resource monitoring and control (RMC)**: HACMP 5.1 and earlier used the event management subsystem. RMC is a distributed subsystem that provides a set of high availability services. It creates events by matching the state of a systems resources with information about the resource conditions of interest to the clients. Clients can then use event notifications to trigger recovery actions.
  However the event manager still exists to support Oracle RAC.
- **Resource managers**: These are daemons that are actually a part of RMC that represent an actual administrative task or system function. HACMP uses RMC for dynamic node priority, Application monitoring and user defined events. For example the resource monitor that represents the percentage that the CPU is idle, will be used if a node is to fallover to the node with the highest CPU idle.
- **Group services**: Provides a system wide and highly available facility for monitoring and coordinating changes in state of an application running on a set of nodes.
- **Topology services**: Handles the heatbeating over the multiple networks in a cluster. Has knowledge of the network configuration and provides information about the state of the network interfaces and adapters as well as the nodes themselves.

Figure 2-3 on page 48 shows the some of the RSCT daemons and how they interact with other HACMP daemons (in HACMP V5.3).

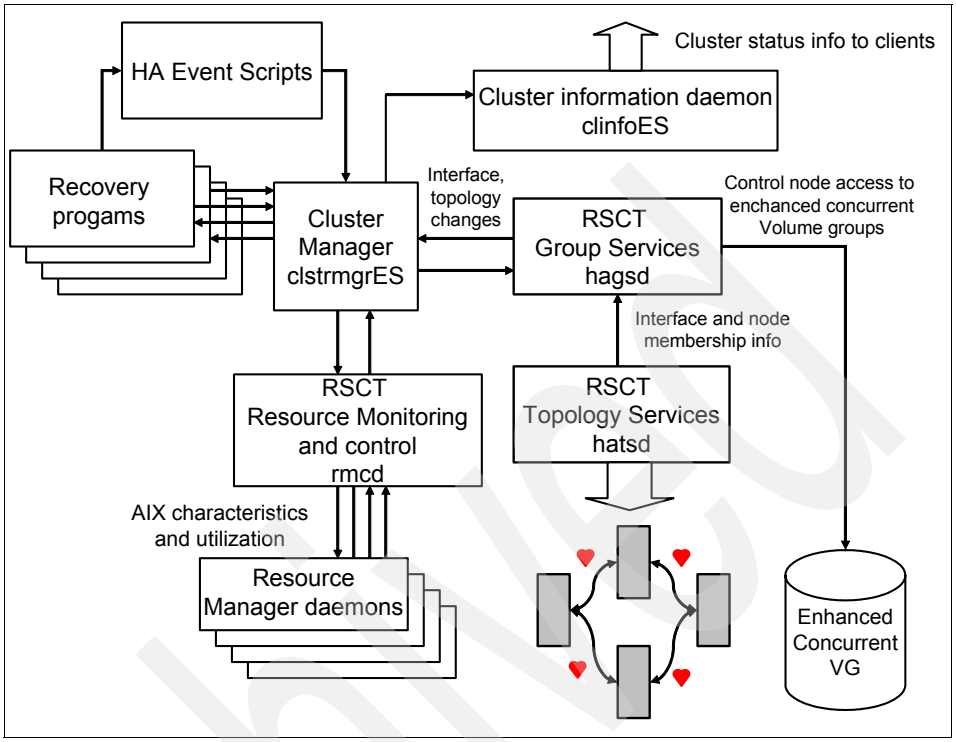

Figure 2-3 RSCT and important cluster daemons

This heartbeating is performed by exchanging messages (heartbeat packets) between the nodes over each communication interface and device defined to the cluster (topology). Each node sends a heartbeat packet and expects to receive a packet over each network within the interval determined by the network sensitivity. As each host only communicates with it's 2 neighbors on each network ring, a host will only receive one packet from a particular node every two heartbeat intervals. This is important in calculating how long it takes HACMP to determine a failure.

RSCT only monitors the base addresses on the interfaces (unless heartbeat over IP alias is selected), so therefore does not monitor the service IP labels (if using IPAT via alias) or the persistent IP labels.

HACMP is responsible for keeping track of the aliased labels (service IP if IPAT via aliasing being used, and the persistent alias labels) - both through the state of the underlying interface, the link status and by monitoring the received packet count (similar to the **`netstat`** command output). HACMP V5.2 and later will attempt to bring up an interface if it finds it in “down” or “detached” state (as reported by the **`lsattr`** command) but with the physical link still active.

Note: This if you use **`ifconfig`** command to bring an adapter down for testing, HACMP will bring it up, without involving any HACMP event processing.

RSCT determines that one of interfaces or adapters on a node has failed if it is no longer receiving heartbeat packets from it, but still receiving heartbeat information through other interfaces and adapters on that node. In this case HACMP preserves the communication to the node by transferring the service (and persistent) IP labels to another network interface on the same network on the same node.

If all interfaces on that HACMP network are unavailable on that node then HACMP transfers all resource groups containing IP labels on that network to another node with available interfaces. If RSCT fails to receive heartbeat packets through any of the interfaces or adapters on a node, then that node is considered to have failed, and HACMP will try to bring the affected resource groups online on another node.

RSCT communications

HACMP is responsible for starting up RSCT (topology and group services) on nodes joining the cluster. RSCT organizes it's networks and the inter-node communications, depending on network type as follows:

- **IP networks**: A ring (RSCT communication group) is formed for each logical subnet for the interfaces defined to HACMP, in IP address order. Each node will communicate with it's two neighbors - the nodes with the next higher and next lower IP address. Each IP ring (a.k.a RSCT communication group) is modified as each incoming node joins the cluster.
- **Serial networks**: RSCT also creates a communication group for each pair of communication devices or HACMP serial network. RSCT then builds a logic network from these communication groups to pass information between nodes using the non-IP communications.

Considering a cluster topology as seen in Figure 2-2 on page 42, RSCT builds up 3 heartbeat networks - one for each IP subnet, and one for the "non-IP device ring" as in Figure 2-4 on page 50.

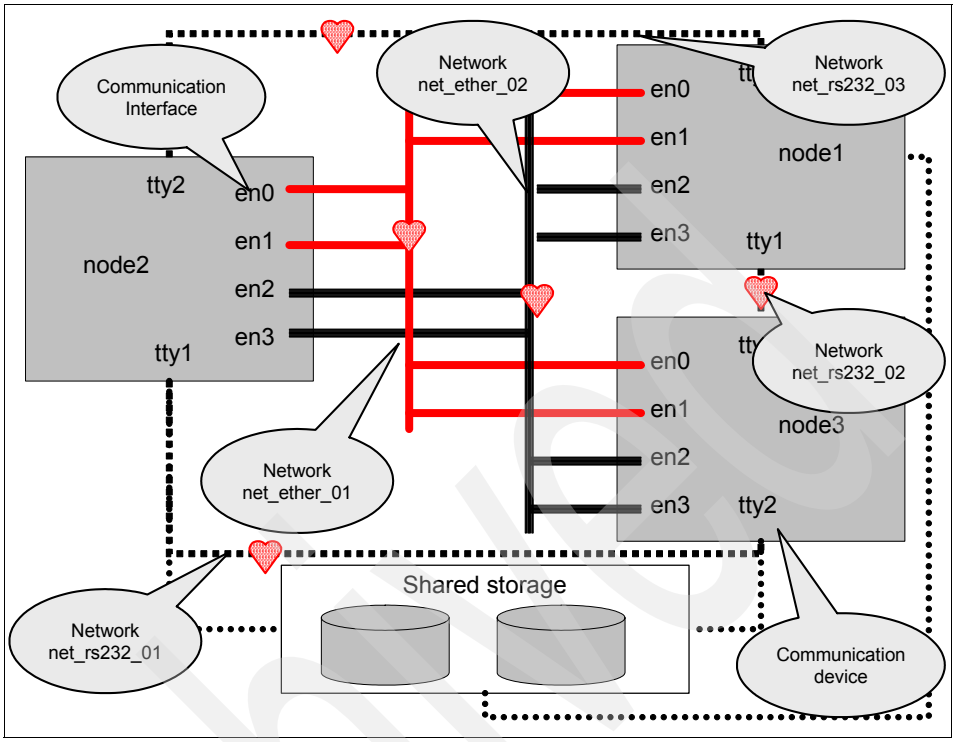

Figure 2-4 HACMP networks superimposed on the topology

Earlier versions of HACMP were limited to two serial (non-IP) communication devices per type per node, so only the ring configuration was possible for three or more nodes. Newer HACMP (5.1 and later) support a configuration with non-IP networks connecting every node to every other node, if there are sufficient devices available on each node.

RSCT uses particular nodes to manage the communications around these communication groups. The nodes that fulfill these tasks are chosen dynamically, and can change each time a node enters or leaves the cluster.

- **Group leader**: The node with the highest IP address in the first communication group created is called the group leader. This node keeps the information about the other nodes in the cluster and the network configuration.
- **Group leader backup**: The node with the second highest IP address in the first communication group is called the crown prince. This node keeps a backup copy of the group leaders topology data, and will take over as the group leader should the group leader leave the cluster.

- **Mayor**: The node with the third highest IP address in the first communication group (or the GL backup if third node is not available). This node is responsible for ensuring that all nodes in the cluster are informed of any changes in cluster topology.

Figure 2-5 shows an example of two communication groups formed for the two IP subnets and the RSCT roles of the nodes.

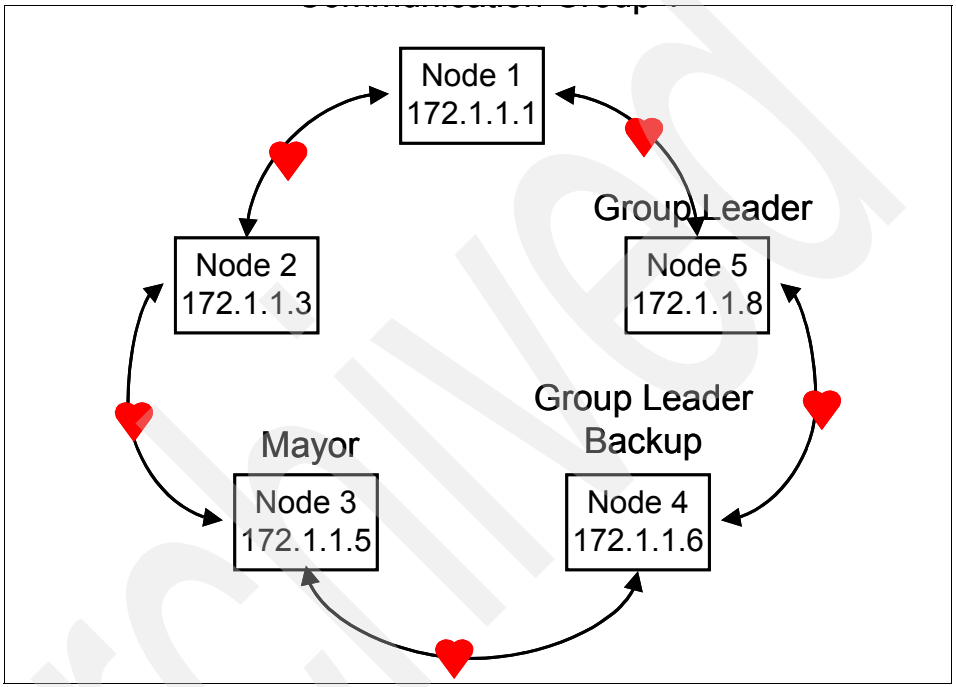

Figure 2-5 Example showing RSCT communication groups and node roles

As discussed, HACMP relies heavily on RSCT and the data received in the heartbeat packets for information about the state of the topology of the cluster, however HACMP must be really sure that a node has actually failed before it takes any action. If there is no redundancy in the network, then HACMP could easily make an incorrect assumption about the state of then nodes. For example if RSCT was relying only on TCP/IP network - a failure of a network component (switches, routes, hubs) or the TCP/IP subsystem would be incorrectly interpreted as a failure of one or more nodes. Figure 2-6 on page 52 shows an example of a cluster relying totally on TCP/IP for heartbeating.

In this example, nodes 1 and 2 will assume that nodes 3 and 4 have failed and proceed to bring their resources on line. Similarly nodes 3 and 4 will assume that nodes 1 and 2 have failed. This is called a “partitioned cluster” can lead to data

corruption as nodes on either side of the split attempt to simultaneously access data and start applications.

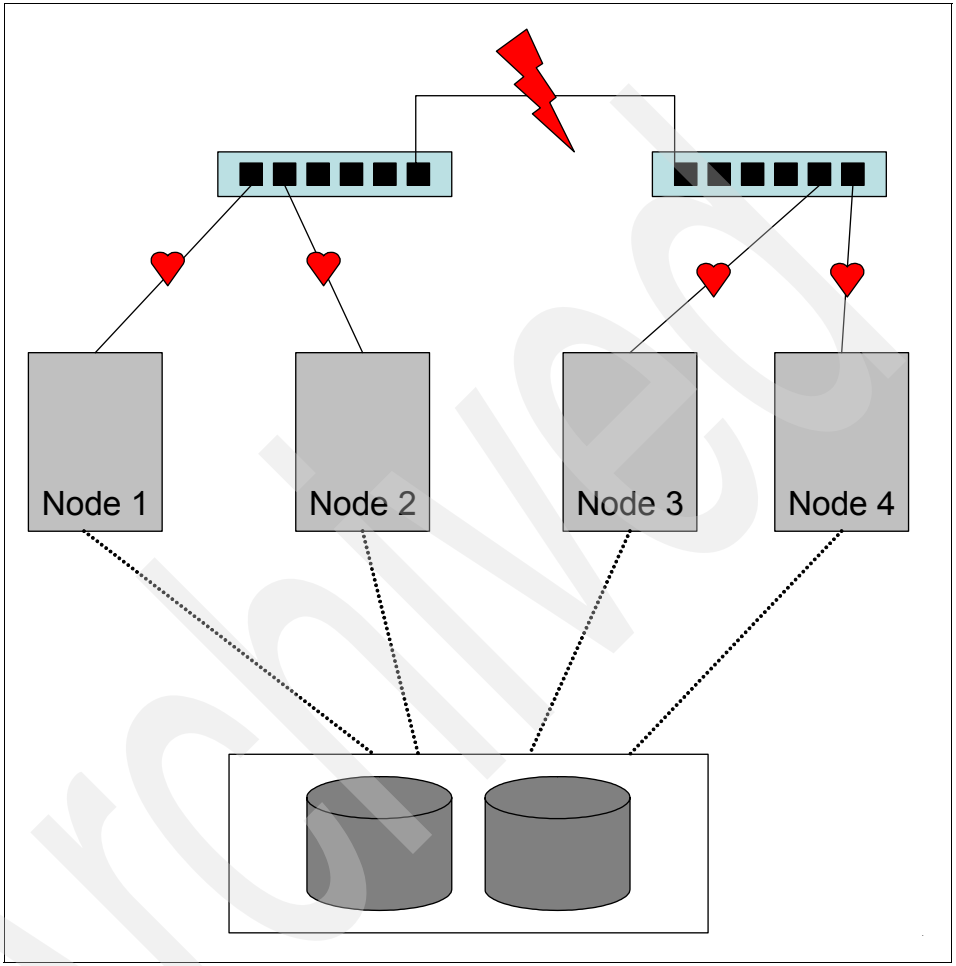

Figure 2-6 Split cluster caused by failure of network component

To help HACMP distinguish between a real node failure and a failure of the TCP/IP subsystem, another communication path between the nodes is required. A path that doesn't rely on TCP/IP. HACMP uses non-IP (point-to-point or device based) serial networks for this. As RSCT monitors both the device based and TCP/IP networks, HACMP can then use this information to distinguish between a node failure and a IP network / subsystem failure. It is recommended that each cluster have at least one non-IP network defined for each of the nodes in the cluster to prevent cluster partitioning. If serial networks had been used in the example shown in Figure 2-6, HACMP would have recognized the failure

correctly and there would have been no risk of data loss or corruption. See Figure 2-7.

For a recommended two-node cluster configuration, see Figure 2-7.

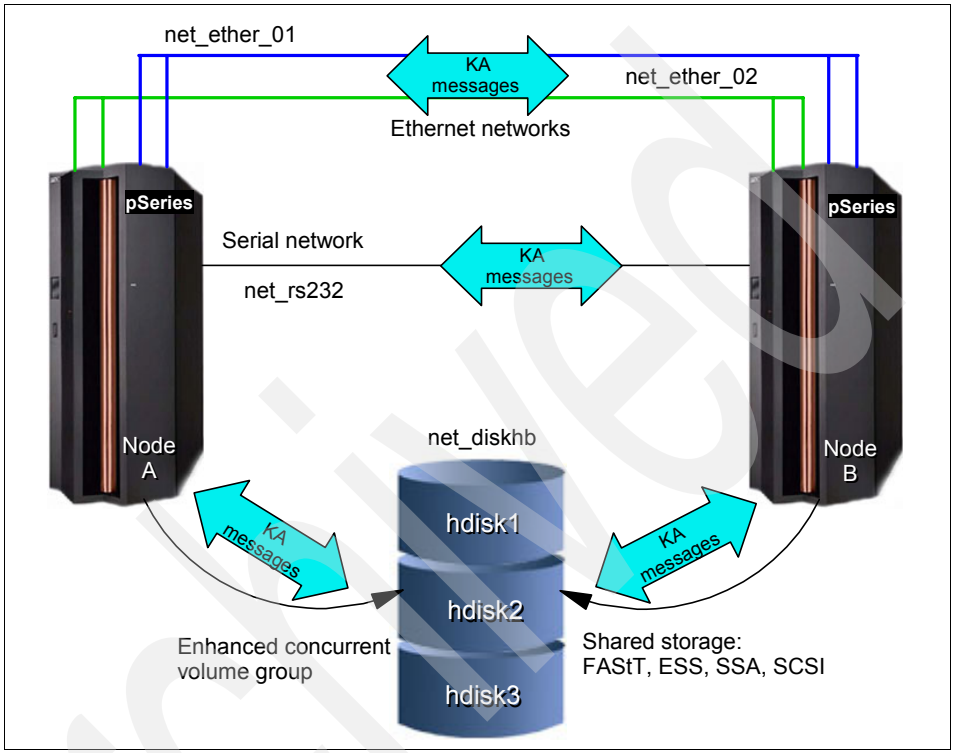

Figure 2-7 Heartbeating in an HACMP cluster

Another scenario where RSCT and HACMP will have trouble accurately determining the state of an interface is the single interface network (singleton interface). If, due to some failure, an interface finds that it is the sole interface on a network, then there is no other address for it to swap heartbeat packets with.

In the situation of a single interface network, RSCT will determine if the interface is alive by monitoring the receive packet count while:

1. Issuing a broadcast ping
2. Sending a ICMP ECHO packet (ping) to all the addresses in the netmon.cf file
3. Creating a temporary route from one of the other communication groups and testing communications

The /usr/es/sbin/cluster/etc/netmon.cf file should:

- Contain a list of IP addresses or resolvable labels, one per line.
- Be available (and the same) on each node as they join the cluster.

Subnetting and RSCT

AIX 5L supports multiple routes to the same destination in the kernel routing table. This implies that if multiple matching routes have the same criteria, routing can be performed alternatively using each of the subnet routes. This is also known as route striping.

Thus the effect of multiple interfaces on the same subnet on one node is that packets will be sent out each of the interfaces alternately. This means that other nodes and therefore RSCT, will not be able to determine which interface the heartbeat packet came from. To avoid this situation where RSCT, and therefore HACMP will not be certain of the state of the interfaces, there are strict rules regarding the subnet configuration. These rules depend on the network configuration and are discussed in the IP address takeover sections.

Note: There is a new no option in AIX 5.3 called *mpr_policy*, which allows the configuration of TCP/IP to ensure that packets for a particular destination will only come from one adapter. To configure TCP/IP to use an adapter based on the destination of the packet, use *mpr_policy = 5*. We recommend that this is set if any applications are sensitive to the adapter from which the packet comes - e.g., for NFS.

2.3.2 Heartbeat over IP aliases

HACMP now supports the use of heartbeating over IP aliases. This configuration removes the subnet restrictions on the base interfaces that are discussed in the above section. Now it is possible to configure the base IP addresses without any subnet restrictions and HACMP and RSCT will then configure and use a range of separate subnets for heartbeat communications.

These subnets do not need to be routable and allow the base IP addresses to be configured according the requirements of the site, rather than according to HACMP requirements. For example, if your network administrator requires that the base IP addresses for each adapter are in the same subnet. Without heartbeat over IP aliases, HACMP would not support this configuration as RSCT would not be able to monitor the state of each adapter.

However, it is still recommend that the service IP address still be on a different subnet to the base IP addresses on the interfaces, so that HACMP can still

accurately monitor the Service IP addresses (unless you can take advantage of the mpr_policy option in AIX 5.3).

To configure heartbeat over IP aliases, a base (starting) heartbeat alias address must be specified in the HACMP configuration. When HACMP starts, it will build up an alias heartbeat network starting from this address, by calculating an IP address for each node based on the node number. This is defined as a separate HACMP network with the number of subnets that matches the number of interfaces on a node. When specifying the heartbeat alias base address, the following rules apply:

- HACMP only supports the use of the subnet mask on the underlying adapter for the heartbeat over IP alias network.
- The base adapter subnet mask on the adapter must be greater than the number of nodes as each node will have an address on that subnet
- There must be sufficient address space above the specified base address to allow for one subnet for each interface on a node.
- There must be no addresses in the site within the range of aliases that HACMP creates. These address must also be out of any range of DNS etc.
- HACMP still requires that each interface can communicate with each other interface - that is are on the same physical network.

When heartbeat over IP aliases is configured, HACMP builds the required alias addresses following the above rules and loads this information into the HACMP ODM. When HACMP integrates a node into the cluster and RSCT starts, the alias addresses are added to each adapter under HACMP control. RSCT then uses these addresses to build up it's communication groups. So it is these IP alias addresses that RSCT monitors, not the base IP addresses on the interface.

Figure 2-8 on page 57 shows an example of a three-node cluster, with each node having three interfaces on the same physical network and the same subnet (See Table 2-1). The subnet mask on the base adapters was 255.255.255.0.

Table 2-1 Base IP addresses

	Node 1	Node 2	Node 3
en0	135.2.5.12	135.2.5.22	135.2.5.27
en1	135.2.5.13	135.2.5.23	135.2.5.28
en2	135.2.5.14	135.2.5.24	135.2.5.29

In this example HACMP will create three IP alias subnets (one for each interface) with three addresses on each (one for each node). Table 2-2 on page 56 shows

the IP addresses that HACMP would use for heartbeat over IP aliases if a base address of 198.10.1.1 was configured.

Attention: If a base of x.x.x.1 is selected, HACMP will start with x.x.x.2 as the first address on the first interface. However if you select x.x.x.0, HACMP will use that address and it will not be usable

Table 2-2 HACMP configured Heartbeat over alias IP addresses (base 198.10.1.1)

	Node 1	**Node 2**	**Node 3**
en0 - communication group 1	198.10.1.2	198.10.1.3	198.10.1.4
en1 - communication group 2	198.10.2.2	198.10.2.3	198.10.2.4
en2 - communication group 3	198.10.3.2	198.10.3.3	198.10.3.4

Note: None of the three subnets (198.10.1/24, 198.10.2/24, 198.10.3/24) need to be routable.

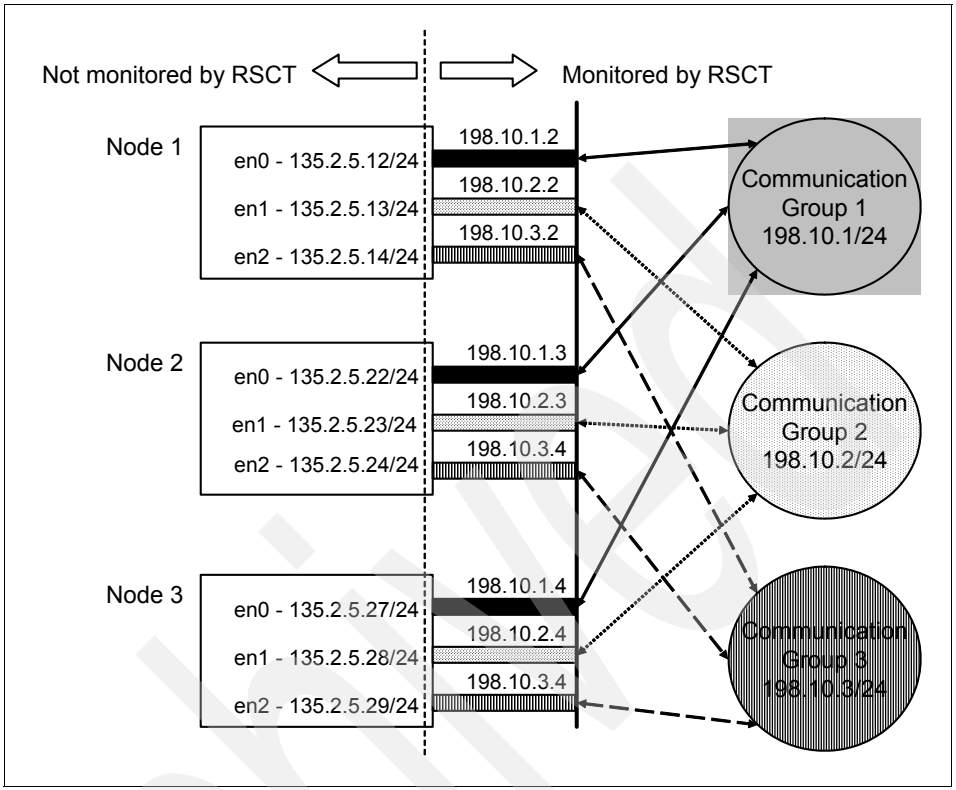

Figure 2-8 3 node cluster with heartbeat over IP alias

Figure 2-8 shows the three heartbeat rings (communication groups) that RSCT uses.

Heartbeat over IP aliases supports both IP address takeover mechanisms:

- **IPAT via replacement**: The service IP label will replace the boot IP address on the interface. The heartbeat IP alias address remains.
- **IPAT via aliasing**: The service IP label will be added as an alias on the interface with the heartbeat IP alias.

2.3.3 TCP/IP networks

The IP network types supported in HACMP 5.1 / 5.2 / 5.3 are:

- Ethernet (ether)
- Token-ring (token)
- Fiber Distributed Data Interface - FDDI (fddi)
- SP Switch1 and SP Switch2 (hps)

- Asynchronous transfer mode- ATM and ATM LAN Emulation) (atm)
- Etherchannel (ether)

The following IP network types are not supported (starting HACMP 5.1):

- Virtual IP Address (VIPA)
- Serial Optical Channel Converter (SOCC)
- Serial Line IP (SLIP)
- Fibre Channel Switch (FCS)
- IEEE 802.3
- IP Version 6 (IPV6)

Note: HACMP now supports IP over aggregated Ethernet (Etherchannel, IEEE 802.3ad) communication interfaces for IP address takeover in AIX 5L. Etherchannel is not supported for:

- Hardware address takeover
- PCI hot plug

HACMP is designed to work with any TCP/IP network, these networks are used to:

- Allow clients to access the nodes (i.e., applications)
- Enable the nodes to exchange heartbeat messages
- Serialize access to data (concurrent access environments, e.g., Oracle Real Application Cluster)

TCP/IP networks can be classified as:

- **Public**: These are logical networks designed for client communication to the nodes. Each is built up from a collection of the IP adapters, so that each network may contain multiple subnets. As these networks are designed for client access, IP Address takeover is supported.
- **Private**: These networks are designed for use by applications that don't support IP takeover - for example Oracle RAC or HAGEO geographic networks. All interfaces are defined as service and heartbeat packets are sent over these networks.

2.3.4 IP address takeover mechanisms

One of the key roles of HACMP is to maintain the service IP labels / addresses as highly available. HACMP does this by starting and stopping each service IP address as required on the appropriate interface. When a resource group is active on a node, HACMP supports two methods of activating the service IP addresses:

- By replacing the base (boot-time) IP address of an interface with the service IP address. This method is known as IP address takeover (IPAT) via IP replacement. This method also allows the takeover of a locally administered hardware address (LAA) - hardware address takeover.
- By adding the service IP address as an alias on the interface, i.e., in addition to the base IP address. This method is known as IP address takeover via IP aliasing. This is the default for HACMP 5.1 and above.

To change this behavior, the network properties can be changed using HACMP extended configuration menus.

It is worth noting that each method imposes subnet restrictions on the boot interfaces and the service IP labels, unless the heartbeat over IP alias feature is used.

IPAT via IP replacement

The service IP label / address replaces the existing address on the interface. Thus only one service IP label / address can be configured on one interface at one time. The service IP label should be on the same subnet as one of the base IP addresses. This interface will the first one used by a service IP label when a resource group becomes active on the node. Other interfaces on this node, cannot be in the same subnet, are traditionally referred to as standby interfaces, and are used if a resource group falls over from another node, or if the boot interface fails. See Figure 2-9 on page 60. This method may save subnets, but requires extra hardware.

When using IPAT via IP replacement (also known as “classic” IPAT), it is also possible to configure hardware address takeover (HWAT). For HWAT a locally administered MAC (Media Access Control) address becomes part of the service IP label definition, and at time of replacement, the MAC address on the interface will also be changed. This ensures that the ARP caches on the subnet do not need to be updated and that MAC address dependant applications will still point to the correct host.

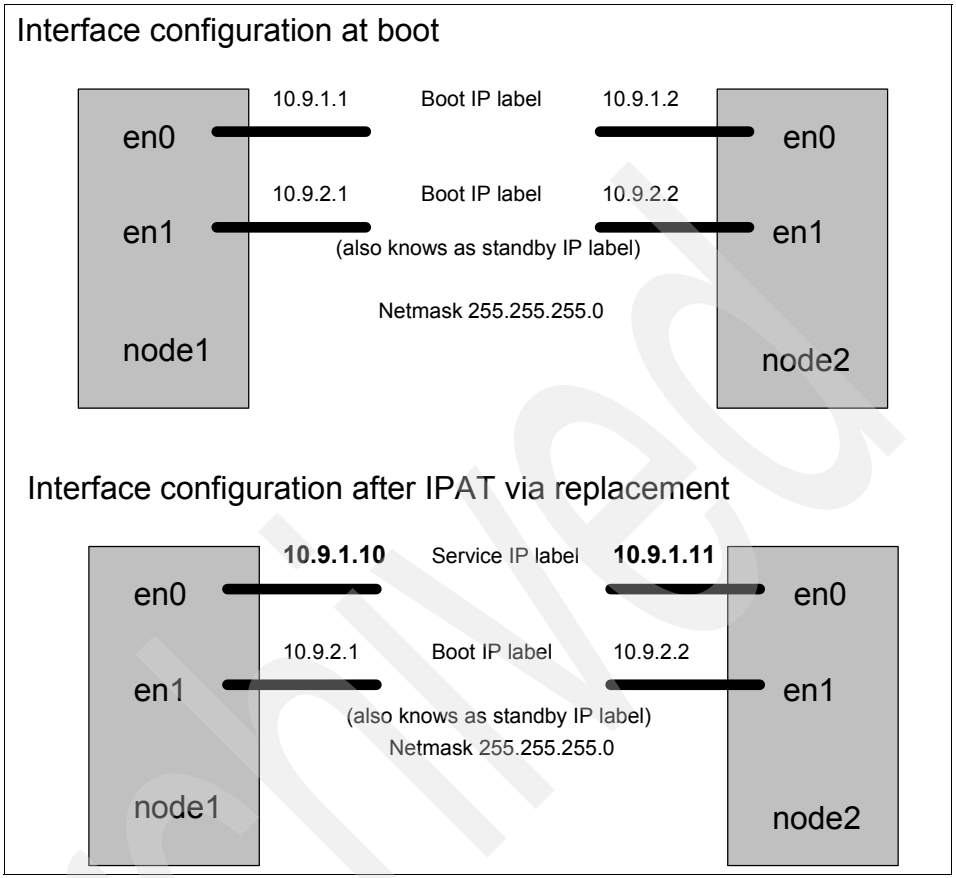

Figure 2-9 IPAT via IP replacement

If the interface holding the service IP address fails, when using the IPAT via IP replacement, HACMP moves the service IP address on another available interface on the same node and on the same network; in this case, the resource group associated is not affected.

If there is no available interface on the same node, the resource group is moved together with the service IP labels to another node with an available interface on the same logical network. If there are no nodes or interfaces available the resource group will go into an ERROR state. When HACMP recovers any adapters, resource groups in error state will be checked to determine if they can be brought back on line.

Restriction: With IPAT via replacement, RSCT and HACMP will have problems monitoring a node that has more than one resource group online if both Service IP labels are on the same subnet - as the node now has two interfaces with base addresses on the same subnet. For this reason heartbeat over IP alias is recommended or in AIX 5.3 mpr_policy can be set to 5

IPAT via aliasing

The service IP label / address is aliased onto the interface without removing the underlying boot IP address using the **`ifconfig`** command, see Figure 2-10. This has been the default since HACMP 5.1. IPAT via aliasing also does away with the concept of standby interfaces - all network interfaces are labeled as boot interfaces.

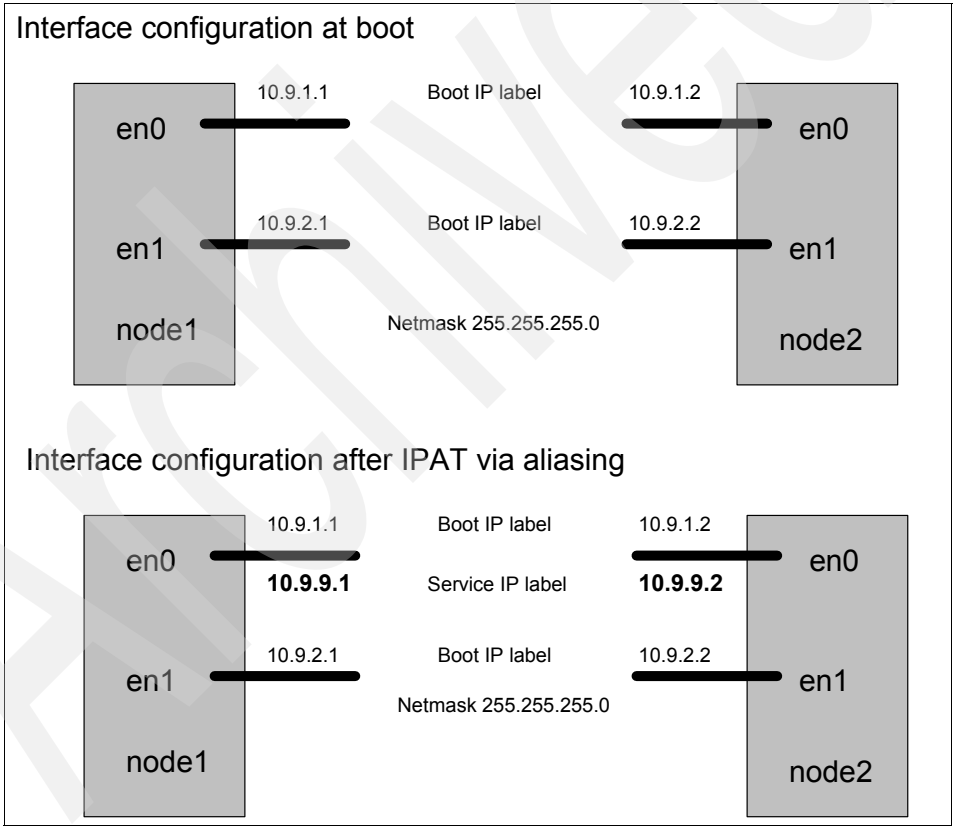

Figure 2-10 IPAT via IP aliases

As IP address are added to the interface via aliasing, more than one service IP label can coexist on the one interface. By removing the need for one interface

per service IP address that the node could host, IPAT via aliasing is the more flexible option and in some cases can require less hardware. IPAT via aliasing also reduces fallover time, as it is much faster to add an alias to an interface, rather than removing the base IP address and then apply the service IP address.

Even though IPAT via aliasing will support multiple service IP labels / addresses, it is still recommended that you configure multiple interfaces per node per network. It is far less disruptive to swap interfaces compared to moving the resource group over to another node.

IPAT via aliasing is only supported on networks that support the gratuitous ARP function of AIX. Gratuitous ARP is when a host sends out an ARP packet prior to using an IP address and the ARP packet contains a request for this IP address. As well as confirming that no other host is configured with this address, it will ensure that the ARP cache on each machine on the subnet is updated with this new address.

If there are multiple service IP alias labels / addresses active on one node, HACMP by default will equally distribute them amongst the available interfaces on the logical network. However in HACMP 5.3 this placement can be controlled. HACMP distributes the aliases on a node, by sorting the available interfaces by the number of aliases already placed on them, and then places the new aliases accordingly. This is only done at integration and fallover time, so if a new interface becomes available, no redistribution will be done.

For IPAT via aliasing each boot interface on a node must be on different subnet, though interfaces on different nodes can obviously be on the same subnet, unless, as mentioned above, heartbeat over IP alias is used. The service IP labels can be on one or more subnets, but they cannot be the same as any of the boot interface subnets.

Important: For IPAT via aliasing networks, HACMP will briefly have the service IP addresses active on both the failed Interface and the takeover interface so it can preserve routing. This may cause a “DUPLICATE IP ADDRESS” error log entry, which can be ignored.

2.3.5 Persistent IP label / address

A persistent node IP label is an IP alias that can be assigned to a network for a specified node. A persistent node IP label is a label that:

- Always stays on the same node (is node-bound)
- Co-exists with other IP labels present on the same interface

- Does not require the installation of an additional physical interface on that node
- Is not part of any resource group

Assigning a persistent node IP label for a network on a node allows you to have a highly available node-bound address on a cluster network. This address can be used for administrative purposes as it will always point to a specific node regardless of whether HACMP is running.

Note: It is only possible to configure one persistent node IP label per network per node. For example, if you have a node connected to two networks defined in HACMP, that node can be identified via two persistent IP labels (addresses), one for each network.

The persistent IP labels are defined in the HACMP configuration, and they become available as soon as the cluster definition is synchronized. A persistent IP label will remain available on the interface it was configured, even if HACMP is stopped on the node, or the node is rebooted. If the interface on which the persistent IP label is assigned fails while HACMP is running, the persistent IP label will be moved to another interface in the same logical network on the same node.

If the node fails or all interfaces on the logical network on the node fail, then the persistent IP label will no longer be available.

The following subnetting restrictions apply to the persistent IP label:

- **For IPAT via replacement networks**: The persistent IP alias must be on a different subnet to the standby interfaces and may be on the same subnet as the boot interfaces (same as the service IP labels)
- **For IPAT via aliasing networks**: The persistent IP alias must be on a different subnet to each of the boot interface subnets.

The persistent node IP labels can be created on the following types of IP-based networks:

- Ethernet
- Token Ring
- FDDI
- ATM LAN Emulator

Restriction: It is not possible to configure a persistent node IP label on the SP Switch, on ATM Classical IP, or on non-IP networks.

2.3.6 Device based or serial networks

Serial networks are designed to provide an alternative method for exchanging information via heartbeat packets between cluster nodes. In case of IP subsystem or physical network failure, HACMP can still differentiate between a network failure and a node failure when an independent path is available and functional.

Serial networks are a point to point network and therefore, if there are more than two nodes in the cluster, the serial links should be configured as a ring, connecting each node in the cluster. Even though each node will only be aware of the state of it's immediate neighbors, the RSCT daemons ensure that the group leader will be aware of any changes in state of any of the nodes.

Even though it is possible to configure an HACMP cluster without non-IP networks, We strongly recommended that you use at least one non-IP connection between each node in the cluster.

Currently HACMP supports the following types of device based networks for non-TCP/IP heartbeat exchange between cluster nodes:

The following devices are supported for non-IP (device-based) networks in HACMP:

- Serial RS232 (rs232)
- Target mode SCSI (tmscsi)
- Target mode SSA (tmssa)
- Disk heartbeat (diskhb)

The following types of serial network can be used:

RS232

A serial network using the RS232 ports, either the built-in serial ports or a multi-port serial adapter. Note: care should be taken ensuring that the ports selected support heartbeating.

The default baud rate for the RS232 network is 38400 and this should be changed if this is not supported by your modem (in case you want to use this network between remote locations).

Target mode SCSI

Another possibility for a non-IP network is a target mode SCSI connection. Whenever you use a shared SCSI device, you can also use the SCSI bus for exchanging heartbeats. Target mode SCSI (tmscsi) is only supported with SCSI-2 Differential or SCSI-2 Differential Fast / Wide devices. SCSI-1

Single-Ended and SCSI-2 Single-Ended do not support serial networks in an HACMP cluster.

Target mode SSA

If you are using shared SSA devices, target mode SSA can be used for non-IP communication in HACMP. This relies on the built in capabilities of the SSA adapters (using the SCSI communication protocol). The SSA devices in a SSA loop (disks and adapters) use the communication between “initiator” and “target”; SSA disks are “targets”, but the SSA adapter has both capabilities (“initiator” and “target”); thus, a tmssa connection uses these capabilities for establishing a serial-like link between HACMP nodes. This is a point-to point communication network, which can communicate only between two nodes.

To configure a tmssa network between two cluster nodes, one SSA Adapter on each node, in the same SSA loop, forms each endpoint.

Disk heartbeat network

In certain situations RS232, tmssa, and tmscsi connections are considered too costly or complex to set up. Heartbeating via disk (diskhb) provides users with an easy to configure alternative that requires no additional hardware. The only requirement is that the “disks” (physical disks or LUNs on external storage) be used in enhanced concurrent mode. Enhanced concurrent mode disks use RSCT group services to control locking - freeing up a sector on the disk that can now be used for communication. This sector was once used for SSA Concurrent mode disks, is now used for writing heartbeat information.

Any disk that is part of an enhanced concurrent VG can be used for a diskhb network, including those used for data storage. Further, the VG that contains the disk used for a diskhb network does not have to be varied on.

Any disk type may be configured as part of an enhanced concurrent VG, making this network type extremely flexible.The endpoint “adapters” for this network are defined as the node and physical volume pair.

In case of disk heartbeat, the recommendation is to have one point-to-point network consisting of one disk per pair of nodes per physical enclosure. One physical disk cannot be used for two point-to-point networks.

Note: An enhanced concurrent volume group is not the same as a concurrent volume group (which is part of a concurrent resource group), rather it refers to the mode of locking - using RSCT.

Heartbeat via disk

The heartbeat via disk (diskhb) is a new feature introduced in HACMP V5.1, with a proposal to provide additional protection against cluster partitioning and simplified non-IP network configuration, especially for environments where the RS232, target mode SSA, or target mode SCSI connections are too complex or impossible to implement.

This type of network can use any type of shared disk storage (Fibre Channel, SCSI, or SSA), as long as the disk used for exchanging KA messages is part of an AIX enhanced concurrent volume group. The disks used for heartbeat networks are not exclusively dedicated for this purpose; they can be used to store application shared data (see Figure 2-7 on page 53 for more information).

Customers have requested a target mode Fibre Channel connection, but due to the heterogeneous (nonstandard initiator and target functions) FC environments (adapters, storage subsystems, SAN switches, and hubs), this is difficult to implement and support.

By using the shared disks for exchanging messages, the implementation of a non-IP network is more reliable, and does not depend of the type of hardware used.

Moreover, in a SAN environment, when using optic fiber to connect devices, the length of this non-IP connection has the same distance limitations as the SAN, thus allowing very long point-to-point networks.

By defining a disk as part of an enhanced concurrent volume group, a portion of the disk will not be used for any LVM operations, and this part of the disk (sector) is used to exchange messages between the two nodes.

The specifications for using the heartbeat via disk are:

- One disk can be used for one network between two nodes. The disk to be used is uniquely identified on both nodes by its LVM assigned physical volume ID (PVID).
- The recommended configuration for disk heartbeat networks is one disk per pair of nodes per storage enclosure.
- Requires that the disk to be used is part of an the enhanced concurrent volume group, though it is not necessary for the volume group to be either active or part of a resource group (concurrent or non-concurrent). The only restriction is that the VG must be defined on both nodes.

Note: The cluster locking mechanism for enhanced concurrent volume groups does not use the reserved disk space for communication (as the “classic” clvmd does); it uses the RSCT group services instead.

2.3.7 Network modules

HACMP has a failure detection rate defined for each type of network and this can be configured using three predefined values or customized. The predefined values are slow, normal (the default) and fast, to customize, the interval between heartbeats and failure cycle can be set.

The interval between heartbeats (heartbeat rate - *hbrate*) defines the rate at which cluster services send “keep alive” packets between interfaces and devices in the cluster. The failure cycle (*cycle*) is the number of successive heartbeats that can be missed before the interface is considered to have failed. HACMP 5.3 supports sub-second heartbeat rates.

The time (in seconds) to detect a failure is:

```
hbrate * cycle * 2
```

HACMP uses double the failure cycle as the time to detect a failure to allow for both the node and it’s neighbors to reach the conclusion. Also to keep network traffic to a minimum, HACMP only sends out one packet and expects to receive one, per logical network per heartbeat interval.

For serial networks, HACMP will declare the neighbor down after the failure detection rate has elapsed for that network type. HACMP will wait the same period again before declaring the device down - if no still no heartbeats are received from the neighbor. HACMP will not run the network_down event until both the local and remote devices have failed.

However if the serial network is the last network left connecting to a particular node, the node_down event will be triggered once the interface is detected as down.

The device based networks different from IP based networks, as device based networks cannot distinguish between a interface down and a network down. Disk heartbeating is the exception. If the node can reach the disk, the interface is considered to be up, and the network is considered up if messages are being exchanged.

There is another characteristic that is defined for each network. This is the network grace period and it is the period of time after a particular network failure is detected, that failures of the same network type will be ignored. This gives the cluster time to make changes to the network configuration without detecting any false failures

Attention: The time to detect failure must be the same for all networks in use in the cluster.

2.3.8 Clients

A client is system that can access cluster nodes over the network. They run some “front end” or client application that communicates with the application running in the cluster via the service IP labels. HACMP ensures that the application is highly available for the clients, but they are not highly available themselves.

AIX 5L clients can make use of the cluster information (clinfo) services to receive notice of cluster events. Clinfo provides an API that displays the cluster status.

During resource group takeover, the application is started on another node, so clients must be aware of the action. In certain cases, the applications client uses the ARP cache on the client machine to reconnect to the server. In this case, there are two possible situations if the client is on the same subnet as the cluster nodes:

- If IPAT via replacement is configured for the network that the applications service IP labels use, then MAC address takeover occurs as well, so there is not need to update the client machine’s ARP cache.
- If IPAT via aliasing is configured for the network, then a gratuitous ARP packet is sent out, so again the client’s ARP cache shouldn’t need updating.
- However if the client does not support the gratuitous ARP, their cache can be updated by /usr/es/sbin/cluster/etc/clinfo.rc. Whenever there is a network change, clinfo.rc will send one ping to each address or label in the PING_CLIENT_LIST variable. So to ensure that the clients’ ARP cache is updated, add each clients address or label to the PING_CLIENT_LIST in clinfo.rc.

However if the client is on another subnet, then the above conditions apply to the router.

Clients running the clinfo daemon will be able to reconnect to the cluster quickly after a cluster event.

2.3.9 Network security considerations

HACMP security is important to both limit unauthorized access to the nodes and unauthorized interception of inter-node communication. Earlier versions of HACMP used rsh to execute commands on other nodes. This was both difficult to secure and IP addresses could be spoofed. HACMP now uses it’s own daemon, the cluster communication daemon (**clcomdES**) to control communication between the nodes.

HACMP provides cluster security by:

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

For details on user access and cluster security, see Chapters 15 and 16 in *High Availability Cluster Multi-Processing Administration Guide*, SC23-4862-06.

Connection authentication and encryption

Authentication ensures the origin and integrity of the message while encryption ensures that only the sender and recipient of the message are aware of its contents.

The following can be used for connection authentication:

- **Standard authentication**: This is the default, the communication daemon, clcomd authenticates against the IP address and limits the commands that can be run with root privilege. There is a set of HACMP commands (those in **`/usr/es/sbin/cluster`**) that can be run as root, the remaining commands are run as nobody.
- **Kerberos authentication**: Kerberos authentication is supported in the SP environment.
- **Virtual private network**: A VPN can be configured for internode communications and the persistent alias labels should be used to define the tunnels.

HACMP supports the following encryption:

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

The key files are stored in /usr/es/sbin/cluster/etc.

Note: This encryption only applies to **`clcomdES`** not the cluster manager.

The cluster communications daemon

With the introduction of **`clcomdES`**, there is no need for an **`/.rhosts`** file to be configured. However some applications may still require the file to be present. The cluster communications daemon runs remote commands based on the principle of “least privilege” This ensures that no arbitrary command can run on a remote node with root privilege. Only a small set of HACMP commands are “trusted” and allowed to run as root - these are the commands /usr/es/sbin/cluster. The remaining commands are run as nobody.

The cluster communications daemon is started by inittab, with the entry being created by the installation of HACMP. The daemon is controlled by the system resource controller, so startsrc, stopsrc and refresh work. In particular refresh is used to re-read /usr/es/sbin/cluster/etc/rhosts and moving the log files.

The cluster communication daemon uses port 6191 and authenticates incoming connects by checking them against:

If /usr/es/sbin/cluster/etc/rhosts file doesn't exist - all connections refused

If the file does exist then connections will be checked against (in order):

- HACMPnode ODM class
- HACMPadapter ODM class
- /usr/es/sbin/cluster/etc/rhosts file

The /usr/es/sbin/cluster/etc/rhosts file is populated when the first synchronization is run with the interface addresses from the synchronizing node (it will still be blank on the synchronizing node). After the first synchronization, the HACMP ODM classes will be populated, so the rhosts file can be emptied.

The real use of the file is before the cluster is first synchronized in an insecure environment. Populate the file on each node with only interface addresses of nodes in the cluster - and no other system will be able to communicate via clcomdES.

Tip: When the initial cluster synchronization is done, the /usr/es/sbin/cluster/etc/rhosts file is populated with the interface addresses of the synchronizing node.

The requesting host is asked to supply their IP label that matches the address found in the above location, and if a valid response is given, the connection is allowed. If all the above entries are empty, the daemon will assume that the cluster hasn't been configured, so will accept incoming entries.

Important: An invalid entry in /usr/es/sbin/cluster/etc/rhosts will cause clcomdES to deny all connections

The cluster communications daemon provides the transport medium for HACMP cluster verification, global ODM changes and remote command execution. The following commands use clcomdES (not supported by use by a user):

`clrexec` to run specific and potentially dangerous commands

`cl_rcp` to copy AIX configuration files

cl_rsh used by the cluster to execute commands in a remote shell.

The cluster communication daemon also offers performance improvement over traditional rsh/rcp communications (“r” command are slow), and clcomdES also keeps the socket connections open, rather than closing after each operation. As many HACMP administration operations require access to the ODM the cluster communications daemon also caches copies of each nodes ODM. When the ODM needs to be accessed, clcomdES will compare the checksum of the cached ODM entries against the ODM on each node in the cluster, and only update as required.

Cluster communications daemon also sends it’s own heartbeat packets out to each node, and will attempt to re-establish connection if there is a network failure.

The cluster communications daemon is also used for (HACMP 5.2 and later):

- file collections
- auto synchronization and automated cluster verification
- user / passwords administration
- C-SPOC

Unlike in HACMP 5.1, starting with HACMP 5.2 it is no longer possible to stop the cluster communications daemon while the cluster is active.

2.4 Resources and resource groups

This section describes the HACMP resource concepts:

- Definitions
- Resources
- Resource groups

2.4.1 Definitions

HACMP uses the underlying topology to ensure that the applications under its control and the resources they require are keep highly available. These resources include:

- Service IP labels / addresses
- Physical disks
- Volume groups
- Logical volumes
- File systems
- Network File Systems
- Application servers (applications)

- Communication adapters and links
- Tape resources
- Fast connect resources
- WLM integration

The applications and the resources required are configured into a resource groups. The resource groups are controlled by HACMP as single entities - who's behavior can be tuned to meet the requirements of the clients / users.

Figure 2-11 shows the resources that HACMP makes highly available superimposed on the underlying cluster topology:

- Service IP Labels
- Applications shared between nodes
- Storage shared between nodes

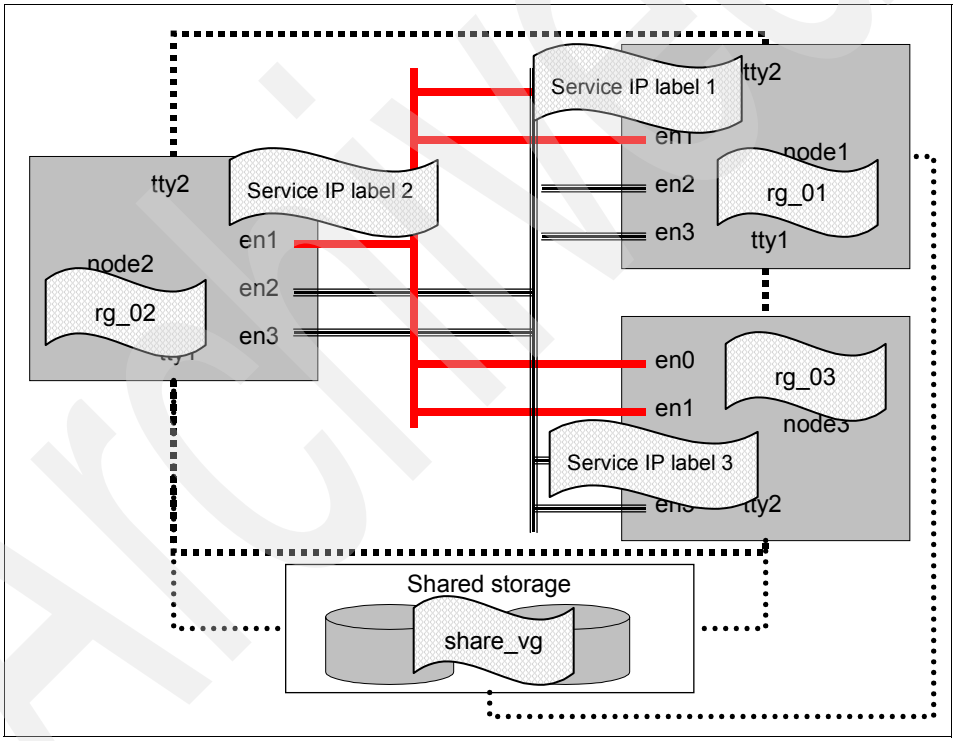

Figure 2-11 Highly available resources superimposed on the cluster topology

2.4.2 Resources

The following are considered resources in a HACMP cluster:

Service IP address / label

As previously discussed, the service IP address is an IP address used by clients to access the applications or nodes. This service IP address (and its associated label) is monitored by HACMP and is part of a resource group.There are two types of service IP address (label):

- **Shared service IP address (label)**: An IP address that can be configured on multiple nodes and is part of a resource group that can be active only on one node at a time.
- **Node-bound service IP address (label)**: An IP address that can be configured only one node (is not shared by multiple nodes). Typically, this type of service IP address is associated with concurrent resource groups.

The service IP addresses become available when HACMP brings the associated resource group into an ONLINE status.

In HACMP 5.3, the placement of the service IP labels to be specified using the following distribution preferences:

- **Anti-Collocation**: This is the default and HACMP will distribute the service IP labels across all the boot IP interfaces in the same HACMP network on the node.
- **Collocation**: HACMP will allocate all service IP addresses on the same boot IP interface.
- **Collocation with persistent label**: HACMP will allocate all service IP addresses on the boot IP interface that is hosting the persistent alias IP label. This may be useful in environments with VPN and firewall configuration, where only one interface is granted external connectivity.
- **Anti-Collocation with persistent label**: HACMP will distribute all the service IP labels across all the boot IP interfaces in the same logical network, that are not hosting the persistent alias IP label. If no other interfaces are available, the service IP labels will share the adapter with the persistent alias IP label.

It should be noted that if there are insufficient interfaces to satisfy the selected distribution preference, then HACMP will distribute IP labels using the interfaces available - to ensure that the service IP labels are available.

The IP label distribution preference can also be changed dynamically - but will only be used in subsequent cluster events. This is to avoid any extra interruptions in service. The command **`cltopinfo -w`** will display the policies.

Storage

The following storage types can all be configured as resources:

- Volume groups (AIX and Veritas VM)

- Logical volumes (all logical volumes in a defined VG)
- File systems (jfs and jfs2) - either all for the defined VGs or can be specified individually
- raw disks - defined by PVID

If storage is to be shared by some or all of the nodes in the cluster then all components must be on external storage and configured in such a way that failure of one node will not affect the access by the other nodes (for example, check the loop rules carefully if using SSA).

There are two ways the storage can be accessed:

- Non-concurrent configurations where one node will own the disks, allowing clients to access them with the other resources required by the application. If this node fails, HACMP will determine the next node to take ownership of the disks, restart applications and provide access to the clients. Enhanced concurrent mode disks are often used in non-concurrent configurations, remember that enhanced concurrent mode refers to the method of locking access to the disks, not whether the access itself will be concurrent or not.
- Concurrent configurations, one or more nodes will be able to access the data concurrently with locking controlled by the application. The disks must be in concurrent volume group.

HACMP supports the following disk technologies as shared external disks:

- SCSI
- SSA
- Fibre channel attached disks systems

Devices supported:

- Traditional SCSI disks and enclosures
- SSA disks and enclosures
- FastT/DS4xxx storage servers
- 2105 Enterprise Storage servers and DS8xxx and 6xxx
- Some 3rd party storage devices

Important: Third party storage subsystems and devices may not be directly supported by IBM, rather by vendors. A list of non-IBM storage devices may be found at:

http://www.availant.com

Multi-path software:

- Data path devices (vpath)
- MPIO

- DAC / DAR

Supported parallel SCSI devices:

- SCSI disk devices
- SCSI disk enclosures
- FastT/DS4xxx Storage Servers
- 2105 Enterprise storage servers (with SCSI connections)

Typically, parallel SCSI devices can be configured in clusters of up to four nodes, where all nodes are connected to the same SCSI bus. Up to 16 devices can be connected to the one SCSI bus (including the SCSI adapters). However, it is not recommended to connect devices other than disks (such as CD-ROMs and tapes).

IBM 2105 Enterprise Storage Server

IBM 2105 Enterprise Storage Server® provides concurrent attachment and disk storage sharing for a variety of open systems servers.

Due to the multitude of platforms supported in a shared storage environment, to avoid interference, it is very important to configure secure access to storage by providing appropriate LUN masking and zoning configurations.

The ESS uses IBM SSA disk technology. ESS provides built-in availability and data protection. RAID technology is used to protect data. Also, the disks have intrinsic predictive failure analysis features to predict errors before they affect data availability.

The ESS has virtually all components doubled and provides protection if any internal component fails. The ESS manages the internal storage (SSA disks) with a cluster of two nodes connected through a high speed internal bus, each of the nodes providing the exact same functionality, Thus, in case one of the internal node fails, the storage remains available to the client systems.

For more information about planning and using the 2105-800 Enterprise Storage Server (including attachment diagrams, and more), see the following Web site:

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

An example of a typical HACMP cluster using ESS as shared storage is shown in Figure 2-12 on page 76.

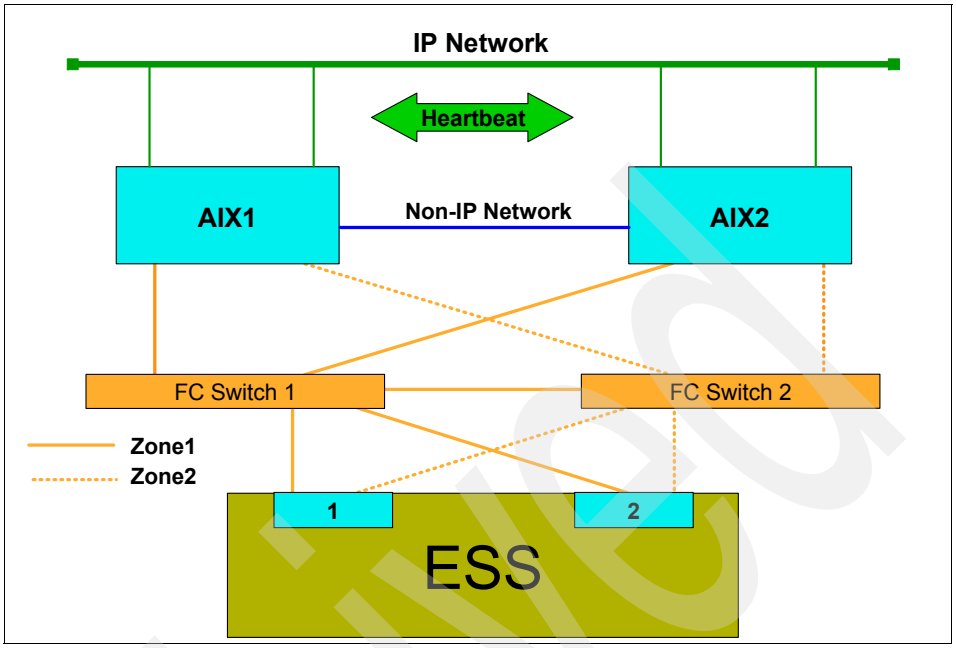

Figure 2-12 ESS Storage

IBM FAStT/DS4xxx Storage Servers

IBM FAStT/DS4xxx Storage Servers deliver flexible and high performance and reliability for applications in multi host environments.

The FAStT/DS4xxx architecture, although not as sophisticated as the one implemented in the ESS, is also based on maximizing the redundant components - storage controllers, power supplies, and storage attachment adapters.

The FAStT/DS4xxx architecture implements native Fibre Channel protocol on both host side and storage side. It does not offer SCSI support, and does not accommodate a dedicated high speed bus between the two controllers, but it provides controller fail-over capability for uninterrupted operations, and host side data caching.

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

`http://www.storage.ibm.com/disk/fastt/index.html`

For a typical FAStT/DS4xxx connection to an HACMP cluster, see Figure 2-13 on page 77.

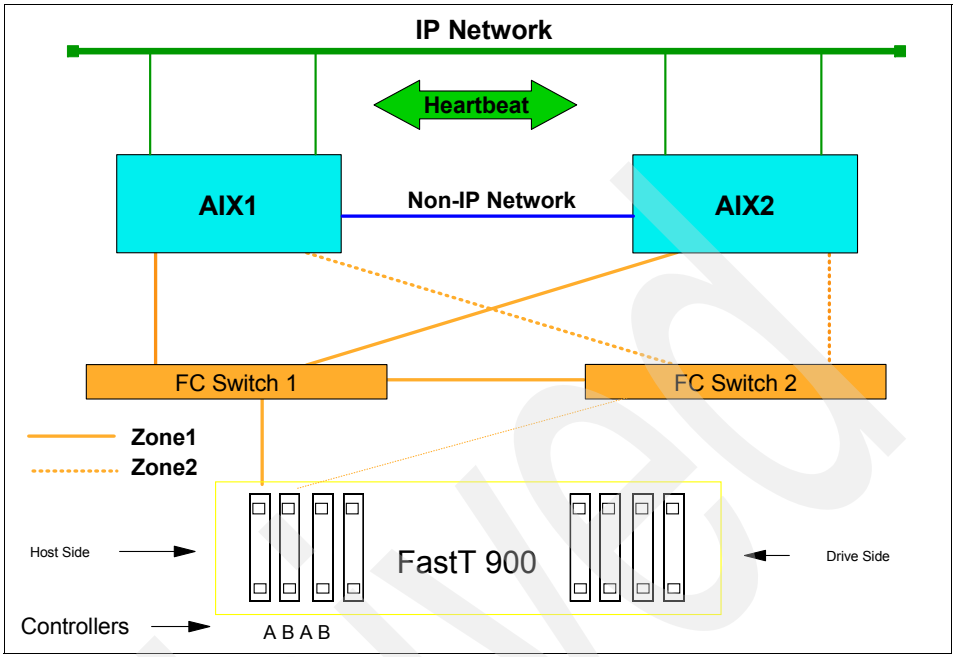

Figure 2-13 DS4xxx Storage

IBM Serial Storage Architecture disk subsystem

Serial Storage Architecture (SSA) storage subsystems provide a more “discrete components” solution, offering features for reducing the number of single points of failure.

SSA storage provides high availability in an HACMP environment through the use of redundant hardware (power supplies and storage connections) and hot swap capability (concurrent maintenance) for power supplies and disks.

SSA storage also offers RAID capability at the adapter (Host Bus Adapter - HBA) level.

Note: By using the SSA RAID option, the number of HACMP nodes able to share the same data is limited to two.

IBM 7133 SSA disk subsystems can be used as shared external disk storage devices to provide concurrent access in an HACMP cluster configuration.

SSA storage provides a flexible, fairly simple, more “custom” approach for configuring HACMP clusters with existing or legacy applications and a limited

number of nodes. We recommend that all new configurations to be implemented using the new technologies (FC storage).

For an example of a two node HACMP cluster, see Figure 2-14.

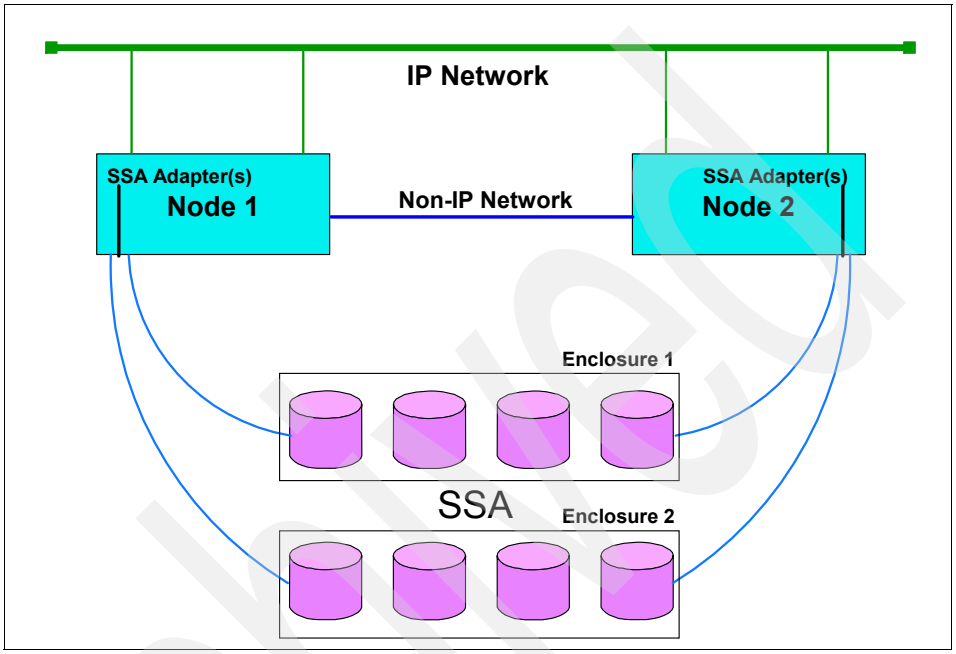

Figure 2-14 SSA storage

Choosing Data protection

Storage protection (data or otherwise) is independent of HACMP; for high availability of storage, you must use storage that has proper redundancy and fault tolerance levels. HACMP does not have any control on storage availability. For data protection, you can use either RAID technology (at storage or adapter level) or AIX LVM mirroring (RAID 1).

- Redundant Array of Independent Disks (RAID)

  Disk arrays are groups of disk drives that work together to achieve data transfer rates higher than those provided by single (independent) drives. Arrays can also provide data redundancy so that no data is lost if one drive (physical disk) in the array fails. Depending on the RAID level, data is either mirrored, striped, or both. For the characteristics of some widely used RAID levels, see Table 2-3 on page 81.

– RAID 0

  RAID 0 is also known as data striping. Conventionally, a file is written out sequentially to a single disk. With striping, the information is split into chunks (fixed amounts of data usually called blocks) and the chunks are written to (or read from) a series of disks in parallel. There are two performance advantages to this:

  Data transfer rates are higher for sequential operations due to the overlapping of multiple I/O streams.

  Random access throughput is higher because access pattern skew is eliminated due to the distribution of the data. This means that with data distributed evenly across a number of disks, random accesses will most likely find the required information spread across multiple disks and thus benefit from the increased throughput of more than one drive.

  RAID 0 is only designed to increase performance. There is no redundancy, so each disk is a single point of failure.

– RAID 1

  RAID 1 is also known as disk mirroring. In this implementation, identical copies of each chunk of data are kept on separate disks, or more commonly, each disk has a “twin” that contains an exact replica (or mirror image) of the information. If any disk in the array fails, then the mirror disk maintains data availability. Read performance can be enhanced because the disk that has the actuator (disk head) closest to the required data is always used, thereby minimizing seek times. The response time for writes can be somewhat slower than for a single disk, depending on the write policy; the writes can either be executed in parallel (for faster response) or sequential (for safety).

– RAID 2 and RAID 3

  RAID 2 and RAID 3 are parallel process array mechanisms, where all drives in the array operate in unison. Similar to data striping, information to be written to disk is split into chunks (a fixed amount of data), and each chunk is written out to the same physical position on separate disks (in parallel). When a read occurs, simultaneous requests for the data can be sent to each disk. This architecture requires parity information to be written for each stripe of data; the difference between RAID 2 and RAID 3 is that RAID 2 can utilize multiple disk drives for parity, while RAID 3 can use only one. If a drive should fail, the system can reconstruct the missing data from the parity and remaining drives. Performance is very good for large amounts of data, but poor for small requests, since every drive is always involved, and there can be no overlapped or independent operation.

– RAID 4

RAID 4 addresses some of the disadvantages of RAID 3 by using larger chunks of data and striping the data across all of the drives except the one reserved for parity. Using disk striping means that I/O requests need only reference the drive that the required data is actually on. This means that simultaneous, as well as independent reads, are possible. Write requests, however, require a read / modify / update cycle that creates a bottleneck at the single parity drive. Each stripe must be read, the new data inserted, and the new parity then calculated before writing the stripe back to the disk. The parity disk is then updated with the new parity, but cannot be used for other writes until this has completed. This bottleneck means that RAID 4 is not used as often as RAID 5, which implements the same process but without the bottleneck.

– RAID 5

RAID 5 is very similar to RAID 4. The difference is that the parity information is also distributed across the same disks used for the data, thereby eliminating the bottleneck. Parity data is never stored on the same drive as the chunks that it protects. This means that concurrent read and write operations can now be performed, and there are performance increases due to the availability of an extra disk (the disk previously used for parity). There are other possible enhancements to further increase data transfer rates, such as caching simultaneous reads from the disks and transferring that information while reading the next blocks. This can generate data transfer rates that approach the adapter speed.

As with RAID 3, in the event of disk failure, the information can be rebuilt from the remaining drives. A RAID 5 array also uses parity information, though it is still important to make regular backups of the data in the array. RAID 5 arrays stripe data across all of the drives in the array, one segment at a time (a segment can contain multiple blocks). In an array with n drives, a stripe consists of data segments written to “n-1” of the drives and a parity segment written to the “n-th” drive. This mechanism also means that not all of the disk space is available for data. For example, in an array with five 72 GB disks, although the total storage is 360 GB, only 288 GB are available for data.

– RAID 0+1 (RAID 10)

RAID 0+1, also known as IBM RAID-1 Enhanced, or RAID 10, is a combination of RAID 0 (data striping) and RAID 1 (data mirroring). RAID 10 provides the performance advantages of RAID 0 while maintaining the data availability of RAID 1. In a RAID 10 configuration, both the data and its mirror are striped across all the disks in the array. The first stripe is the data stripe, and the second stripe is the mirror, with the mirror being placed on the different physical drive than the data. RAID 10

implementations provide excellent write performance, as they do not have to calculate or write parity data. RAID 10 can be implemented via software (AIX LVM), hardware (storage subsystem level), or in a combination of the hardware and software. The appropriate solution for an implementation depends on the overall requirements. RAID 10 has the same cost characteristics as RAID 1.

Important: While all the RAID levels (other than RAID 0) have data redundancy, data should be regularly backed up. This is the only way to recover data in the event that a file or directory is accidentally corrupted or deleted.

The most common RAID levels used in today's IT implementations are listed in Table 2-3.

Table 2-3 Characteristics of RAID levels widely used

RAID level	Available disk capacity	Performance in read / write operations	Cost	Data Protection
RAID 0	100%	High both read / write	Low	No
RAID 1	50%	Medium / High read, Medium write	High	Yes
RAID 5	80%	High read Medium write	Medium	Yes
RAID 0+1	50%	High both read / write	High	Yes

LVM Quorum issues

Quorum must be enabled for concurrent VGs as each node may be accessing different disk - data divergence.

Leaving quorum on (by default) will cause RG fallover if quorum lost, and the VG will be forced varyon on other node if forced varyon of volume groups has been enabled. When forced varyon of VGs is enabled, HACMP checks:

- That there is at least one copy of each mirrored set in the volume group
- Each disk is readable
- There is at least one accessible copy of each logical partition in every logical volume.

If these conditions are fulfilled, then HACMP forces the VG varyon.

Using enhanced concurrent mode volume groups

Traditionally access to a volume group was controlled by SCSI locks, and HACMP had utilities to break these locks if a node didn't release the volume groups cleanly. With AIX 5.x the enhanced concurrent volume group was introduced and RSCT was used for locking. A further enhancement was the ability to active the volume group in two modes:

- **Active state**: The VG behaves the same way as the traditional varyon, operations can be performed on the volume group, logical volumes and file systems can be mounted.
- **Passive state**: The passive state allows limited read only access to the VGDA and the LVCB.

When a node is integrated into the cluster, HACMP will build a list of all enhance concurrent volume groups that are a resource in any resource group containing the node. These volume groups will then be activated in passive mode

When the resource group comes online on the node, the enhanced concurrent volume groups will then be activated in active mode. When the resource group goes offline on the node, the volume group will be returned to passive mode.

The **lspv** and **lsvg** commands can be used to show the state of an enhance mode concurrent volume group; **lspv** will list active, passive, **lsvg** will show active under VG STATE for both modes, VG PERMISSION will be read / write or passive-only.

Important: It is important when using enhanced concurrent volume groups that multiple networks exist for RSCT heartbeats. As there is no SCSI locking, a partitioned cluster can very quickly active a volume group, and then potentially corrupt data.

RAID and SSA concurrent mode

Since HACMP 5.x, enhanced concurrent mode (ECM) volume groups must be used for 64 bit kernels. Support continues for RAID and SSA concurrent modes on the 32bit kernel, but since AIX 5.2 it is not possible to create concurrent volume groups in any mode other than enhanced.

- Enhanced concurrent mode is recommended as it
- supports disk heartbeat - more flexible than using serial connections
- fast disk takeover - cuts down on fallover time
- easier to keep VGDA consistent across nodes.

Shared Physical volumes

For applications that access the raw disk, the physical volume identifier (PVID) can be added as a resource in a resource group.

Shared logical volumes

Whilst not explicitly configured as a part of a resource group, each logical volume in a shared VG will be available on a node when the resource group is online. These shared logical volumes can be configured to accessible by one node at a time or concurrently by a number of nodes. If the ownership of the LV needs to be modified, remember to re-set it after each time the parent volume group is imported.

Although this is not an issue purely related to HACMP, be aware that some applications using raw logical volumes will start writing from the beginning of the device - therefore overwriting the logical volume control block (LVCB).

Custom disk methods

The extended resource SMIT menus allow the creation of custom methods to handle disks, volumes and file systems. To create a custom method, you need to define to HACMP the appropriate scripts to manage the item in a highly available environment, for example:

For custom disks, HACMP provides scripts to identify ghost disks, determines if a reserve is held, breaks a reserve and makes the disk available

For volume groups: HACMP provides scripts to list volume group names, list the disks in the volume group, bring the volume group online and offline.

For file systems, HACMP provides scripts to mount, unmount, list and verify status.

In HACMP 5.3, custom methods are provided for Veritas Volume Manager (VxVM) using the Veritas foundation suite v4.0.

File systems (jfs and jfs2) - fsck and logredo

AIX native file systems use database journaling techniques to maintain their structural integrity. So after a failure, AIX will use the journal file system log (JFSlog) to restore the file system to it's last consistent state. This is faster than using the **`fsck`** utility. If the process of replaying the JFSlog fails, there will be an error and the file system will not be mounted.

The **`fsck`** utility performs a verification of the consistency of the file system, checking the inodes, directory structure and files. While this is more likely to recover damaged file systems, it does take longer.

Important: Restoring the file system to a consistent state does not guarantee that the data is consistent, that is the responsibility of the application.

2.4.3 NFS

HACMP works with AIX network file system (NFS) to provide a highly available NFS server, that allows the backup NFS server to recover the current NFS activity should the primary NFS server fail. This feature is only available for 2 node clusters as HACMP preserves locks for the NFS file systems and handles the duplicate request cache correctly. The attached clients will experience the same hang if the NFS resource group is acquired by another node as they would if the NFS server reboots.

When configuring NFS through HACMP, you can control:

- The network that HACMP will use for NFS mounting
- NFS exports and mounts at the directory level
- Export options for NFS exported directories and file systems. This information is kept in **/usr/es/sbin/cluster/etc/exports**, which is the same format as the AIX exports file - /**etc/exports**.

NFS and HACMP restrictions

- Only 2 nodes in the cluster
- Shared volume groups that contain file systems that will be exported by NFS must have the same major number on all nodes, or the client applications will not recover on a fallover.
- If NFS exports are defined on the node through HACMP, all NFS exports must be controlled by HACMP. AIX and HACMP NFS exports cannot be mixed.
- If a resource group has NFS exports defined, the field “file systems mounted before IP configured” must be set to true.
- By default, a resource group that contains NFS exported file systems, will automatically be cross-mounted. This also implies that each node in the resource group will act as an NFS client, so must have a IP label on the same subnet as the service IP label for the NFS server.

NFS cross-mounts

NFS cross-mounts work as follows:

- The node that is hosting the resource group NFS exports all NFS file systems.

- Each node in the resource group mounts all the NFS file systems defined in the resource group.
- If the resource group is acquired by another node, that node will mount the NFS file systems and then re-export them.

For example

- Node1 with service IP label svc1 will mount and export /fs1
- Node2 will mount svc1:/fs1 on /mntfs1
- Node1 will also mount svc1:/fs1 on /mntfs1

2.4.4 Applications servers

Virtually any application that can run on a standalone AIX Server can run in a clustered environment protect by HACMP. The application must be able to be started and stopped by scripts and able to be recovered by a script after an unexpected shutdown.

Applications are defined to HACMP as an application server with the following attributes:

- **Start script**: This script must be able to start the application from both a clean and an unexpected shutdown. Output from the script will be logged in the **`hacmp.out`** log file. The exit code from the script will be monitored by HACMP.
- **Stop script**: This script must be able to successfully stop the application. Output is also logged and the exit code monitored.
- **Application monitors**: To keep applications highly available, HACMP is able to monitor the application itself, not just the required resources.

Since HACMP 5.2 these scripts are the same on all nodes, so if node specific configurations are necessary, the hostname of the machine should be checked as part of the routine.

As the exit code from the application scripts are monitored, HACMP will assume that a non-zero return code from the script means that the script failed and therefore the start or stop of the application was not successful. If this is the case, the resource group will go into error and a **`config_too_long`** recorded.

When configuring the application for HACMP, the follow should be considered:

- The application is compatible with the version of AIX
- The storage environment is compatible with a highly available cluster
- The application and platform interdependencies must be well understood. The location of the application code, data, temporary files, sockets, pipes and

other components of the system such as printers must be replicated across all nodes that will host the application.

- As already discussed, the application must be able to be started and stopped without any operator intervention - particularly after an unexpected halt of a node. The application start and stop scripts must the thoroughly tested before implementation and with every change in the environment.
- The resource group that contains the application, must contain all the resources required by the application, or be the child of one that does.
- Application licensing must be taken into account. Many applications have licenses that depend on the CPU ID, careful planning must be done to ensure that the application can start on any node in the resource group node list. Care must also be taken with the numbers of CPU's etc. on each node, as some licensing is sensitive to this as well.

2.4.5 Application monitors

HACMP uses application monitors to ensure that applications are kept highly available. Since HACMP 5.2 each application can have multiple monitors.

HACMP uses two types of application monitors:

- **Process monitors**: Detects the termination of one or more processes, using RSCT Resource Monitoring and Control (RMC). Care must be taken in selecting the correct process name - use the output from `ps -el` to ensure that you have selected the correct process. More than one process can be monitored. One application monitor can monitor a fixed number of the same process, or single instances of a variety of processes.
- **Custom monitors**: Tests the health of an application with a user customized script. The method must be an executable programme (can be a shell script) that tests the application and exits. If a zero is returned, the application is considered to be healthy, a non-zero value indicates that the application has not started or failed.
  For example, a monitor for a database could query the database, and return an exit code depending on the response.

Since HACMP 5.2, each type of monitor can have different modes of operation:

- **Startup monitor**: The application monitor checks that the application server has started successfully within the specified stabilization interval. This type of monitor is specifically designed for parent resource groups in a parent / child dependency relationship, as HACMP runs the application start script in the background and doesn't wait for it to be completed. Using a startup monitor, the child resource groups will not be brought on line until the monitor confirms that the application has been successfully started. If there is more than one

startup monitor configured for an application, the longest stabilization interval is the event time used.

- **Long running monitor**: The application monitor periodically checks that the application server is running once the specified stabilization interval has passed. This is the default mode.
- **Both**: The application monitor will check that the application has started during the stabilization period, then monitor it periodically after the stabilization interval has passed.

When configuring process monitors, the following need to be defined:

- Monitor Name:Unique name
- Monitor Mode:Startup, long running or both
- Application server:The application server to be monitored
- Stabilization interval:Depending on the mode
  - In startup mode, the application will be monitored regularly during this period to determine if it has started successfully. If the monitor reports that the application has started successfully, then HACMP terminates the monitor and continues processing. However if it hasn't started successfully by the end of the period the resource group acquisition is considered to have failed and HACMP will start recovery actions.
  - In long running mode, HACMP will wait this time for the application to stabilize before starting the monitoring of the application.
  - In both mode, the application will be checked periodically during this period to see if the application started successfully, then after this time expires, it will be monitored to ensure that it keeps running.
- Restart count:This is the number of times that HACMP will attempt to restart the application before HACMP taking the action specified below. This is 0 for startup monitoring
- Restart interval:This is the time that the application must be stable before resetting the restart count. It is recommended that it be equal or greater than the **`restart count x stabilization interval`**. The default is 110% of this value
- Action if application fails*:
  If the application fails to restart after the restart count number of attempts, the action that HACMP takes can be set. The default action is notify, which runs a notification event. The other choice is fallover, where HACMP will acquire the resource group on the next node in the resource groups node list, or next node that matches the dynamic node priority criteria.
- Notify method:The notify method that can be run if the application fails

- Cleanup method*:The optional method that HACMP can use to clean up the application after it has failed, before attempting the restart method. If there is only one application server, this will default to it's the stop script, however care should be taken as the application is already stopped this script may fail.
- Restart method*:The optional method that HACMP will use to attempt to restart the application, if the restart count is not zero. Again if there is one on application server, this will default to it's start script.

* Not relevant for startup monitors

For process monitors:

- Process to monitor:The name of the processes to monitor (use **`ps -el`** to determine the correct name to use).
- Process owner:The user name for the own of all the processes being monitored (e.g., **`db2adm`**)
- Instance count:The number of instances must be specified. If only one process is being monitored, then this must equal the number of instances, and an error will be reported if there are more or less. If more than one process is being monitored, then there can be only 1 instance of each.

For custom monitors:

- Monitor method:The programme name that will be used by HACMP to monitor the application server. If it returns a non-zero code, then HACMP will assume a problem with the application.
- Monitor interval:This is the number of seconds that HACMP will wait for the monitor method to return. If the monitor does not respond in this time, HACMP will assume it hung.
- Hung monitor signal:The signal that HACMP will send the application monitor, if it hasn't return a response in the monitor interval. The default is a **`SIGKILL`**.

If there are multiple monitors for a particular application, HACMP will handle them as follows:

If a monitor with the fallover policy gets triggered first, then the fallover process will start and other monitors for this application will be disabled until the application is started again.

If a monitor with the notify policy gets triggered first, then the particular notify method will be triggered and the other monitors will continue unaffected.

Application availability

HACMP also provides the application availability analysis tool, which is useful for auditing the overall application availability, and for assessing the cluster environment.

2.4.6 Communication adapters and links

HACMP supports three types of communication links:

- SNA configured over a LAN interface
- SNA over X.25
- X.25

Because of the way that X.25 is used, these interfaces are treated as a different class of interfaces or devices and are not included in the HACMP topology, and therefore not managed by the usual methods. In particular heartbeats are not used to monitor the status of the X.25 interfaces. A new daemon **`clcommlinkd`**, which uses **`x25status`**, to monitor the X.25 link status.

2.4.7 Tape resources

Some SCSI and fibre channel connected tape drives can be configured as a highly available resource as part of any non-current behavior resource group.

2.4.8 Fast connect resources

The fast connect application server doesn't need start and stop scripts to be configured as they are already integrated into HACMP. Once fast connect is configured as an HACMP resource, then HACMP supports the start, stop, fallover, fallback and recovery of fast connect services. Fast connect services cannot be running on the cluster when the cluster is being brought up as HACMP needs to be controlling fast connect.

If IP address takeover and hardware address have been configured, clients will not need to re-establish their connection. HACMP allows WLM classes to be configured as part of a resource group, thus ensuring that applications have sufficient access to critical system resources during times of peak workload.

2.4.9 WLM integration

Workload manager is the AIX resource administration tool that allows targets and limits to be set for applications and users use of CPU time, physical memory usage and disk I/O bandwidth. WLM classes can be configured, each with a

range of system resources. Rules are then created to assign applications or groups of users to a class, and thus a range of resources that they can use.

WLM, using HACMP configuration, will start either when a node joins the cluster, or as the result of DARE involving WLM - and only on nodes part of resource groups containing WLM classes. HACMP works with WLM in two ways:

If WLM is already running, then HACMP will save the running configuration, stop WLM and then restart with the HACMP configuration files. When HACMP stops on a node, the previous WLM configuration will be activated.

If WLM is not running, it will start with the HACMP configuration, and stopped when HACMP stops on the node.

Attention: HACMP can only perform limited verification of the WLM configuration. Proper planning must be performed in advance.

The configuration that WLM uses on a node is specific to the node and the resource groups that may be brought online on that node. Workload manager classes can be assigned to resource groups either as:

- Primary class
- Secondary class

When a node is integrated into the cluster, HACMP will check each resource group that the node appears in the node list for. The WLM classes used will then depend on the startup policy of each resource group, and the nodes priority in the node list.

Primary WLM class

If the resource group is either online on home node only, or online on first available node:

- If the node is the highest priority node in the node list, the primary WLM class will be used.
- If the node is not the highest priority node and no secondary WLM class is defined, the node will use the primary WLM class.
- If the node is not the highest priority node and a secondary WLM class is defined, the node will use the secondary WLM class.

If the resource group has startup policy of either online on all available nodes (concurrent) or online using a node distribution policy, then the node will use the primary WLM class.

Secondary WLM class

This is optional and only used for nodes that are not the primary node for resource groups with a startup policy of either online on home node only, or online on first available node

2.4.10 Resource groups

Each resource must be included in a resource group to be made highly available by HACMP. Resource groups allow HACMP to manage a related group of resources as a single entity. For example an application may consist of start and stop scripts, a database, an IP address. These resources would then be included in a resource group for HACMP to control as a single entity.

HACMP ensures that resource groups remain highly available by moving them from node to node as conditions within the cluster change. The main states of the cluster and the associated resource group actions are:

- **Cluster startup**: The nodes in the cluster are up and then the resource groups are distributed according to their startup policy
- **Resource failure / recovery**: When a particular resource that is part of a resource group becomes unavailable, the resource group may be moved to another node. Similarly it may be moved back when the resource becomes available
- **HACMP shutdown on a node**: There are a number of ways of stopping HACMP on a node. One method will cause the node's resource groups to fallover to other nodes. Another method will take the resource groups offline. Under some circumstance it is possible to stop the cluster services on the node, while leaving the resources active.
- **Node failure / recovery**: If a node fails, the resource groups that were active on that node are distributed amongst the other nodes in the cluster, depending on their fallover distribution policies. When a node recovers and is re-integrated into the cluster, resource groups may be re-acquired depending on their fallover policies.
- **Cluster shutdown**: When the cluster is shutdown, all resource groups are taken offline. However there are some configurations were the resources can be left active, but the cluster resources stopped.

Before understanding the types of behavior and attributes that can be configured for resource groups, the following terms need to be understood:

- **Node list**: This is the list of nodes that is able to host a particular resource group. Each node must be able to access the resources that make up the resource group.

- **Default node priority**: This is the order in which the nodes are defined in the resource group. A resource group with default attributes will move from node to node in this order as each node fails.
- **Home node**: This is the highest priority node in the default node list. By default this is the node a resource group will initially be activated on. This does not specify the node that the resource group is currently active on.
- **Startup**: The process of bringing a resource group into an online state.
- **Fallover**: The process of moving a resource group that is online on one node to another node in the cluster in response to an event.
- **Fallback**: The process of moving a resource group that is currently online on a node that is not it's home node, to a re-integrating node.

Resource group behavior - policies and attributes

The behavior of resource groups is defined by configuring the resource group policies and behavior. Versions of HACMP prior to 5.1 supported three predefined resource groups:

- **Cascading**: Initially designed for resource groups to have an affinity for a particular node - will start on this node by preference, and fallback to it when it becomes available after a failure. Behavior could be further controlled by a combination of three attributes:
  - **Inactive takeover (ITO)**: The resource group would be brought online by nodes other than the priority node in the resource group's node list.
  - **Cascading without fallback (CWOF)**: Whether the resource group would cascade over to a higher priority node when it integrated into the cluster
  - **Dynamic Node Priority (DNP)**: Same as the attribute for customer resource groups.
- **Rotating**: Designed for resource groups to be spread across the nodes in the cluster. When a node starts, it will attempt to start any inactive rotating resource groups for which it is the highest priority node. Active rotating resource groups will not move when another node is integrated into the cluster.

  If there are multiple rotating resource groups in a cluster, the node preference is defined by the node order in the node list. Each node joining the cluster will acquire the rotating resource group for which it is the highest priority. If the number of resource groups exceeds the number of nodes, then the extra resource groups will not be brought online. Unless there are multiple HACMP networks defined, then each node will take one resource group for each network.
- **Concurrent**: Designed for concurrent applications - the resource group will be active on every node in the node list, just go offline on a particular node if

that node has a problem, and go online on a node that is integrated into the cluster.

Custom resource groups

Actually, what is important for HACMP implementors and administrators is the resource groups' behavior at startup, fallover and fallback. While in HACMP 5.1 both "custom" and "classic" RGs are supported, starting with 5.2, the only RGs available are the "custom" ones. The custom RG behavior options are:

Startup options

These options control the behavior of the resource group on initial startup.

- **Online on home node only**
  The resource group is brought online when it's home node joins the cluster. If the home node is not available, it will stay in an offline state until it is. See Figure 2-15.

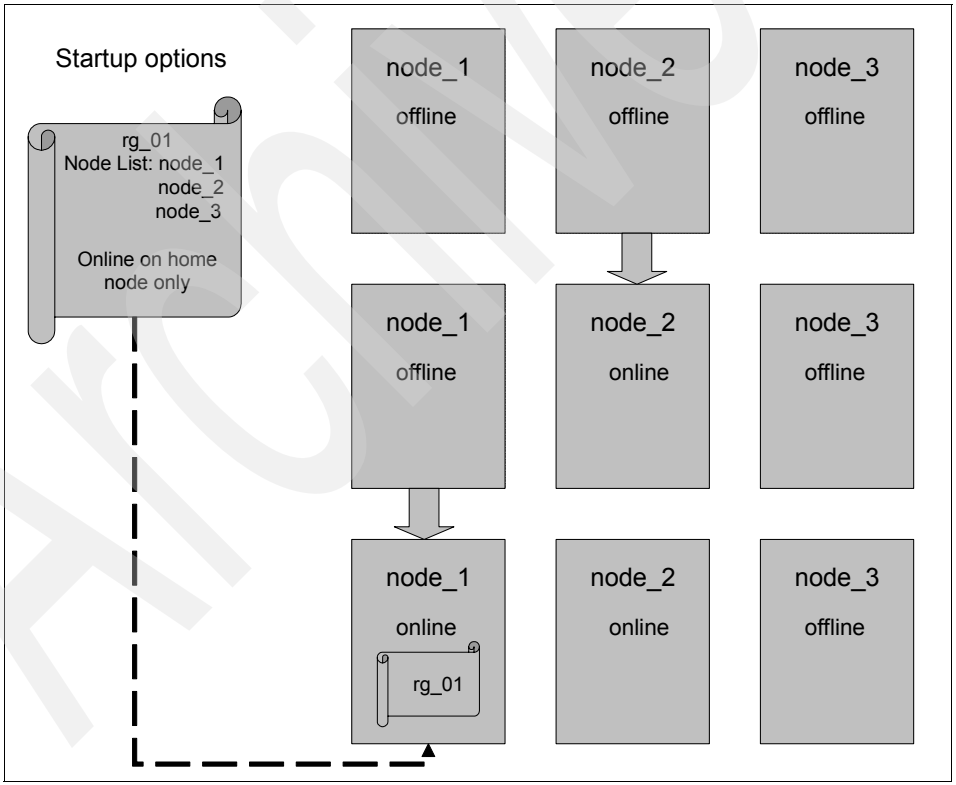

Figure 2-15 Online on home node only

- **Online on first available node**
  The resource group will be brought online when the first node in it's node list joins the cluster. See Figure 2-16.

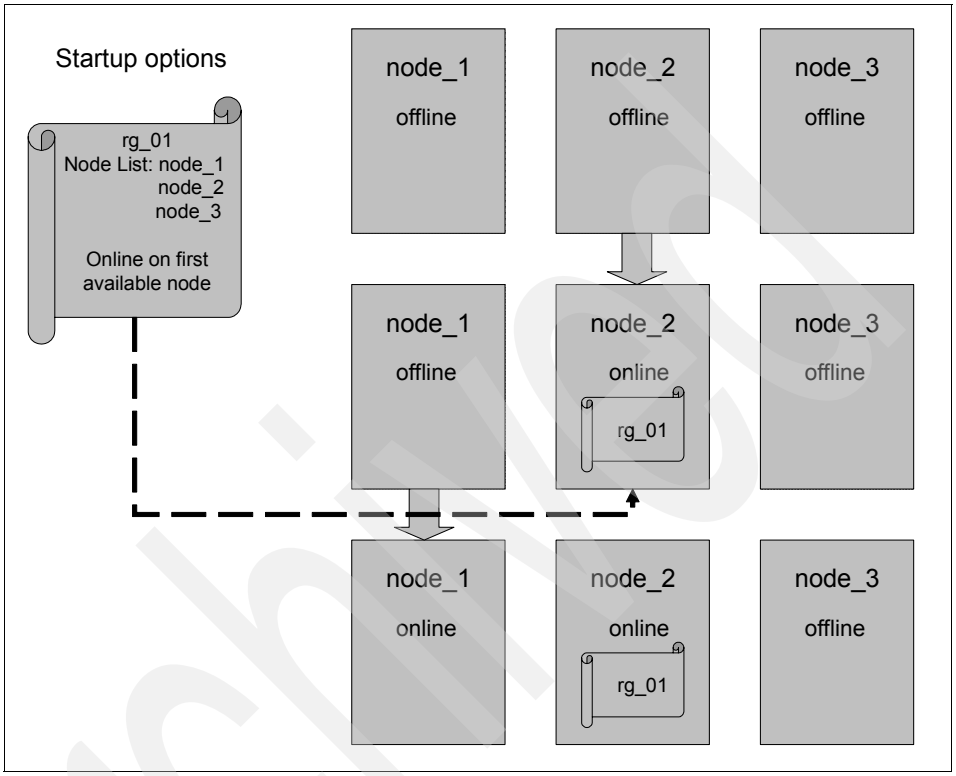

Figure 2-16 Online on first available node

- **Online on all available nodes**
  The resource group will be brought online on all nodes in it's node list as they join the cluster. See Figure 2-17 on page 95

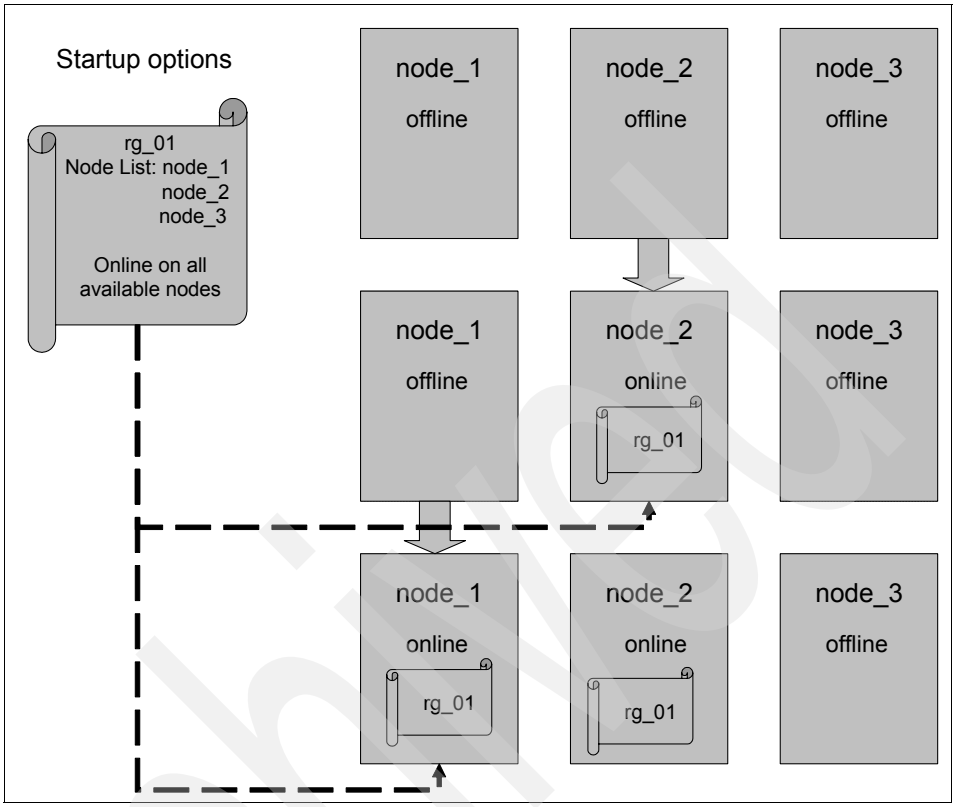

Figure 2-17 Online on all available nodes

- **Online using distribution policy**
  The resource group will only be brought online if the node has no other resource group of this type already online.
  If there is more than one resource group of this type when a node joins the cluster, HACMP will select the resource group with fewer nodes in it's node list. If that is the same, HACMP will choose the first node alphabetically. However if one node has a dependant resource group (that is it is a parent in a dependency relationship), it will be given preference. See Figure 2-18 on page 96.

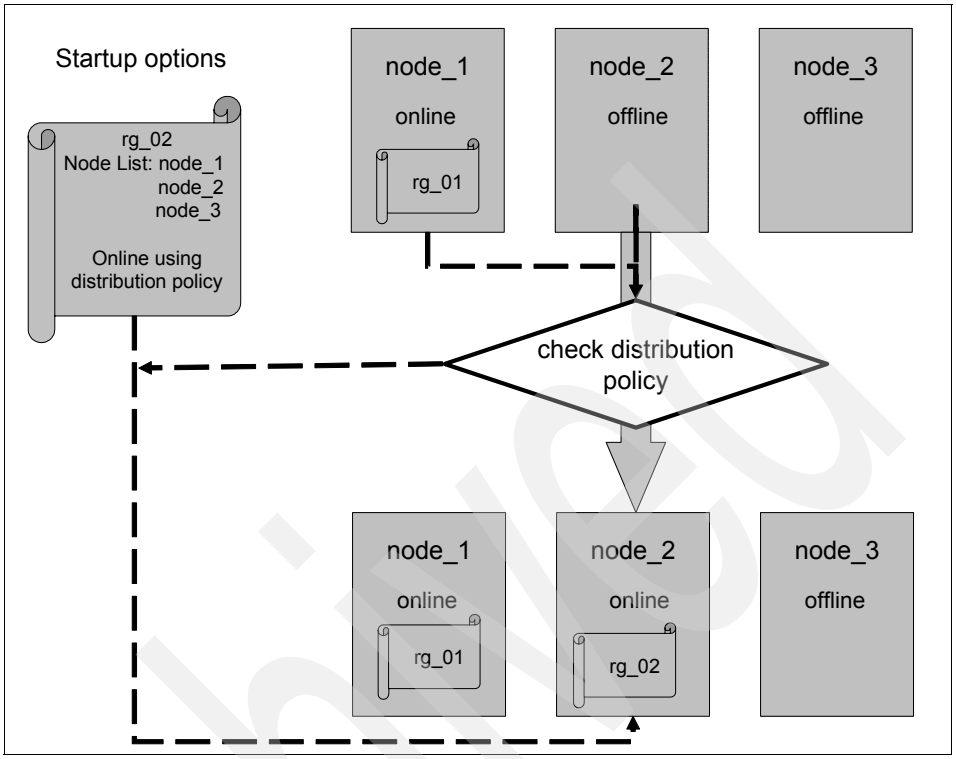

Figure 2-18 Online using distribution policy

Fallover options

These options control the behavior of the resource group should HACMP have to move it to another node in the response to an event.

- **Fallover to next priority node in list**
  The resource group will fallover to the next node in the resource groups node list. See Figure 2-19 on page 97.

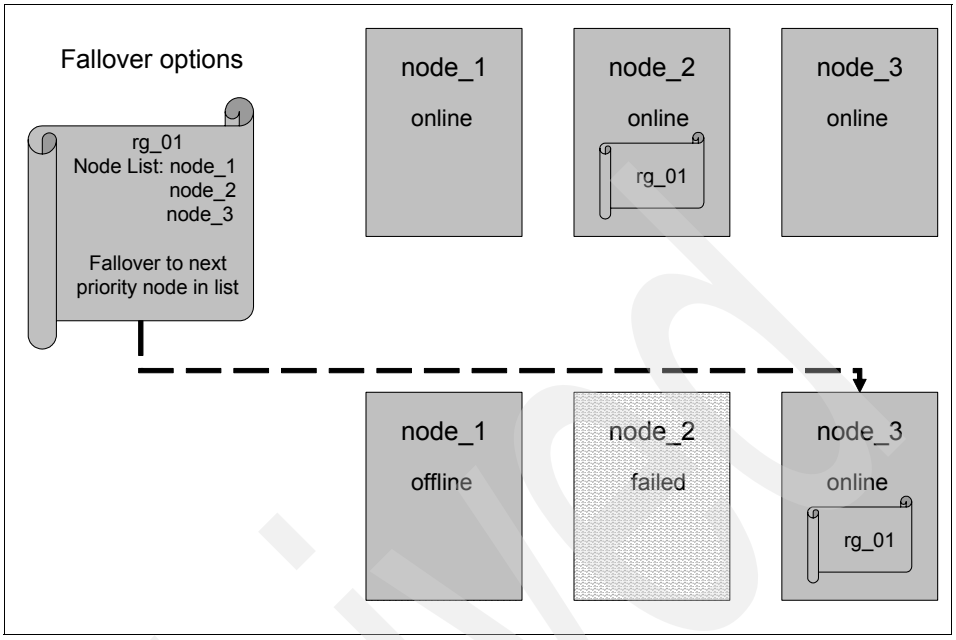

Figure 2-19 Fallover to next priority node in list

- **Fallover using dynamic node priority**
  The fallover node can be selected on the basis of either its available CPU, its available memory or the lowest disk usage. HACMP uses RSCT to gather the data for the selected variable from each of the nodes in the node list, then the resource group will fallover to the node that best meets the criteria. See Figure 2-20 on page 98.

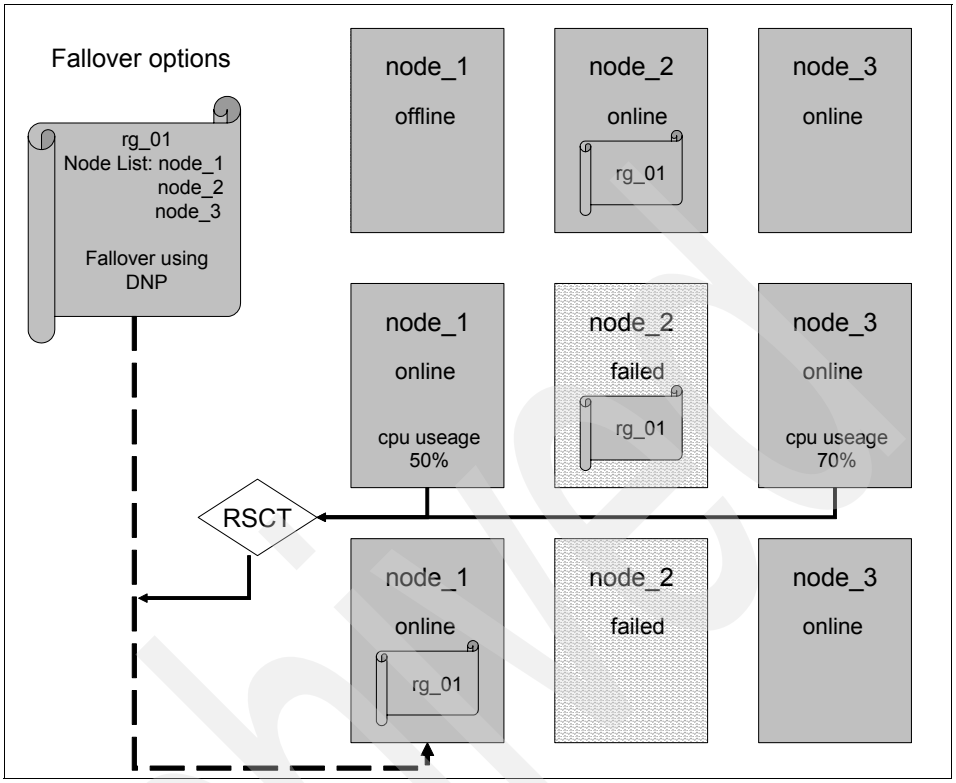

Figure 2-20 Fallover using dynamic node priority

- **Bring offline (on error only)**
  The resource group will be brought offline in the event of an error. This option is designed for resource groups that are online on all available nodes. See Figure 2-21 on page 99.

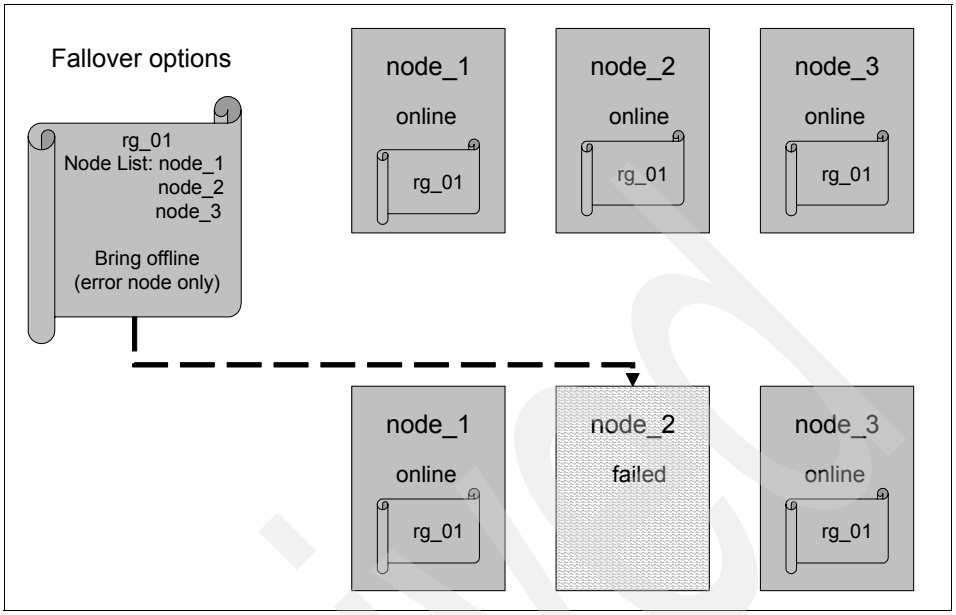

Figure 2-21 Bring offline (on error node only)

Fallback options

These options control the behavior of an online resource group when a node joins the cluster.

- **Fallback to higher priority node in list**
  The resource group will fallback to a higher priority node when it joins the cluster. See Figure 2-22 on page 100.

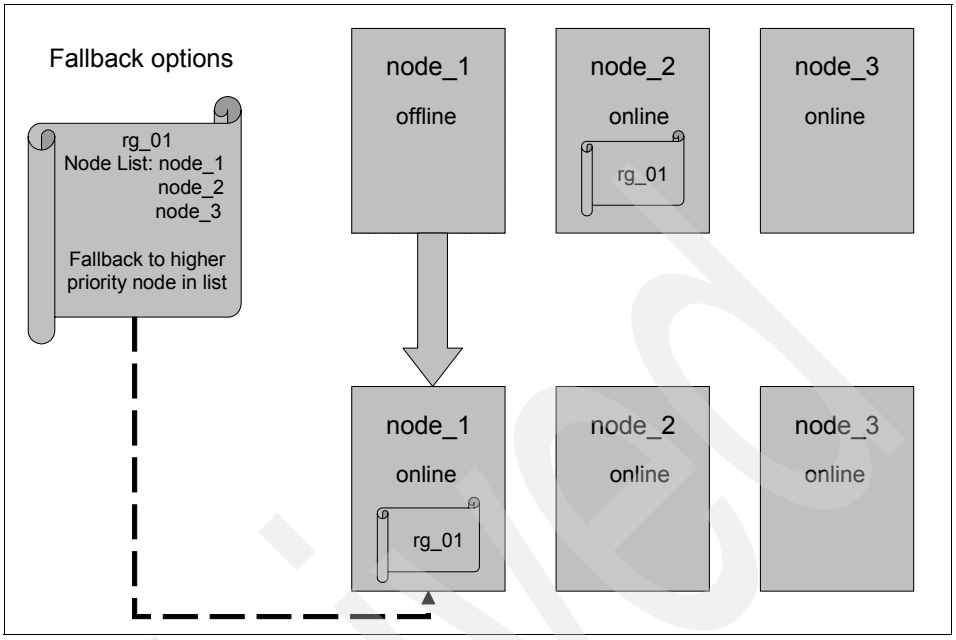

Figure 2-22 Fallback to higher priority node in list

- **Never fallback**
  The resource group will not move if a high priority node joins the cluster. Resource groups with online on all available nodes must be configured with this option. See Figure 2-23 on page 101.

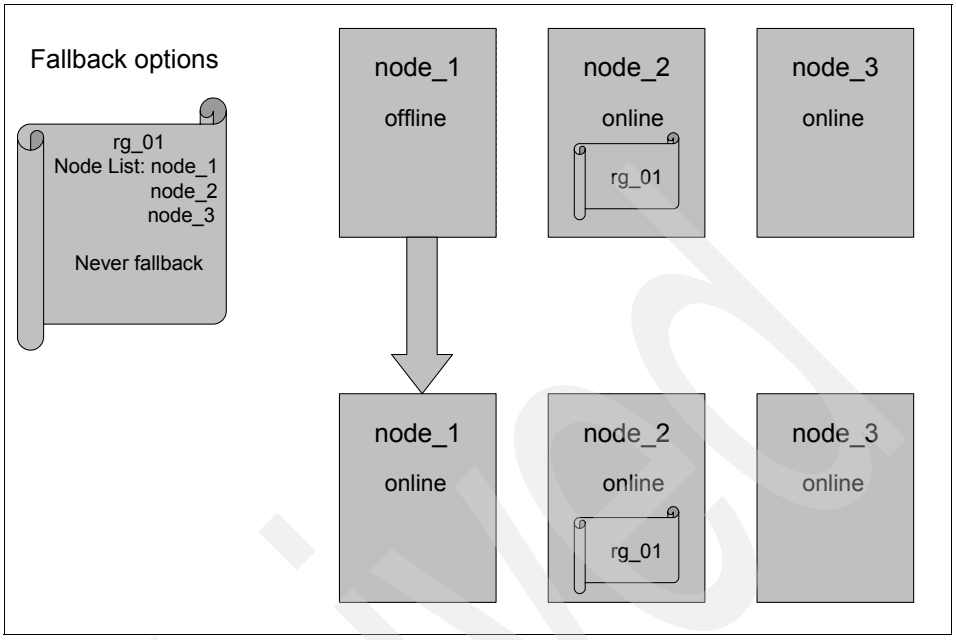

Figure 2-23 Never fallback

Table 2-4 shows how the startup, fallover and fallback options compare with the original cascading, rotating and concurrent resource groups.

Table 2-4 Resource group behavior compared

Old Configuration	Startup	Fallover	Fallback
Cascading ITO = false CWOF = false DNP = false	Online on home node only	Fallover to next priority node in the list	Fallback to higher priority node in the list
Cascading ITO = true CWOF = false DNP = false	Online on first available node	Fallover to next priority node in the list	Fallback to higher priority node in the list
Cascading ITO = false CWOF = true DNP = false	Online on home node only	Fallover to next priority node in the list	Never fallback

Old Configuration	Startup	Fallover	Fallback
Cascading ITO[a] = true CWOF[b] = true DNP[c] = false	Online on first available node	Fallover to next priority node in the list	Never fallback
Cascading ITO = false CWOF = false DNP = true	Online on home node only	Fallover using dynamic node priority	Fallback to higher priority node in the list
Cascading ITO = true CWOF = false DNP = true	Online on first available node	Fallover using dynamic node priority	Fallback to higher priority node in the list
Cascading ITO = false CWOF = true DNP = true	Online on home node only	Fallover using dynamic node priority	Never fallback
Cascading ITO = true CWOF = true DNP = true	Online on first available node	Fallover using dynamic node priority	Never fallback
Rotating	Online using distribution policy	Fallover to next priority node in the list	Never fallback
Concurrent	Online on all available nodes	Bring offline on error node only	Never fallback

a. ITO - Inactive take over
b. CWOF - Cascading without fallback
c. DNP - Dynamic Node Priority

Resource group attributes

Resource group behavior can now be further tuned by setting resource group attributes:

- Settling time
- Delayed fallback timers
- Distribution policy
- Dynamic node priorities
- Resource group processing order
- Priority override location
- Resource group dependencies - parent / child
- Resource group dependencies - location

Table 2-5 Resource group attributes and how they affect RG behavior

Attribute	Startup	Fallover	Fallback
Settling time	P		
Delayed fallback timer			P
Distribution policy	P		
Dynamic node priority		P	
Resource group processing order	P	P	P
Priority override location	P		P
Resource group Parent / Child dependency	P	P	P
Resource group location dependency	P	P	P

Settling time

This is a cluster wide attribute that affects the behavior of resource groups that have a startup policy of online on first available node. If not set, these resource groups will start on the first node in their resource group that integrates into the cluster. If the settling time is set for a resource group and the node that integrates into the cluster is it's highest priority node then it will come on line immediately, else it will wait the settling time to see if another higher priority node joins.

This is to ensure that a resource group does not start on an early integrated node low in it's priority list, then keep falling over to higher priority nodes as they integrate.

Delayed fallback timers

This is used to configure the time that a resource group will fallback. It can be set at a specific date and time, or a particular time, either daily, weekly, monthly or yearly.

The delayed fallback timer ensures that the resource group will fallback to it's highest priority node at a specific time. This is ensures that if there is to be a small disruption in services, it can occur at a time convenient to the users. The resource must not already be on it's highest priority node.

Distribution policy

This node based distribution policy ensures that on cluster startup, each node will only acquire one resource group with this policy set.
In HACMP 5.2 there was also a network based distribution policy that would ensure that there was only 1 resource group coming online on the same network

and node, so nodes with multiple networks could host multiple resource groups of this type.

Dynamic node priority

f there are three or more nodes in the cluster, a dynamic node priority fallover policy can be configured. One of the following three variables can be chosen to determine which node the resource group will fallover too:

- Highest free memory
- Highest idle CPU
- Lowest disk busy

The cluster manager keeps a table of these values for each node in the cluster. So at the time of fallover, the cluster manager can quickly determine which node best meets the criteria. These values are updated every 2 minutes, unless the node cannot be reached, in which case, the previous values remain unchanged.

Important: To measure highest free memory, HACMP will sample the paging space in use rather than the actual memory in use.

To display the current values, use **lssrc -ls clstrmgrES**, as shown:

Example 2-1 Checking DNP values as known to clstrmgrES

```
odin:/# lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#)36   1.135.1.37   src/43haes/usr/sbin/cluster/hacmprd/main.C,
hacmp.pe, 51haes_r530, r5300525a 6/20/05 14:13:01"
i_local_nodeid 1, i_local_siteid 1, my_handle 2
ml_idx[1]=0    ml_idx[2]=1    ml_idx[3]=2
There are 0 events on the Ibcast queue
There are 0 events on the RM Ibcast queue
CLversion: 8
Example 2-1cluster fix level is "0"
The following timer(s) are currently active:
Current DNP values
DNP Values for NodeId - 1  NodeName - frigg
    PgSpFree = 0  PvPctBusy = 0  PctTotalTimeIdle = 0.000000
DNP Values for NodeId - 2  NodeName - odin
    PgSpFree = 130258  PvPctBusy = 0  PctTotalTimeIdle = 99.325169
DNP Values for NodeId - 3  NodeName - thor
    PgSpFree = 0  PvPctBusy = 0  PctTotalTimeIdle = 0.000000
```

Resource group processing order

If a node is attempting to bring more than one resource group online, the default behavior is to merge all the resources into one large resource group and then

process them as one ‘resource group’. This is called parallel processing, though it is not true parallel processing as it is single thread.
This default behavior can be altered and a serial processing can be specified for particular resource groups by specifying a serial acquisition list. This order only defines the order of processing on a particular node, not across nodes. If serial processing is specified:

- The specified resource groups will be processed in order
- Resource groups containing only NFS mounts will be processed in parallel
- The remaining resource groups will be processed in order.
- The reverse order will be used on release.

Priority override location (POL)

When a resource group is moved to another node or site, taken offline or brought online by an administrator, then a priority override location (POL) is set for the node and resource group. Because this move goes against the defined behavior of the resource groups, a priority override location is set to stop the resource group immediately falling back to it’s appropriate node. This attribute replaces the “sticky attribute” from previous versions. Resource groups that are configured to be online on all available nodes, can only be brought online or offline on particular nodes and no POL will be set.

This POL can be set to be:

- **Persistent**: Will survive a restart of all cluster services on all nodes in the cluster.
- **Non-persistent**: Only remains in effect until cluster services is restarted on all nodes in the cluster. The resource group will return to it’s default behavior after a reboot of the cluster.

Note: If a resource group with a never fallback policy is moved, a POL will be set and the resource group will fallback to this node until the POL is cleared.

When moving an online resource group to another node, the following selection is offered:

- **Node list**: The node that the resource group is to be moved to. The priority override location will be set to this node
- **Restore_node_priority_order**: The resource group will be moved to it’s highest priority node and no priority override location will be set.

This information is kept in `/usr/es/sbin/cluster/etc/clpol` on each node in the cluster.

Resource group dependencies

A combination of two types of resource group dependency can be set:

- Parent / child dependency
- Location dependencies

Parent / child relationships between resource groups are designed for multi-tier applications, where one or more resource groups cannot successfully start till a particular resource group is already active. When a parent / child relationship is defined, the parent resource group must be online before any of it's children can be brought online on any node. If the parent resource group is to be taken offline, then the children must be taken offline first.
Up to three levels of dependency can be specified, that is a parent resource group can have children that are also parents to other resource groups. However circular dependencies are not allowed.

Figure 2-24 on page 107 shows an example where resource group 2 has two children, one of which also has a child. Thus resource group 2 must be online before resource groups 3 and 4 can be brought online. Similarly resource group 4 must be online before resource group 5 can be brought online. Resource group 3 has two parents (resource groups 1 and 2) that must be online before it can come online.

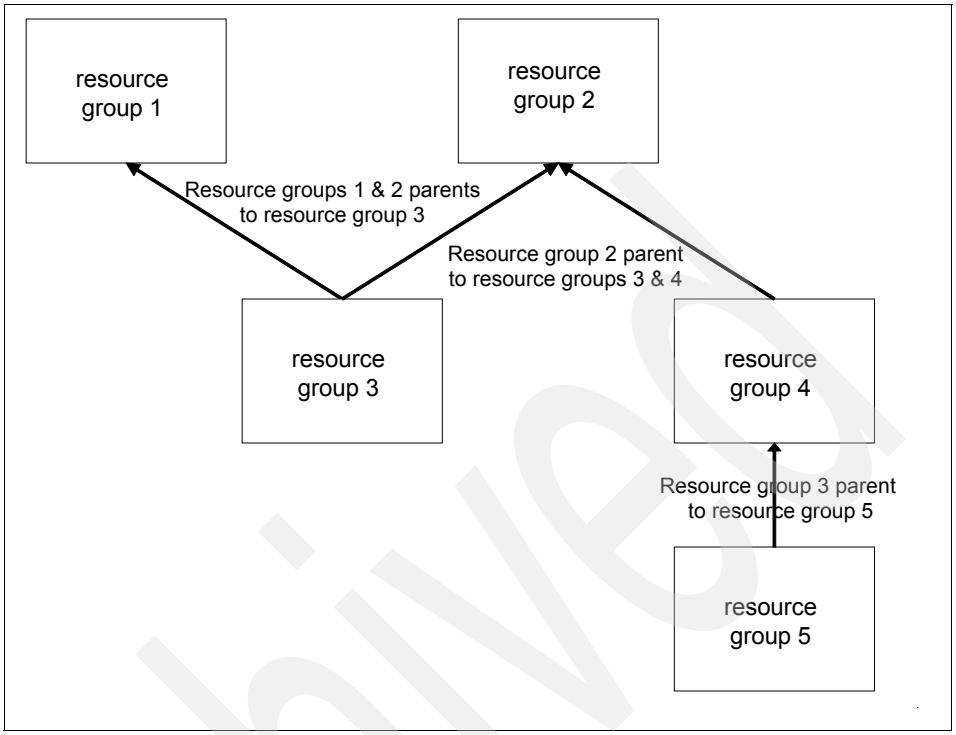

Figure 2-24 Parent / child resource group relationships

As HACMP starts applications in background (so a hang of the script will not stop HACMP processing), it is important to have startup application monitors for the parents in any parent / child resource group dependency. Once the startup application monitor, or monitors, have confirmed that the application has successfully started, the processing of the child resource groups can then commence.

Location dependencies can also be defined for resource groups in HACMP 5.3. The choices are:

- **Online on same node**: The specified resource groups will always startup, fallover and fall back to the same node, that is the nodes will move a set A resource group with this dependency can only be brought online on the node where other resource groups in the same set are already online, unless it is the first resource group in the set to be brought online.
- **Online on different nodes**: The specified resource groups will startup, fallover and fall back to different nodes. A priority is assigned to the resource groups, so that the higher priority nodes will handled first and kept in an online state should there be a limited number of nodes.

Low priority nodes will be taken off line on a node if a higher priority resource group is without a node. Intermediate priority nodes will not be taken offline. A resource group with this dependency can only be brought online on a node where there are no other resource groups that are part of this dependency already online.

- **Online on same site**: The specified resource groups will always be in an online state at the same site.
  A resource group with this dependency can only be brought online on the site where other resource groups with this dependency are currently in an online state, unless it is the first with the dependency to be brought online.

Figure 2-25 Shows and example of a three node cluster, with two databases and two applications. The applications cannot start up before the databases are online - so the parent / child dependency is configured. For performance reasons, the databases should be on different nodes, and the applications should be on the same nodes as the applications, so the location dependencies are configured.

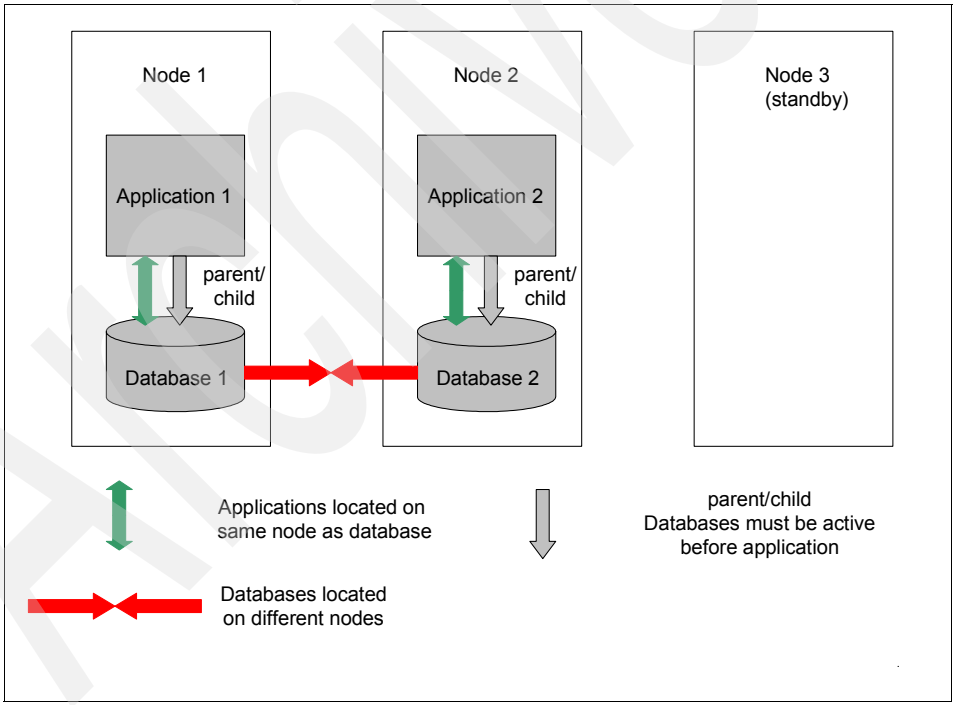

Figure 2-25 Resource group dependencies

To set or display the RG dependencies, you can use the **clrgdependency** command, as shown in Example 2-2 on page 109:

Example 2-2 Modifying and checking RG dependencies

```
odin:>#  clrgdepdency -t [PARENT_CHILD | NODECOLLOCATION | ANTICOLLOCATION |
SITECOLLOCATION ] -sl
odin:>#  clrgdepdency -t PARENT_CHILD -sl
#Parent                Child
rg1                    rg2
rg1                    rg3

odin:>#  clrgdepdency -t NODECOLLOCATION -sl

odin:>#  clrgdepdency -t ANTILOCATION -sl
#HIGH:INTERMEDIATE:LOW
rg01::rg03frigg

odin:>#  clrgdepdency -t SITECOLLOCATION -sl
rg01 rg03 frigg
```

Another way to check is by using the **odmget HACMPrg_loc_dependency** command.

Resource group manipulation

Resource groups can be:

- **Brought online**: A resource group can be brought online on a node in the resource group's node list. The resource group would be currently offline, unless it was an online on all available nodes resource group.
- **Brought offline**: A resource group can be taken offline from a particular node.
- **Moved to another node while online**: A resource group that is online on one node can be taken offline and then brought online on another node in the resource group's node list. This may include moving the resource group to another site.

A priority override location will be set as previously discussed.

Certain changes will not allowed:

- A parent resource group may not be taken offline or moved if there is a child resource group in an online state.
- A child resource group cannot be started until the parent resource group is online.

Resource groups states

HACMP 5x has modified how resource group failures are handled, and as such manual intervention is not always required.

If a node fails to bring a resource group online when it joins the cluster, the resource group will be left in ERROR state. If this fails and the resource group is not configured as online on all available nodes, then HACMP will attempt to bring the resource group online on the other active nodes in the resource group's node list.

Since HACMP 5.2, each node that joins the cluster will automatically attempt to bring online any of the resource groups that are in the ERROR state.

If a node fails to acquire a resource group during fallover, the resource group will be marked as "recoverable" and HACMP will attempt to bring the resource group online in all the nodes in the resource groups node list. If this fails for all nodes, then the resource group will be left in ERROR state

If there is a failure of a network on a particular node, HACMP will determine what resource groups are affected (those that had service IP labels on the network) and them attempt to bring them online on another node. If there are node nodes with the required network resources, then the resource groups are left in ERROR state. Should any interfaces now become available, HACMP will work out what ERROR resource groups can be brought on line, then attempt to do so.

Tip: If you want to override the automatic behavior of bringing a resource group in ERROR state online, specify that it must remain offline on a node and thereby set a POL.

Selective fallovers

- Interface failure
  - HACMP will swap interfaces if possible, else will move RG to highest priority node with an available interface, and if not successful, RG will be brought in ERROR state.
- Network failure
  - Local - move affected RGs to another node
  - Global - node_down for all nodes
- Application failure
  - If an application monitor indicates an application has failed, depending on the configuration, HACMP will attempt to first restart the application on the same node (usually three times), then, if this is not possible, HACMP will move the RG to another node, and if this fails too, the RG is brought in ERROR state.
- Communication link failure
  - HACMP will attempt to move the RG to another node

– If selective fallover for VG if “LVM_SA_QUORCLOSE” error is configured, the HACMP will attempt to move the affected RGs to another node.

2.5 HACMP plug-ins

The HACMP plug-in software contains example scripts to help you configure the following services as part of a highly available cluster:

- Name server
- Print server
- DHCP server

Each plug-in consists of application start and stop scripts, application monitor scripts and cleanup scripts. There is also a script to confirm that the correct configuration files are available on a shared file system. Each plug-in contains a README file with further information. They are found under /usr/es/sbin/cluster/plugin/<plugin_name>.

2.6 Features (HACMP 5.1, 5.2 and 5.3)

This section lists some new features and enhancements and those no longer supported.

2.6.1 New features

Clinfo daemon intra-cluster communication enhancements

The clinfo daemon now contains version information. and has a new logfile **/tmp/clinfo.debug**.

The SMUX peer daemon (clsmuxpd) functionality has been added into the cluster manager daemon so SNMP queries are possible even if the cluster is not active. Two new states not_configured and not_synced have been created.

The cluster manager now has two log files:

/tmp/clstrmgr.debug	The location is configurable and the file contains the cluster manager default logging
/tmp/clsmuxtrmgr.debug	This is a new logfile for tracing the new SNMP function of the cluster manager.

Cluster verification enhancements

Automatic verification and synchronization

HACMP verifies the nodes configuration when it starts (either as the first node in the cluster, or joining an active cluster). The following are automatically checked and corrected if required:

- RSCT instance numbers consistent.
- IP interfaces configured as RSCT expects.
- Shared volume groups are not set to automatically varyon.
- File systems are not set to automatically mount.

If the joining nodes configuration doesn't match that of the running cluster, it will be synchronized with one of the running nodes.

Verification also detects potential single points of failure that were previously only detected by automatic error notification.

Additional cluster verification

Additional checks are performed by HACMP:

- Each node has same version of RSCT.
- Each IP interface has same setting for MTU, and the AIX and HACMP settings for the IP interfaces are consistent.
- The AIX network options used by HACMP and RSCT are consistent across nodes.
- Volume groups and PVIDs are consistent across nodes that are members of the owning resource group.
- If HACMP/XD installed, site management policy cannot be set to ignore.
- If HACMP/XD, then the GeoRM, PPRC or GLVM configuration will be verified.

clhosts file automatically populated

The clhosts file which is used by many of the monitor programmes, has two forms:

- **HACMP server version**: This file is on each node in /usr/es/sbin/cluster/etc and has the entry 127.0.0.1 added on installation of HACMP.
- **HACMP client version**: The file clhosts.client is found in /usr/es/sbin/cluster/etc and is populated by cluster verification with each the address and label of each interface and defined service IP address. Date stamped versions are kept.

Cluster definition file in XML format

XML file format is the common format for user created cluster definition files and the online planning worksheet files. SMIT can be used to convert existing cluster snapshot files into a XML cluster definition file

OEM and Veritas Volumes and filesystem integration

HACMP can now routinely manage OEM volume groups and their corresponding file systems. This feature means that OEM disks, volumes, and file systems can be included in an HACMP resource group. Either the supplied custom methods can be used, or individually customized methods.

In particular HACMP will automatically detect volume groups created with Veritas volume manager using the Veritas foundation suite (v4.0).

SMS capability

A new custom remote notification method has been added. It is now possible to send remote notification messages to any address such as a mobile phone or as an E-mail to an E-mail address.

Resource group location dependencies

In addition to the policies that define resource group parent / child dependencies and the startup distribution, HACMP now offers cluster wide location dependencies for resource groups:

- Online on same node
- Online on different nodes
- Online on same site

Note: The startup distribution policy is only node based, HACMP 5.2 supported the choice of node or network based.

Service IP label / address distribution preference

By default HACMP distributes the service IP labels / addresses across the available interfaces. When a service IP label is to be activated, HACMP will calculate the number of alias addresses currently on each interface, and then use the interface with the least aliases.

- **Anti-Collocation**: This is the default and HACMP will distribute the service IP labels across all the boot IP interfaces in the same HACMP network on the node.
- **Collocation**: HACMP will allocate all service IP addresses on the same boot IP interface.

- **Collocation with persistent label**: HACMP will allocate all service IP addresses on the boot IP interface that is hosting the persistent alias IP label. This may be useful in environments with VPN and firewall configuration, where only one interface is granted external connectivity.
- **Anti-Collocation with persistent label**: HACMP will distribute all the service IP labels across all the boot IP interfaces in the same logical network, that are not hosting the persistent alias IP label. If no other interfaces are available, the service IP labels will share the adapter with the persistent alias IP label.

HACMP/XD

Parallel processing of the primary and secondary instances of HACMP/XD replicated resource groups is the default, though serial processing can be specified for this release at least. DARE and rg_move processes support parallel processing across sites.

Site management policies can be specified for the startup, fallover and fallback behavior of both the primary and secondary instance of a resource group.

- Concurrent like (online on all available nodes) resource group inter site behavior can be combined with a non-concurrent site policy.
- Parent / child dependency relationships can be specified.
- Node based distribution start policy can be used.
- Resource group collocation and Anti-collocation supported.
- Cluster verification also verifies HACMP/XD configurations, however the configuration must be manually propagated to other nodes as often significant amounts of customization must be done.

WebSMIT security enhancements

There is server side validation of parameters passed to WebSMIT prior to execution.

The WebSMIT authentication tools are more fully integrated with AIX authentication mechanisms

HACMP smart assist programs

- HACMP now supports - Smart assist for WebSphere - although supported in 5.2, has been updated.
- Smart assist for DB2 - includes monitoring and recovery support for DB2 Universal Database™ Enterprise Server Edition.
- Smart assist for Oracle - assists with the install of the Oracle application server 10g.

2.6.2 Features no longer supported

- cllockd or cllockdES no longer supported.
- clinfo no longer uses shared memory - uses message queues.
- clsmuxpd is not longer supported and functions rolled into the cluster manager.
- Cascading, rotating and concurrent resource groups no longer supported.
- Event management subsystem has been replaced with RSCT Resource Monitoring and Control (RMC) subsystem.
- cldiag is no longer supported from the command line.
- clverify is no longer supported from the command line.

2.7 Limitations

This section lists some of the common limits HACMP is subject to. These limits are presented in Table 2-6:

Table 2-6 HACMP limits

Component	Maximum number supported in a cluster
Nodes	32
Resource groups	64
Networks	48
Network interfaces, devices and labels	256
Cluster resources	While 128 is the maximum clinfo can handle, there can be more in the cluster.
Parent-Child dependencies	max of 3 levels
Sites	2
Interfaces	7 interfaces per node per network
Application monitors per site	128
netmon.cf	30 names or IP addresses
Persistent IP alias	one per node per network
XD_networks	1 per cluster

Component	Maximum number supported in a cluster
GLVM Modes	Synchronous, non concurrent
GLVM Devices	All PVs supported by AIX, no need to be same local and remote, Enhanced concurrent mode not supported.
Dynamic reconfiguration	Not possible in HACMP/XD:HAGEO and HACMP/XD:GLVM

Subnet requirements

The AIX 5L kernel routing table supports multiple routes for the same destination. If multiple matching routes have the same weight, each of the subnet routes will be used alternately. The problem that this poses for HACMP is that if one node has multiple interfaces that shares the same route, then HACMP has no means to determine its health.

Therefore we recommend that each interface on a node belongs to a unique subnet, so that each interface can be monitored. Using heartbeat over alias is an alternative.

2.8 Storage considerations

This section presents some of the most common storage subsystems and associated management software and characteristics, along with HACMP storage handling capabilities.

IBM DS4xxx series storage subsystems

These devices were previously known under the name of “Fiber Attach Storage Server (FAStT)”. There are different models of DS4xxx storage subsystems supported in HACMP. Covering all models is not within the scope of this book.

To understand how to configure the DS4xxx storage, we present an example of the DS4500 Storage Server.

DS4500 Storage Server

The DS4500 Storage Server supports direct attachment of up to four hosts that contain two host adapters each, and is designed to provide maximum host-side and drive-side redundancy. By using external Fibre Channel switches in conjunction with the DS4500 Storage Server, you can attach up to 64 hosts (each with two host bus adapters) to a DS4500 Storage Server.

Before configuring the DS4500 storage, you must make sure all hardware and cabling connection is done, as per the required configuration. For more information about DS4500 cabling, see *IBM TotalStorage DS4500 Fibre Channel Storage Server Installation Guide*, GC26-7530.

DS4xxx Storage Manager software

The only way to configure DS4500 Storage is to use the DS4xxx Storage Manager software. The DS4xxx Storage Manager software is available on most popular operating systems, such as AIX, Linux, and Windows® XP/2000. With DS4xxx Storage Manager, you can configure supported RAID levels, logical drives, and partitions. Supported RAID levels are RAID 0, RAID 1, RAID 5, and RAID 0+1.

There is no option to configure RAID 10 in DS4xxx Storage Manager. Selecting RAID 1 with multiple disks, DS4xxx Manager takes care of striping and mirroring of the data.

It allows a user to format the logical drives as required by the host operating systems. There are different versions of Storage Manager.

Some of the new features supported by DS4xxx with Storage Manager are:

- FlashCopy®

  A FlashCopy logical drive is a logical point-in-time image of another logical drive, called a base logical drive, that is in the storage subsystem. A FlashCopy is the logical equivalent of a complete physical copy, but you create it much more quickly and it requires less disk space (20% of the original logical drive).

- Remote mirror option

  The remote mirror option is used for online, real-time replication of data between storage subsystems over a remote distance.

- Volumecopy

  The *volumecopy* option is a firmware-based mechanism for replicating logical drives data within a storage array. Users submit *volumecopy* requests by specifying two compatible drives. One drive is designated as the source and the other as a target. The *volumecopy* request is persistent so that any relevant result of the copy process can be communicated to the user.

- Storage partitioning

  Storage partitioning allows the user to present all storage volumes to a SAN through several different partitions by mapping storage volumes to a LUN number, each partition presenting LUNs 0-255. This volume or LUNs mapping applies only to the host port or ports that have been configured to access that LUN. This feature also allows the support of multiple hosts using

different operating systems and their own unique disk storage subsystems settings to be connected to the same DS4xxx storage server at the same time.

Enterprise Storage Server (ESS / Shark)

The IBM Enterprise Storage Server (ESS) is a second-generation Seascape® disk storage system that provides industry-leading availability, performance, manageability, and scalability. RAID levels in ESS are predefined in certain configurations and have limited modification capabilities. Available RAID levels are RAID 1, RAID 5, and RAID 0+1.

The IBM Enterprise Storage Server (ESS) does more than simply enable shared storage across enterprise platforms; it can improve the performance, availability, scalability, and manageability of enterprise-wide storage resources through a variety of powerful features. Some of the features are similar in name to those in available FAStT/DS4xxx Storage, but the technical concepts differ to a great extent. Some of those features are:

- FlashCopy

  FlashCopy provides fast data duplication capability. This option helps eliminate the need to stop applications for extended periods of time in order to perform backups and restores.

- Peer-to-peer remote copy

  This feature maintains a synchronous copy (always up-to-date with the primary copy) of data in a remote location. This backup copy of data can be used to quickly recover from a failure in the primary system without losing any transactions; this is an optional capability that can literally keep your e-business applications running.

- Extended remote copy (XRC)

  This feature provides a copy of data at a remote location (which can be connected using telecommunications lines at unlimited distances) to be used in case the primary storage system fails. The ESS enhances XRC with full support for unplanned outages. In the event of a telecommunications link failure, this optional function enables the secondary remote copy to be resynchronized quickly without requiring duplication of all data from the primary location for full disaster recovery protection.

- Custom volumes

  Custom volumes enable volumes of various sizes to be defined for high-end servers, enabling administrators to configure systems for optimal performance.

- Storage partitioning

  Storage partitioning uses storage devices more efficiently by providing each server access to its own pool of storage capacity. Storage pools can be shared among multiple servers.

  For more information about the configuration of the Enterprise Storage Server, refer to the *IBM TotalStorage Enterprise Storage Server Service Guide 2105 Model 750/800 and Expansion Enclosure, Volume 1*, SY27-7635.

IBM TotalStorage DS6000 and DS8000 series

These new storage subsystems are supported with HACMP, but at the time of writing this redbook, we did not have enough information available, nor the hardware to test.

For the latest support matrix and information about new DS6000 and DS8000 series, see:

`http://www-1.ibm.com/servers/eserver/pseries/ha/`

Serial Storage Architecture (SSA)

Serial storage architecture is an industry-standard interface that provides high-performance fault-tolerant attachment of I/O storage devices. In SSA subsystems, transmissions to several destinations are multiplexed; the effective bandwidth is further increased by spatial reuse of the individual links. Commands are forwarded automatically from device to device along a loop until the target device is reached. Multiple commands can be travelling around the loop simultaneously.

SSA supports RAID 0, RAID 1, RAID 5, and RAID 0+1. To use any of the RAID setups, it is necessary to follow the looping instruction of SSA enclosures. Specific looping across the disk is required to create RAID. For more information about IBM SSA RAID Configuration, refer to *IBM Advanced SerialRAID Adapters Installation Guide*, SA33-3287.

2.8.1 Shared LVM

For a HACMP cluster, the key element is the data used by the highly available applications. This data is stored on AIX Logical Volume Manager (LVM) entities. HACMP clusters use the capabilities of the LVM to make this data accessible to multiple nodes. AIX Logical Volume Manager provides shared data access from multiple nodes. Some of the components of shared logical volume manager are:

- A *shared volume group* is a volume group that resides entirely on the external disks shared by cluster nodes.
- A *shared physical volume* is a disk that resides in a shared volume group.

- A *shared logical volume* is a logical volume that resides entirely in a shared volume group.
- A *shared file system* is a file system that resides entirely in a shared logical volume.

If you are an system administrator of an HACMP cluster, you may be called upon to perform any of the following LVM-related tasks:

- Create a new shared volume group.
- Extend, reduce, change, or remove an existing volume group.
- Create a new shared logical volume.
- Extend, reduce, change, or remove an existing logical volume.
- Create a new shared file system.
- Extend, change, or remove an existing file system.
- Add and remove physical volumes.

When performing any of these maintenance tasks on shared LVM components, make sure that ownership and permissions are reset when a volume group is exported and then re-imported.

After exporting and importing, a volume group is owned by root and accessible by the system group.

Note: Applications, such as some database servers, that use raw logical volumes may be affected by this change if they change the ownership of the raw logical volume device. You must restore the ownership and permissions back to what is needed after this sequence.

Shared logical volume access can be made available in any of the following data accessing modes:

- Non-concurrent access mode
- Concurrent access mode
- Enhanced concurrent access mode

2.8.2 Non-concurrent access mode

HACMP in a non-concurrent access environment typically uses journaled file systems to manage data, though some database applications may bypass the journaled file system and access the logical volume directly.

Both mirrored and non-mirrored configuration is supported by non-concurrent access of LVM. For more information about creating mirrored and non-mirrored

logical volumes, refer to the *HACMP for AIX 5L V5.3 Planning and Installation Guide*, SC23-4861-06.

To create a non-concurrent shared volume group on a node, perform the following steps:

1. Use the fast path **`smitty mkvg`**.
2. Use the default field values unless your site has other specific requirements.
   - VOLUME GROUP name

     The name of the shared volume group should be unique within the cluster.
   - Activate volume group AUTOMATICALLY at system restart?

     Set to No so that the volume group can be activated as appropriate by the cluster event scripts.
   - ACTIVATE volume group after it is created?

     Set to Yes.
   - Volume Group MAJOR NUMBER

     Make sure to use the same major number on all nodes. Use the **`lvlstmajor`** command on each node to determine a free major number common to all nodes.

To create a non-concurrent shared file system on a node, perform the following steps:

1. Use the fast path **`smitty crjfs`**.
2. Rename both the logical volume and the log logical volume for the file system and volume group.

   AIX assigns a logical volume name to each logical volume it creates. Examples of logical volume names are /dev/lv00 and /dev/lv01. Within an HACMP cluster, the name of any shared logical volume must be unique. Also, the journaled file system log (jfslog) is a logical volume that requires a unique name in the cluster.
3. Review the settings for the following fields:
   - Mount automatically at system restart?

     Make sure this field is set to No.
   - Start Disk Accounting

     Set this field to No unless you want disk accounting.
4. Test the newly created file system by mounting and unmounting it.

Importing a volume group to a fall-over node

Before you import the volume group, make sure the volume group is varied off from the primary node. You can then run the discovery process of HACMP, which will collect the information about all volume groups available across all nodes.

Importing the volume group onto the fall-over node synchronizes the ODM definition of the volume group on each node on which it is imported.

When adding a volume group to the resource group, you may choose to manually import a volume group onto the fall-over node or you may choose to automatically import it onto all the fall-over node in the resource group.

For more information about importing volume groups, see the *HACMP for AIX 5L V5.3 Planning and Installation Guide*, SC23-4861-06.

Note: After importing a volume group on the fall-over node, it is necessary to change the volume group startup status. Run following command to change the volume group status, as required by HACMP:

```
# chvg -an -Qn <vgname>
```

This will disable automatic varyon when the system restarts and also disable the quorum of the volume group.

2.8.3 Concurrent access mode

Using concurrent access with HACMP requires installing an additional fileset. Concurrent access mode is not supported for file systems; instead, you must use raw logical volumes or physical disks.

Creating a concurrent access volume group

1. The physical volumes (hdisk*) should be installed, configured, and available. You can verify the disks' status using the following command:

   ```
   # lsdev -Cc disk
   ```

2. To use a concurrent access volume group, you must create it as a concurrent capable volume group. A concurrent capable volume group can be activated (varied on) in either non-concurrent mode or concurrent access mode.

   To create a concurrent access volume group, do the following steps:

   a. Enter **`smit cl_convg`**.

   b. Select **Create a Concurrent Volume Group**.

   c. Enter the field values as desired.

d. Press Enter.

Import the concurrent capable volume group

Importing the concurrent capable volume group is done by running the following command:

```
# importvg -C -y vg_name physical_volume_name
```

Specify the name of any disk in the volume group as an argument to the **importvg** command. By default, AIX automatically varies on non-concurrent capable volume groups when they are imported. AIX does not automatically varyon concurrent capable volume groups when they are imported.

Varyon the concurrent capable VGs in non-concurrent mode

It is necessary to varyon the concurrent capable volume group in a non-concurrent mode to create logical volume. Use the **varyonvg** command to activate a volume group in non-concurrent mode:

```
# varyonvg <vgname>
```

Create logical volumes on the concurrent capable volume group

You can create logical volumes on the volume group, specifying the logical volume mirrors to provide data redundancy.

To create logical volumes on a concurrent capable volume group on a source node, perform the following steps:

1. Use the SMIT fast path **smit cl_conlv**.
2. Specify the size of the logical volume as the number of logical partitions.
3. Specify the desired values to the other available option.
4. Press Enter.

Varyoff a volume group

After creating the logical volume, varyoff the volume group using the **varyoffvg** command so that it can be varied on by the HACMP scripts. Enter:

```
# varyoffvg <vgname>
```

Define a concurrent volume group in an HACMP resource group

To start the concurrent volume group simultaneously on all the nodes, specify the volume group name in the startup scripts of HACMP.

On cluster startup, you may find the concurrent volume group is activated on all the configured nodes.

2.8.4 Enhanced concurrent mode (ECM) VGs

With HACMP V5.1, you now have the ability to create and use enhanced concurrent VGs. These can be used for both concurrent and non-concurrent access. You can also convert existing concurrent (classic) volume groups to enhanced concurrent mode using C-SPOC.

For enhanced concurrent volume groups that are used in a non-concurrent environment, rather than using the SCSI reservation mechanism, HACMP V5.1 uses the fast disk takeover mechanism to ensure fast takeover and data integrity.

Note: Fast disk takeover in HACMP V5.1 is available only in AIX 5L V5.2.

The ECM volume group is varied on all nodes in the cluster that are part of that resource group. However, the access for modifying data is only granted to the node that has the resource group active (online).

Active and passive varyon in ECM

An enhanced concurrent volume group can be made active on the node, or varied on, in two modes: active or passive.

Active varyon

In the active state, all high level operations are permitted. When an enhanced concurrent volume group is varied on in the 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 the passive state, the LVM provides the equivalent of fencing for the volume group at the LVM level. The node that has the VG varied on in passive mode is allowed 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 4 KB 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 the passive state:

- Operations on file systems, such mount

- Any open or write operation on logical volumes
- Synchronizing volume groups

Creating a enhanced concurrent access volume group

1. When concurrent volume groups are created on AIX 5L 5.1 and later, they are automatically created in enhanced concurrent mode.
2. To create a concurrent capable volume group from the AIX command line, use the **mkvg** command. For example:

   ```
   # mkvg -n -s 32 -C -y myvg hdisk11 hdisk12
   ```

   This will create an enhanced concurrent VG on hdisk11 and hdisk12. The flags do the following:

   -n Do not vary on VG at boot.

   -s 32 Gives a partition size of 32 MB.

   -C Creates an enhanced concurrent VG.

   -y Specifies the VG name.

2.8.5 Fast disk takeover

This is a new feature of HACMP V5.1, which has the following main purposes:

- Decreases the application downtime, with faster resource group fallover (and movement)
- Concurrent access to a volume group (preserving the data integrity)
- Uses AIX Enhanced Concurrent VGs (ECM)
- Uses RSCT for communications

The enhanced concurrent volume group supports active and passive mode varyon, and can be included in an non-concurrent resource group.

The fast disk takeover is set up automatically by the HACMP software. For all shared volume groups that have been created in enhanced concurrent mode and contain file systems, HACMP will activate the fast disk takeover feature. When HACMP starts, all nodes in a Resource Group that share the same enhanced Volume Group will varyon that Volume Group in passive mode. When the Resource Group is brought online, the node that acquires the resources will varyon the Volume Group in active mode.

The other nodes will maintain the Volume Group varied on in passive mode. In this case, all the changes to the Volume Group will be propagated automatically to all the nodes in that Volume Group. The change from active to passive mode

and the reverse are coordinated by HACMP at cluster startup, Resource Group activation and fallover, and when a failing node rejoins the cluster.

The prerequisites for this functionality are:

- HACMP V5.1
- AIX 5L 5.2 or higher
- bos.clvm.5.2.0.11 or higher
- APAR IY44237

For more information about fast disk takeover, see the *HACMP for AIX 5L V5.3 Planning and Installation Guide*, SC23-4861-06.

2.9 Shared storage configuration

Most of the HACMP configurations require shared storage. The IBM disk subsystems that support access from multiple hosts include SCSI, SSA, ESS, and FAStT.

There are also third-party (OEM) storage devices and subsystems that may be used, although most of these are not directly certified by IBM for HACMP usage. For these devices, check the manufacturer's respective Web sites.

Table 2-7 lists a subset of IBM storage devices (the most commonly used) that can be used for shared access in an HACMP cluster.

Table 2-7 External storage subsystems

IBM 7133 SSA Disk Subsystem Models D40 and T40 (up to 72.8 GB disk modules, and up to eight nodes per SSA loop).
IBM Enterprise Storage Server (ESS) Models E10, E20, F10, and F20 (supports up to eight nodes using SCSI and Fibre Channel interfaces via IBM FC/FICON, Feature Code: 3021, 3022, and 3023)
IBM 2105-800 (ESS) Total Storage Enterprise Storage Server (FS and SCSI)
IBM Total Storage FAStT 200, 500, 600, 700, and 900 models.
IBM 2106 Total Storage DS6000 and DS8000 series

HACMP also supports shared tape drives (SCSI or FC). The shared tape(s) can be connected via SCSI or FC. Concurrent mode tape access is *not* supported. See Table 2-8 on page 127 for some of the supported tape subsystems.

Table 2-8 Tape drive support

IBM 3583 Ultrium Scalable Tape Library Model L18, L32 and L72
IBM 3584 Ultra™ Scalable Tape Library Model L32 and D32
IBM Total Storage Enterprise Tape Drive 3590 Model H11
IBM Magstar® 3590 Tape Drive Model E11 & B11
IBM 3581 Ultrium Tape Autoloader Model H17 & L17
IBM 3580 Ultrium Tape Drive Model H11 & L11

For an updated list of supported storage and tape drives, check the IBM Web site at:

`http://www-1.ibm.com/servers/eserver/pseries/ha/`

HACMP may also be configured with non-IBM shared storage subsystems (disk and tape subsystems). For a list of non-IBM storage, refer to the respective manufacturer's Web sites, and at the Availant Web site:

`http://www.availant.com/`

Storage configuration is one of the most important tasks you have to perform before starting the HACMP cluster configuration. Storage configuration can be considered a part of HACMP configuration.

Depending on the application needs, and on the type of storage, you have to decide that how many nodes in a cluster will have shared storage access, and which resource groups will use which disks.

Most of the IBM storage subsystems are supported with HACMP. To find more information about storage server support, see the *HACMP for AIX 5L V5.3 Planning and Installation Guide*, SC23-4861-06.

2.9.1 Shared LVM requirements

Planning shared LVM for an HACMP cluster depends on the method of shared disk access and the type of shared disk device. The elements that should be considered for shared LVM are:

- Data protection method
- Storage access method
- Storage hardware redundancy

Note: HACMP itself does not provide storage protection. Storage protection is provided via:

- AIX (LVM mirroring)
- Hardware RAID

In this section, we provide information about data protection methods at the storage level, and also talk about the LVM shared disk access modes.

- Non concurrent
- Concurrent “classic” (HACMP concurrent logical volume manager - clvm)
- Enhanced concurrent mode (ECM), a new option in AIX 5L V5.1 and higher

2.9.2 Non-concurrent, enhanced concurrent, and concurrent

In a non-concurrent access configuration, only one cluster node can access the shared data at a time. If the resource group containing the shared disk space moves to another node, the new node will activate the disks, and check the current state of the volume groups, logical volumes, and file systems.

In non-concurrent configurations, the disks can be shared as:

- Raw physical volumes
- Raw logical volumes
- File systems

In a concurrent access configuration, data on the disks is available to all nodes concurrently. This mode does not support file systems (either JFS or JFS2).

Fast disk takeover

HACMP V5.1 exploits the new AIX enhanced concurrent LVM. In AIX 5L V5.2, any new concurrent volume group must be created in enhanced concurrent mode.

In AIX 5L V5.2 only, the enhanced concurrent volume groups can also be used for file systems (shared or non-shared). This is exploited by the fast disk takeover option to speed up the process of taking over the shared file systems in a fail-over situation.

The enhanced concurrent volume groups are varied on all nodes in the resource group, and the data access is coordinated by HACMP. Only the node that has the resource group active will vary on the volume group in “concurrent active” mode; the other nodes will vary on the volume group in “passive” mode. In “passive” mode, no high level operations are permitted on that volume group.

Attention: When using the resource groups with fast disk takeover option, it is extremely important to have redundant networks and non-IP networks. This will avoid data corruption (after all, the volume groups are in concurrent mode) in a “split brain” situation.

LVM requirements

The Logical Volume Manager (LVM) component of AIX manages the storage by coordinating data mapping between physical and logical storage. Logical storage can be expanded and replicated, and can span multiple physical disks and enclosures.

The main LVM components are:

- Physical volume

  A physical volume (PV) represents a single physical disk as it is seen by AIX (hdisk*). The physical volume is partitioned into physical partitions (PPs), which represent the physical allocation units used by LVM.

- Volume group

  A volume group (VG) is a set of physical volumes that AIX treats as a contiguous, addressable disk region. In HACMP, the volume group and all its logical volumes can be part of a shared resource group. A volume group cannot be part of multiple resource groups (RGs).

- Physical partition

  A physical partition (PP) is the allocation unit in a VG. The PVs are divided into PPs (when the PV is added to a VG), and the PPs are used for LVs (one, two, or three PPs per logical partition (LP)).

- Volume group descriptor area (VGDA)

  The VGDA is a zone on the disk that contains information about the storage allocation in that volume group.

  For a single disk volume group, there are two copies of the VGDA. For a two disk VG, there are three copies of the VGDA: two on one disk and one on the other. For a VG consisting of three or more PVs, there is one VGDA copy on each disk in the volume group.

- Quorum

  For an active VG to be maintained as active, a “quorum” of VGDAs must be available (50% + 1). Also, if a VG has the quorum option set to “off”, it cannot be activated (without the “force” option) if one VGDA copy is missing. If the quorum is be turned off, the system administrator must know the mapping of that VG to ensure data integrity.

- Logical volume

  A logical volume (LV) is a set of logical partitions that AIX makes available as a single storage entity. The logical volumes can be used as raw storage space or as file system's storage. In HACMP, a logical volume that is part of a VG is already part of a resource group, and cannot be part of another resource group.

- Logical partition

  A logical partition (LP) is the space allocation unit for logical volumes, and is a logical view of a physical partition. With AIX LVM, the logical partitions may be mapped to one, two, or three physical partitions to implement LV mirroring.

  Note: Although LVM mirroring can be used with any type of disk, when using IBM 2105 Enterprise Storage Servers or FAStT storage servers, you may skip this option. These storage subsystems (as well as some non-IBM ones) provide their own data redundancy by using various levels of RAID.

- File systems

  A file system (FS) is in fact a simple database for storing files and directories. A file system in AIX is stored on a single logical volume. The main components of the file system (JFS or JFS2) are the logical volume that holds the data, the file system log, and the file system device driver. HACMP supports both JFS and JFS2 as shared file systems, with the remark that the log must be on a separated logical volume (JFS2 also may have inline logs, but this is not supported in HACMP).

Forced varyon of volume groups

HACMP V5.1 provides a new facility, the forced varyon of a volume group option on a node. If, during the takeover process, the normal **`varyon`** command fails on that volume group (lack of quorum), HACMP will ensure that at least one valid copy of each logical partition for every logical volume in that VG is available before varying on that VG on the takeover node.

Forcing a volume group to varyon lets you bring and keep a volume group online (as part of a resource group) as long as there is one valid copy of the data available. You should use a forced varyon option only for volume groups that have mirrored logical volumes, and use caution when using this facility to avoid creating a partitioned cluster.

Note: You should specify the super strict allocation policy for the logical volumes in volume groups used with the forced varyon option. In this way, the LVM makes sure that the copies of a logical volume are always on separate disks, and increases the chances that forced varyon will be successful after a failure of one or more disks.

This option is useful in a takeover situation in case a VG that is part of that resource group loses one or mode disks (VGDAs). If this option is not used, the resource group will not be activated on the takeover node, thus rendering the application unavailable.

When using a forced varyon of volume groups option in a takeover situation, HACMP first tries a normal **`varyonvg`**. If this attempt fails due to lack of quorum, HACMP checks the integrity of the data to ensure that there is at least one available copy of all data in the volume group before trying to force the volume online. If there is, it runs **`varyonvg -f`**; if not, the volume group remains offline and the resource group results in an error state.

Note: The users can still use quorum buster disks or custom scripts to force varyon a volume group, but the new forced varyon attribute in HACMP automates this action, and customer enforced procedures may now be relaxed.

For more information see Chapter 5, “Planning Shared LVM Components” in the *HACMP for AIX 5L V5.3 Planning and Installation Guide*, SC23-4861-06.

Part 2

Planning, installation, and migration

In Part 2 we provide information about HACMP cluster and environment planning, and how to install a sample cluster.

We also present examples for migrating a cluster from an earlier HACMP version to the latest HACMP 5.3. Our scenarios provide step-by-step instructions and comments, and also some problem determination for migration.

3

Planning

This chapter discusses the planning aspects for an HACMP cluster. Adequate planning and preparation is required to successfully install and maintain an HACMP cluster. Time spent properly planning your cluster configuration and preparing your environment will result in a cluster that is easier to install, provide higher application availability, and easier to maintain.

Before you begin planning your cluster, you must have a good understanding of your current environment, your application, and your expectations for HACMP. Building on this information, you can develop an implementation plan that will allow you to easily integrate HACMP into your environment, and more importantly, have HACMP manage your application availability to your expectations.

In addition to Chapter 2, “High availability components” on page 37 which discusses HACMP concepts and basic design considerations, this chapter focuses on the steps required to plan for an HACMP 5.3 implementation. For ease of explanation, we use the planning and preparation for a 2-node mutual takeover cluster as an example. This configuration is the starting point for more advanced installations.

3.1 High availability planning

The primary goal of planning a high availability cluster solution is to eliminate or minimize service interruptions for a chosen application. To that end, single points of failure, in both hardware and software, must be addressed. This is usually accomplished through the use of N+1 redundant hardware, such as power supplies, network interfaces, SAN adapters, and mirrored or RAID disks. All of these components carry an additional server cost, and may not protect the application in the event of a server or operating system failure.

Note: Detailed planning information is found in the manual *High Availability Cluster Multi-Processing for AIX 5L Planning and Installation Guide*, SC23-4861-06.

Here is where HACMP comes in. HACMP can be configured to monitor server hardware, operating system, and application components. In the event of a failure, HACMP can take corrective actions, such as moving specified resources (service IP addresses, storage, and applications) to surviving cluster components in order to restore application availability as quickly as possible.

Since HACMP is an extremely flexible product, designing a cluster to fit your organization requires thoughful planning. Knowing your application requirements and behavior provides important input into your HACMP plan and will be primary factors in determining the cluster design. Ask yourself the following questions while developing your cluster design:

- Which application services are required to be highly available?
- What are the service level requirements for these application services (24/7, 8/5) and how quickly must service be restored in the event of a failure?
- What are the potential points of failure in the environment and how can they be addressed?
- Which points of failure can be automatically detected by HACMP and which would require custom code to be written to trigger an event?
- What is the skill level within the group implementing and maintaining the cluster?

Although the AIX system administrators are typically responsible for the implementation of HACMP, usually they cannot do it on their own. A team consisting of the following representatives should be assembled to assist with the HACMP planning as each will play a role in the success of the cluster:

- Network administrator
- AIX system administrator
- Database administrator
- Application programmer

- Support personnel
- Application users

3.2 Planning for HACMP

The major steps to a successful HACMP implementation are identified in Figure 3-1 on page 138. Notice that a cluster implementation does not end with the successful configuration of a cluster. Cluster testing, backups, documentation and ongoing management procedures are equally important to ensure the ongoing integrity of the cluster.

Using the concepts discussed in Chapter 1, “Introduction to HACMP” on page 3, begin an HACMP implementation by developing a detailed HACMP cluster configuration and implementation plan. Planning tools such as the Paper Planning Worksheets (found in the *High Availability Cluster Multi-Processing for AIX 5L Planning and Installation Guide*, SC23-4861-06), and Online Planning Worksheets can be used to guide you through this process and record the cluster details.

Important: Remember that time spent planning the cluster will translate into an easier implementation, so take your time.

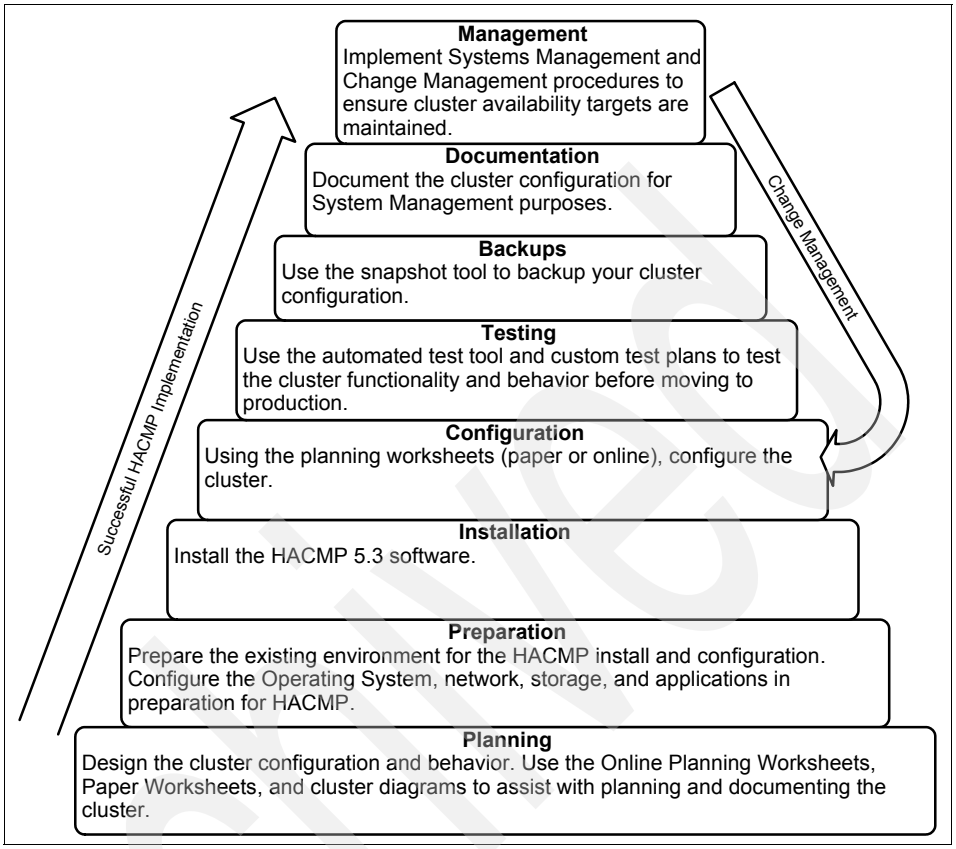

Figure 3-1 HACMP implementation steps

As illustrated in Figure 3-1, planning is the foundation upon which the implementation is built. Proper planning should touch on all aspects of the cluster implementation. It should include:

- The cluster design and behavior.
- A detailed cluster configuration.
- Installation considerations and plan.
- A plan to test the integrity of the cluster.
- A backup strategy for the cluster.
- A procedure for documenting the cluster.
- A plan to manage problems and changes in the cluster.

Important: For a successful implementation, HACMP Cluster planning and preparation should be completed before you install or configure HACMP.

For ease of explanation, we will use the planning of a simple two-node mutual takeover cluster as an example. Sample planning worksheets are included as we work through this chapter in order for you to see how the cluster planning is developed.

3.2.1 Planning tools

Three tools are available to help with the planning of an HACMP cluster:

- Cluster diagram
- Paper Planning Worksheets
- Online Planning worksheets

Both the cluster diagram and the Paper Planning Worksheets provide a manual method of recording your cluster information. The Online Planning Worksheets provides an easy to use java-based interface that can be used to record and configure your cluster.

All three tools are discussed in more detail towards the end of this chapter.

Note: If you decide to use the Online Planning Worksheets (OLPW), or the two-node cluster configuration assistant, it is still important that HACMP planning and preparation be completed. The OLPW and two-node cluster configuration assistant are simply intended to ease with the documentation and configuration of the cluster, you must still have a good understanding of the planning and preparation.

3.3 Getting started

Begin cluster planning by assessing the current environment and your expectations for HACMP:

- Which applications need to be highly available?
- How many nodes are required to support the applications?
- Are the existing nodes adequate in size (CPU/memory) to run multiple applications or is this a new installation?
- How do the clients connect to the application, what is the network configuration?
- The type of shared disk is to be used?
- What are the expectations for HACMP?

3.3.1 Current environment

Figure 3-2 on page 141 illustrates a simple starting configuration which we will use as our example. It focuses on two applications to be made highly available. This could be an existing pair of servers or two new servers.

The starting configuration shows:

- Each application resides on a separate node (server).
- Clients access each application over a dedicated Ethernet connection on each server.
- Each node is relatively the same size in terms of CPU and memory, each with additional capacity.
- Each node has redundant Power supplies and mirrored internal disks.
- The applications reside on external SAN disk.
- The applications each have their own robust start and stop scripts.
- There is a monitoring tool to verify the health of each application.
- AIX 5.3 is already installed

Important: Each application to be integrated into the cluster must run in standalone mode. You also must be able to fully control the application (start, stop, and validation test).

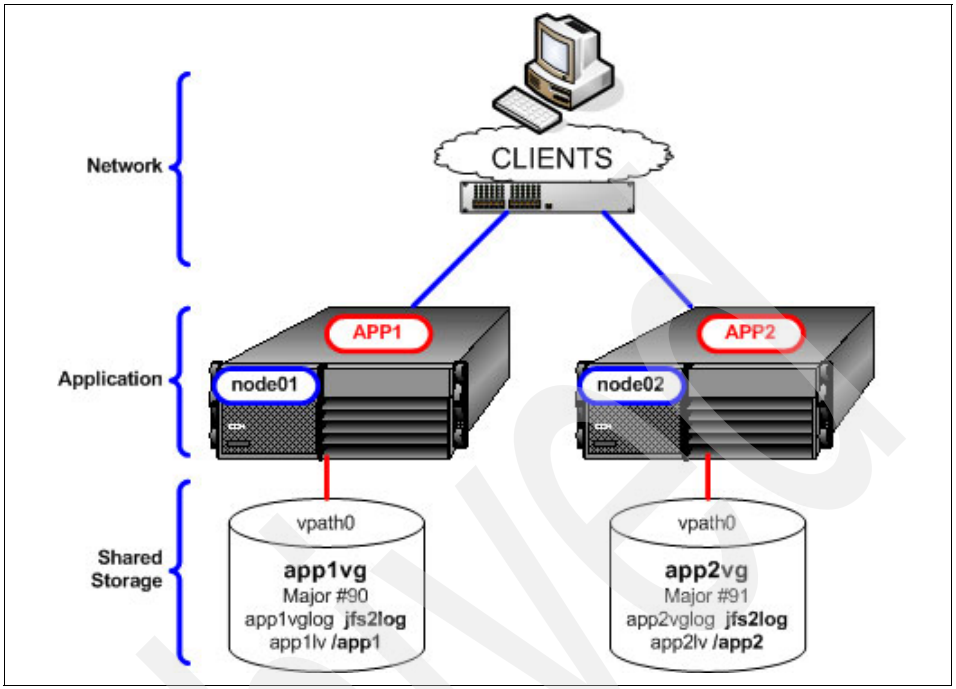

Figure 3-2 Initial Environment

The intention is to make use of the two nodes in a mutual takeover configuration where app1 normally resides on node01, and app2 normally resides on node02. In the event of a failure we want both applications to run on the surviving server. We can see from the diagram that we need to prepare the environment in order to allow each node to run both applications.

Attention: Each application to be integrated into the cluster must be able to run in standalone mode on any node it may have to run on (under both normal and failover situations).

Analyzing the HACMP cluster requirements, we have three key focus areas as illustrated in Figure 3-2; network, application and storage. All planning activities will be in support of one of these three items to some extent.

1. **Network** - How will the clients connect to the application (the service address). The service address will float between all designated cluster nodes.
2. **Application** - What resources are required by the application. The application must have everything it needs to run on a failover node including, CPU and memory resources, licensing, run-time binaries, and configuration data. It should have robust start and stop scripts as well as a tool to monitor its status.

3. **Storage** - What type of shared disk will be used. The application data must must reside on a shared disk that is available to all required cluster nodes.

3.3.2 Addressing single points of failure

Table 3-1 summarizes the various single points of failure found in the cluster infrastructure and how to protect against them. These items should be considered during the development of the detailed cluster design.

Table 3-1 Single Points of Failure

Cluster objects	To eliminate as single point of failure	HACMP / AIX supports
Nodes	Use Multiple Nodes.	Up to 32.
Power sources	Use Multiple circuits or uninterruptible power supplies (UPS).	As many as needed.
Networks	Use multiple networks to connect nodes.	Up to 48.
Network interfaces, devices, and IP addresses.	Use redundant network adapters.	Up to 256.
TCP/IP subsystem	Use point-to-point networks to connect adjoining nodes and clients.	As many as needed.
Disk adapters	Use redundant disk adapters.	As many as needed.
Storage controllers	Use redundant disk controllers.	As many as needed (hardware limited).
Disks	Use redundant hardware and disk mirroring, striping, or both.	As many as needed.
Applications	Assign a node for application takeover, configuring application monitors, 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 objects	To eliminate as single point of failure	HACMP / AIX supports
Cluster resources	Use multiple cluster resources.	Up to 128 for Clinfo (more can exist in a cluster).

3.3.3 Initial cluster design

Now that we have an understanding of the current environment, HACMP concepts, and our expectations for the cluster, we can begin the cluster design.

Now is a good time to create a diagram of the HACMP cluster. Start simple at first and gradually increase the level of detail as you go through the planning process. The diagram will help identify single points of failure, application requirements, and will help guide you along the planning process.

The paper or online planning worksheets should also be used to record the configuration and cluster details as you go.

Figure 3-3 on page 144 illustrates the initial cluster diagram used in our example. At this point, the focus is on high level cluster functionality, cluster details will be developed as we move through the planning phase.

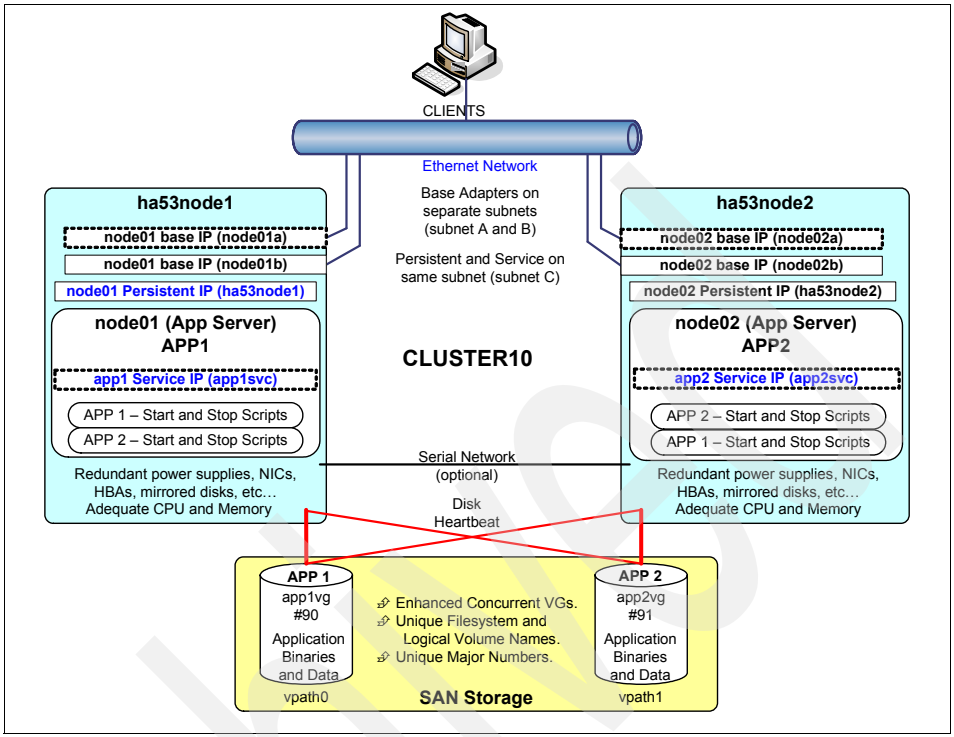

Figure 3-3 Initial cluster design

We begin to make design decisions for the cluster topology and behavior based on our requirements. For example, based on our requirements, the initial cluster design for our example includes the following:

- The cluster will be a two node mutual takeover cluster.
- Hostnames could be used as cluster node names but we choose to specify cluster node names instead.

> **Note:** If you plan to use DLPAR, the AIX hostname, the cluster node name, and the HMC LPAR name (as seen in the HMC GUI) must all match.

- Each node contains one application but is capable of running both (consider network, storage, memory, CPU, software).
- Each node has two Ethernet adapters connected to the same physical Ethernet network, each to a separate switch (or some type of redundant switch).

- We use IPAT (IP Address Takover) via aliasing (as opposed to IPAT via replacement).
- Each node has a persistent IP address (an IP alias always available while the node is up) and one service IP (aliased to one of the adapters under HACMP control). The base ethernet adapter addresses are on separate subnets. Since heartbeating over aliases will be used, this is not mandatory, both adapters could reside in the same subnet if desired.
- Shared disks reside on a SAN and are available on both nodes.
- All VGs on the shared disks are created in Enhanced Concurrent Mode (ECM) in order to allow for the use of heartbeating over disk and fast disk takeover.
- A serial RS232 connection is shown but since heartbeating over disk will be used, you can skip this in your design. It is simply shown as optional.
- Heartbeating over IP aliases will be used (not shown).
- Each node has sufficient CPU/memory resources required to run both applications.
- Each node has redundant hardware and mirrored internal disks.
- AIX 5.3 ML02 is installed.
- HACMP 5.3 will be used.

This list simply captures the basic components of the cluster design. Each item will be investigated in further detail as we progress through the planning stage.

3.3.4 Comprehensive the cluster overview planning worksheet

Complete the initial worksheet as we have done in or our example. There are 11 worksheets found in this chapter, each covering different aspects of the cluster planning. Table 3-2 shown the first worksheet listing the basic cluster elements.

Table 3-2 Cluster overview

HACMP CLUSTER WORKSHEET - PART 1 of 11 **CLUSTER OVERVIEW**	**DATE: July 2005**
CLUSTER NAME	cluster10
ORGANIZATION	IBM ITSO
NODE 1 HOSTNAME	ha53node1
NODE 2 HOSTNAME	ha53node2
NODE 1 HACMP NAME	node01

HACMP CLUSTER WORKSHEET - PART 1 of 11 CLUSTER OVERVIEW		DATE: July 2005
NODE 2 HACMP NAME	node02	
COMMENTS	This is a set of planning tables for a simple 2-node HACMP 5.3 mutual takeover cluster using IPAT via Aliasing.	

3.4 Planning cluster hardware

Cluster design starts by determining how many and what type of nodes are required. This depends largely on a couple of factors:

- The amount of resources required by each application.
- The failover behavior of the cluster.

Note: The number of nodes in a cluster can range from 2 to 32.

A primary consideration when choosing nodes is that in a failover situation, the surviving node or nodes must be capable of running the failing node's application(s). That is, if you have a two node cluster and one node fails, the surviving node must have all the resources required to run the failing node's applications (in addition to its own applications). If this is not possible, you might consider implementing an additional node as a standby node, or consider using the dynamic LPAR (DLPAR) feature (POWER4™ or POWER5™ systems). As you will notice, HACMP allows for a wide range of cluster configurations depending on your requirements.

HACMP supports virtually any AIX supported node, from desktop systems to high end servers. When choosing a type of node, consider the following:

- Ensure there are sufficient CPU and memory resources available on all nodes to allow the system to behave as desired in a failover situation. The CPU and memory resources must be capable of sustaining the selected applications during failover, otherwise clients may experience performance problems. If you are using LPARs, you may want to make use of the DLPAR capabilities to increase resources during failover. If you are using standalone servers, you do not have this option and so you might have to look at using a standby server.
- Make use of highly available hardware and redundant components where possible in each server. For example, use redundant power supplies and connect them to separate power sources.
- Protect each node's rootvg (local operating system copy) through the use of mirroring or raid.

- Allocate at least two Ethernet adapters per node and connect them to separate switches to protect from a single adapter or switch failure.
- Allocate two SAN adapters per node to protect from a single SAN adapter failure.

Although not mandatory, we suggest using cluster nodes with similar hardware configurations in order to make it easier to distribute the resources and perform administrative operations. That is, don't try to failover from a high-end server to a desktop model and expect everything to work properly, be thoughtful in your choice of nodes.

3.4.1 Complete the cluster hardware planning worksheet

The following worksheet (Table 3-3) contains the hardware specifics for our example. Where possible we have made use of redundant hardware, additional Ethernet and SAN switches, and ensured we have enough resources to sustain both applications simultaneously on any node.

Table 3-3 Cluster hardware

HACMP CLUSTER WORKSHEET - PART 2 of 11 CLUSTER HARDWARE		**DATE: July 2005**
HARDWARE COMPONENT	**SPECIFICATIONS**	**COMMENTS**
p520	pSeries Server 2 CPU and 4GB Memory 4 Internal SCSI Disks	Quantity 2 *Latest Firmware Redundant power supplies. Sufficient resources to run both applications on one server.
Ethernet Adapters	10/100/1000 Ethernet	2 NICs per node (minimum).
Network Switches	Vendor Name - Model	2 Switches. Each NIC on a node connected to a separate switch. All ports are configured in the same VLAN. Switches support Gratuitous Arp and Spanning Tree Disabled. Switch port speed set appropriately.
SAN Adapters	6239 2GB Fibre Channel	2 HBAs per node
SAN Switches	IBM 2109	2 switches. Each HBA on a node connected to a separate switch. Shared disk Zoned to all nodes requiring access.

HACMP CLUSTER WORKSHEET - PART 2 of 11 CLUSTER HARDWARE		DATE: July 2005
SAN Storage	IBM ESS	Model number 800
COMMENTS	All Hardware compatibility verified.	

3.5 Planning cluster software

Review all software components to be used in the cluster to ensure compatibility. Items to consider are AIX, RSCT, HACMP, application, and storage software. This section discusses the various software levels and compatibilities.

3.5.1 AIX and RSCT Levels

HACMP 5.3 is supported on AIX versions 5.2 and 5.3. The specific levels of AIX and RSCT combinations we have used in our environment are listed in Table 3-4.

Table 3-4 AIX and RSCT levels

AIX Version	RSCT Version	Minimum RSCT Filesets
AIX 5.3 ML02 (5300-02) or later with APARS.	2.4.2	rsct.compat.basic.hacmp.2.4.2.0 rsct.compat.clients.hacmp.2.4.2.0 rsct.core.sec.2.4.2.1 rsct.core.rmc.2.4.2.1
AIX 5.2 ML06 (5200-06) or later with APARS IYXXXXX	2.3.6	rsct.compat.basic.hacmp.2.3.6.0 rsct.compat.clients.hacmp.2.3.6.0 rsct.core.sec.2.3.6.1 rsct.core.rmc.2.3.6.1

3.5.2 Virtual LAN and SCSI Support

The following software levels are required to support IBM's Virtual LAN and Virtual SCSI features found in IBM's P5 Virtual I/O Server.

- AIX 5.3 ML02 (5300-02) with APARs IY70082 and iFIX IY72974.
- VIO Server V1.1 with Fixpack 6.2 and iFIX IY71303.062905.epkg.Z
- Minimum RSCT levels
  - rsct.basic.hacmp 2.4.2.1
  - rsct.basic.rte 2.4.2.2
  - rsct.compat.basic.hacmp 2.4.2.0

3.5.3 Required AIX Filesets

The following filesets are required for HACMP. They must be installed with the latest version of fixes for the appropriate AIX level before HACMP is installed.

- bos.adt.lib
- bos.adt.libm
- bos.adt.syscalls
- bos.net.tcp.client
- bos.net.tcp.server
- bos.rte.SRC
- bos.rte.libc
- bos.rte.libcfg
- bos.rte.libcur
- bos.rte.libpthreads
- bos.rte.odm
- bos.rte.lvm.rte (required only using existing or legacy 32-bit Concurrent Logical Volume Manager for concurrent access)
- bos.clvm.enh (required for Enhanced Concurrent Logical Volumes - disk heartbeating makes use of this feature)

3.5.4 AIX Security Filesets

The following filesets are required if you plan to use message authentication or encryption for HACMP communication between cluster nodes. They can be installed from the AIX 5L Expansion Pack CD-ROM.

- rsct.crypt.des - for data encryption with DES message authentication
- rsct.crypt.3des - for data encryption standard Triple DES message authentication.
- rsct.crypt.aes256 - for data encryption with Advanced Encryption Standard (AES) message authentication.

Note: These filesets are not supported on AIX 5.1.

3.5.5 Software required by WebSmit

The following software is required if you plan to install and configure WebSmit. Versions shown are the latest at the time of writing.

- Apache compliant Web server (Apache or IBM Http Server - IHS). Apache 1.3.31 can be downloaded from:
  http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html
  - IBM IHS v2.0.47.1 can be downloaded from:
    http://www-306.ibm.com/software/webservers/httpservers/
- RPM Package Manager (if not already on your system)

  - expat-195.7-1.aix5.1.ppc.rpm can be downloaded from:
    http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html
- openssl can be downloaded from:
  http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html
  - Select “AIX Toolbox Cryptographic Content” and download the following files:
    - apache-1.3.31-1ssl.aix5.1.ppc.rpm
    - mod_ssl-2.8.19-1ssl.aix5.1.ppc.rpm
    - openssl-0.9.7d-1.ssl.aix5.1.ppc.rpm

Note: You need to register with **ibm.com®**, thus make sure you allow 24 hours for the registration to complete. If you are NOT registered, you can’t download the cryptographic content files.

3.5.6 HACMP Filesets

The following HACMP filesets can be installed from the install media (excluding additional language filesets):

- cluster.adt.es.client.include
- cluster.adt.es.client.samples.clinfo
- cluster.adt.es.client.samples.clstat
- cluster.adt.es.client.samples.libcl
- cluster.adt.es.java.demo.monitor
- cluster.assist.license HACMP Smart Assist Feature
- cluster.doc.en_US.assist.db2.html
- cluster.doc.en_US.assist.db2.pdf
- cluster.doc.en_US.assist.oracle.html
- cluster.doc.en_US.assist.oracle.pdf
- cluster.doc.en_US.assist.websphere.html
- cluster.doc.en_US.assist.websphere.pdf
- cluster.doc.en_US.es.html HAES Web-based HTML
- cluster.doc.en_US.es.pdf HAES PDF Documentation - U.S.
- cluster.doc.en_US.glvm.html
- cluster.doc.en_US.glvm.pdf
- cluster.doc.en_US.pprc.pdf
- cluster.doc.en_US.pprc.html
- cluster.es.assist.common HACMP Smart Assist Common
- cluster.es.assist.db2 HACMP Smart Assist for DB2
- cluster.es.assist.oracle HACMP Smart Assist for Oracle
- cluster.es.assist.websphere
- cluster.es.cfs.rte
- cluster.es.client.lib ES Client Libraries
- cluster.es.client.rte ES Client Runtime

- cluster.es.client.utils ES Client Utilities
- cluster.es.client.wsm Web based Smit
- cluster.es.clvm.rte ES for AIX Concurrent Access
- cluster.es.cspoc.cmds ES CSPOC Commands
- cluster.es.cspoc.dsh ES CSPOC dsh
- cluster.es.cspoc.rte ES CSPOC Runtime Commands
- cluster.es.ercmf.cmds
- cluster.es.ercmf.rte
- cluster.es.plugins.dns
- cluster.es.plugins.printserver
- cluster.es.plugins.dhcp
- cluster.es.pprc.cmds
- cluster.es.pprc.rte
- cluster.es.server.cfgast ES Two-Node Configuration
- cluster.es.server.diag ES Server Diags
- cluster.es.server.events ES Server Events
- cluster.es.server.rte ES Base Server Runtime
- cluster.es.server.testtool
- cluster.es.server.utils ES Server Utilities
- cluster.es.svcpprc.cmds
- cluster.es.svcpprc.rte
- cluster.es.worksheets Online Planning Worksheets
- cluster.hativoli.client
- cluster.hativoli.server
- cluster.license HACMP Electronic License
- cluster.man.en_US.assist.data
- cluster.man.en_US.es.data
- cluster.msg.en_US.assist
- cluster.msg.en_US.cspoc
- cluster.msg.en_US.ercmf
- cluster.msg.en_US.es.client
- cluster.msg.en_US.es.server
- cluster.msg.en_US.hativoli
- cluster.msg.en_US.pprc
- cluster.msg.en_US.svcpprc
- cluster.xd.glvm
- cluster.xd.license
- glvm.rpv.util Geographic LVM
- glvm.rpv.client
- glvm.rpv.server
- glvm.rpv.msg.en_US
- hageo.doc.en_US.data
- hageo.gmdsizing
- hageo.man.en_US.message.data
- hageo.man.en_US.mirror.data

- hageo.manage.utils
- hageo.message.ext
- hageo.message.utils
- hageo.mirror.ext
- hageo.mirror.utils
- hageo.msg.en_US.message
- hageo.msg.en_US.mirror

3.5.7 AIX files altered by HACMP

Be aware that the following system files may be altered by HACMP during cluster packages installation, verification and synchronization process.

/etc/hosts

The cluster event scripts use the /etc/hosts file for name resolution. All cluster node IP interfaces must be added to this file on each node. HACMP may modify this file to ensure that all nodes have the necessary information in their /etc/hosts file, for proper HACMP operations.

If you delete service IP labels from the cluster configuration using SMIT, we recommend that you also remove them from /etc/hosts.

/etc/inittab

The /etc/inittab file is modified in each of the following cases:

- HACMP is installed
  - The following line is added when you initially install HACMP. It will start the clcomdES and clstrmgrES subsystems if they are not already running.
    hacmp:2:once:/usr/es/sbin/cluster/etc/rc.init >/dev/console 2>&1

Important: This HACMP entry is used to start the following daemons using the startsrc command if they are not already running.

- startsrc -s syslogd
- startsrc -s snmpd
- startsrc -s clcomdES
- startsrc -s clstrmgrES

- HACMP is configured for IP address takeover
  - **harc:2:wait:/usr/es/sbin/cluster/etc/harc.net # HACMP network startup**
  - When IP address takeover is enabled, the system edits /etc/inittab to change the rc.tcpip and inet-dependent entries from run level "**2**" (the default multi-user level) to run level "**a**".

- Entries that have run level “a” are processed only when the telinit command is executed specifying that specific run level.

- The Start at System Restart option is chosen on the SMIT System Management (C-SPOC) > Manage HACMP Services > Start Cluster Services panel
  - **hacmp6000:2:wait:/usr/es/sbin/cluster/etc/rc.cluster -boot -i # Bring up Cluster**
  - When the system boots, the /etc/inittab file calls the /usr/es/sbin/cluster/etc/rc.cluster script to start HACMP.
  - Because the inet daemons must not be started until after HACMP-controlled interfaces have swapped to their service IP address, HACMP also adds the following entry to the end of the /etc/inittab file to indicate that /etc/inittab processing has completed:
    - **clinit:a:wait:/bin/touch /usr/es/sbin/cluster/.telinit #HACMP for AIX These must be the last entry in run level “a” in inittab!**
    - **pst_clinit:a:wait:/bin/echo Created /usr/es/sbin/cluster/.telinit > /dev/console #HACMP for AIX These must be the last entry in run level “a” in inittab!**

Attention: Although it is possible to start Cluster Services from the inittab, we strongly recommend that is option not be used. It is best to manually control the starting of HACMP. For example, in the case of a failure, it is best to investigate the cause of the failure before restarting HACMP on a node.

- Concurrent Logical Volume Manager (cluster.es.clvm) is installed with HACMP.
  - The following entry is automatically added to the /etc/inittab file: **haclvm_cfg:2:wait:/usr/es/sbin/cluster/clvm/config_mode3**

Note: ha_star is also found as an entry in the inittab. This fileset is delivered with the bos.rte.control fileset and not HACMP.

/etc/rc.net

The /etc/rc.net file is called by cfgmgr, (cfgmgr is the AIX 5L utility that configures devices and optionally installs device software into the system), to configure and start TCP/IP during the boot process. It sets hostname, default gateway, and static routes.

/etc/services

HACMP makes use of the following network ports for communication between cluster nodes. These are all listed in the /etc/services file.

- clinfo_deadman6176/tcp

- clsmuxpd 6270/tcp
- clm_lkm 6150/tcp
- clm_smux 6175/tcp
- godm 6177/tcp
- topsvcs 6178/udp
- grpsvcs 6179/udp
- emsvcs 6180/udp
- #clver 6190/tcp (commented out because clverify now used clcomd)
- clcomd 6191/tcp
- clinfo_client 6174/tcp
- #cllockd 6100/udp (commented out - cllockd not supported since HACMP 5.2)
- #clm_mig_1k 6151/tcp (commented out - cllockd not supported since HACMP 5.2)

Note: If you install HACMP/XD for GLVM, the following entry for the port number and connection protocol is automatically added to the /etc/services file on each node on the local and remote sites on which you installed the software:

- rpv 6192/tcp. Application Requirements

In addition to HACMP, RMC uses the following ports:

- #rmc 657/tcp
- #rmc 657/udp

WebSmit typically uses the following port. The WebSmit port is configurable.

- #http 42267 (WebSmit port)

/etc/snmpd.conf

The version of snmpd.conf depends on whether you are using AIX 5L V5.1 or later. The default version of the file for versions of AIX 5L later than V5.1 is snmpdv3.conf.

The SNMP daemon reads the /etc/snmpd.conf configuration file when it starts up and when a refresh or kill -1 signal is issued. This file specifies the community names and associated access privileges and views, hosts for trap notification, logging attributes, snmpd-specific parameter configurations, and SMUX configurations for the snmpd. The HACMP installation process adds a clsmuxpd password to this file.

The following entry is added to the end of the file, to include the HACMP MIB supervised by the Cluster Manager:

- **smux 1.3.6.1.4.1.2.3.1.2.1.5 "clsmuxpd_password" # HACMP clsmuxpd**

/etc/snmpd.peers

The /etc/snmpd.peers file configures snmpd SMUX peers. During installation, HACMP adds the following entry to include the clsmuxpd password to this file:

- **clsmuxpd 1.3.6.1.4.1.2.3.1.2.1.5 "clsmuxpd_password" # HACMP clsmuxpd**

/etc/syslog.conf

- The /etc/syslog.conf configuration file is used to control output of the syslogd daemon, which logs system messages. During the install process HACMP adds entries to this file that direct the output from HACMP-related problems to certain files.

Example:

- # HACMP Critical Messages from HACMP
- local0.crit /dev/console
- # HACMP Informational Messages from HACMP
- local0.info /usr/es/adm/cluster.log
- # HACMP Messages from Cluster Scripts
- user.notice /usr/es/adm/cluster.log
- # HACMP/ES for AIX Messages from Cluster Daemons
- daemon.notice /usr/es/adm/cluster.log

The /etc/syslog.conf file should be identical on all cluster nodes.

/etc/trcfmt

The /etc/trcfmt file is the template file for the system trace logging and report utility, trcrpt. The installation process adds HACMP tracing to the trace format file. HACMP tracing is performed for the clstrmgr and clinfo daemons.

/var/spool/cron/crontab/root

The HACMP installation process adds HACMP logfile rotation to the /var/spool/cron/crontab/root file.

0 0 * * * /usr/es/sbin/cluster/utilities/clcycle 1>/dev/null 2>/dev/null # > HACMP for AIX Logfile rotation

3.5.8 Application software

Typically applications are not dependant on HACMP versions as they are not aware of the underlining HACMP functionality. That is, HACMP simply starts and stops them (HACMP can also monitor applications, but generally using an application dependent method).

There are a few applications however, such as Oracle RAC 9i that are tied closer to the version of HACMP being used.

Check with the application vendor to ensure there are no issues (such as licensing) with the use of HACMP 5.3.

3.5.9 Licensing

You have to pay attention to two aspects of licensing: HACMP (features) licensing and application licensing.

HACMP

Beginning with HACMP V5.2, HACMP licensing is based on the number of processors, where the number of processors is the sum of the number of processors on which HACMP V5 will be installed or run. An HACMP license is required for each machine on which HACMP will be installed and run.

HACMP/XD V5 is licensed by the number of processors. The number of processors is the sum of the number of processors on which HACMP/XD V5 will be installed or run. A license for HACMP V5 and a license for HACMP/XD V5 is required for each machine on which HACMP/XD will be installed and run.

HACMP V5 Smart Assist is licensed by the number of processors. The number of processors is the sum of the number of processors on which HACMP V5 Smart Assist will be installed or run. A license for HACMP V5 and a license for HACMP V5 Smart Assist is required for each machine on which HACMP Smart Assist will be installed or run.

So what does this mean.

- If you have a pSeries server with 4 CPU's running in full system partition mode, you require a licence for 4 CPU's.
- If you have a pSeries server with 4 CPU's running logical partitions and you only run HACMP in a 2 CPU partition, you require a licence for 2 CPU's.
- Of course, you require a licence for each server you plan to run HACMP on.

Note: Micro partition licensing for HACMP is not available. You must license by full processors.

Application

Some applications require specific licensing requirements such as 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.

Important: Check with your application vendor for any licence issues when using HACMP

3.5.10 Complete the software planning worksheet

The following worksheet (Table 3-5) contains a list of all software installed in our example:

Table 3-5 Cluster Software

HACMP CLUSTER WORKSHEET - PART 3 of 11 CLUSTER SOFTWARE		**DATE: July 2005**
SOFTWARE COMPONENT	**VERSION**	**COMMENTS**
AIX	5.3 ML02	Latest AIX Version
RSCT	2.4.2.1	Latest RSCT Version
HACMP	5.3 BASE	GA Version
IBM SDD	1.6.0.2	Storage Multipathing Software
APPLICATION	Test Application Version 1	Add your application versions.
COMMENTS	All Software compatibility verified. No issues running Applications with HACMP. HACMP Licensing for 4 CPU's on each node. Application licensing verified and licence purchased for both servers.	

3.6 Operating system considerations

In addition to the AIX operating system levels and filesets, there are a few other operating system aspects to consider during the planning stage.

Disk space requirements

HACMP requires the following available space in roovg volume group for installation:

- **/usr** requires **82MB** of free space for a full installation of HACMP.
- **/** (root) requires **710KB** of free space.

It is also good practice to allow approximately 100MB free space in **/var** and **/tmp** for HACMP logs (The space required depends on the number of nodes in the cluster which dictate the size of the messages stored in the various HACMP logs).

Time synchronization

Time synchronization is important between cluster nodes for both application and hacmp log issues. This is standard system administration practice and we recommend you make use of an ntp server or other procedure to keep the cluster nodes time in sync.

Note: Maintaining time synchronization between the nodes is especially useful for auditing and debugging cluster problems.

Operating system settings

There are no additional operating system settings required for HACMP. Follow normal AIX tuning as required by application workload.

3.7 Planning security

Protecting you cluster nodes (and application) from unauthorized access is an important factor of the overall system availability. There are certain general security considerations, as well as HACMP related aspects, which we emphasize in this section.

3.7.1 Cluster security

HACMP needs a way to authenticate to all node in the cluster for executing remote commands related to cluster verification, synchronization and certain administrative operations (C-SPOC).

Cluster security is required to prevent unauthorized access to cluster nodes. Starting in HACMP 5.1, a new security mechanism, facilitated by the cluster communication daemon (clcomdES), provides additional cluster security.

The dependency on AIX rsh (and thus on /.rhosts file) has been eliminated. As some commands external to HACMP, for example user-defined scripts, may still require remote command execution using “r” commands, you need to asses if you still need to keep the ~/.rhosts file.

HACMP inter-node communication relies on a cluster daemon (clcomdES), which eliminates the need for AIX “classic” remote commands. See further explanations in this chapter for detailed information about clcomdES mechanism.

HACMP modes for connection authentication

- Standard security mode
  - Standard security is the default security mode.
  - Implemented directly by cluster communication daemon (clcomdES)
  - Uses node and adapter information stored in HACMP ODM classes and the /usr/es/sbin/cluster/etc/rhosts file to determine legitimate partners.
- Enhanced security mode (kerberos).
  - Kerberos security is available only for HACMP clusters implemented in an SP cluster.
  - Takes advantage of the Kerberos authentication method.

For improved security, you can also use VPN tunnels between cluster nodes. In this case, clcomdES traffic for IP interfaces/addresses configured in HACMP is sent through VPN tunnels provided AIX. If you use a VPN, use persistent addresses for the VPN tunnels. The VPNs are configured within AIX and then HACMP. HACMP provides a smit menu for ease of configuration.

In standard security mode, the remote command execution for HACMP remote commands in /usr/es/sbin/cluster uses the principle of least privilege. This ensures that no command can run on a remote node with root privilege, except for the ones in /usr/es/sbin/cluster. A select set of HACMP commands are considered trusted and allowed to run as root; all other commands run as user nobody.

To manage inter-node communications, the cluster communication daemon requires a list of valid cluster IP labels or addresses to use. There are two ways to provide this information:

- Automatic node configuration (default method)
- Individual node configuration (manual method)

Automatic node configuration

If you are configuring HACMP for the first time, the /usr/es/sbin/cluster/etc/rhosts file on a node is empty. Because clcomdES has to authenticate the IP address of the incoming connection to be sure that is from a node in the cluster, the rules for validating the addresses are based on the following process:

- If the /usr/es/sbin/cluster/etc/rhosts file is empty and there is no HACMP cluster defined on that node, then the first connection from another node will be authenticated and accepted. The content of the /usr/es/sbin/cluster/etc/rhosts file will be changed to include all “ping-able” base addresses of the network adapters from the communicating node.

- If a cluster is already defined on the node (HACMPcluster ODM class is not empty), then clcomdES looks for a communication path (IP address) in the HACMPnode and subsequently in the HACMPadapter ODM class. If it finds a valid communication path, it takes the first occurrence (HACMPnode, then HACMPadapter), otherwise the /usr/es/sbin/cluster/etc/rhosts file will be checked for a valid IP address.
- If clcomdES cannot authenticate incoming connections, it will fail, and you have to manually update the /usr/es/sbin/cluster/etc/rhosts file, and then recycle the clcomdES daemon (stopsrc -s clcomdES, startsrc -s clcomdES)

Typically, the user should not have to manually populate the rhosts file, but rather lets clcomdES do it. Since this file is empty upon installation, the first connection from another node will populate it. The first connection is usually verification and synchronization, and afterwards the HACMPnode and HACMPadapter ODMs are complete. After the cluster is synchronized, the rhosts file can be emptied but not removed. The information in HACMPnode and HACMPadapter is then used for clcomd authentication.

Attention:

- To ensure that an unauthorized host does not connect to a node between the time when you install HACMP software and the time when you initiate a connection from one cluster node to another, you can manually populate (edit) the /usr/es/sbin/cluster/etc/rhosts file to add one or more IP labels/addresses (that will be part of your cluster).
- If at a later time you decide to redo (start from scratch, or change the base IP addresses of the nodes) the cluster configuration, it is a good practice to also empty the contents (DO NOT delete) of the rhosts file on ALL the nodes you plan to (re)use for your cluster.

Individual node configuration

As an alternate solution, if you are especially concerned about network security (they may be building the cluster on an unsecured network) you may want to put all the IP addresses/labels in the /usr/es/sbin/cluster/etc/rhosts file prior to configuring the cluster.

The HACMP installation creates this empty file with read-write permissions for root only.

Note: Ensure that each IP address/label is valid for the cluster, otherwise an error is logged in the /var/hacmp/clcomd/clcomd.log.

To set up the /usr/es/sbin/cluster/etc/rhosts file:

1. As root, open the file /usr/es/sbin/cluster/etc/rhosts on a node.
2. Edit the file to add all possible network interface IP addresses for each node.
3. Put only one IP label or address on each line.

Note: If you disable the cluster communications daemon or completely REMOVE the /usr/es/sbin/cluster/etc/rhosts file, programs that require inter-node communication, such as C-SPOC, cluster verification and synchronization, file collections, and message authentication and encryption will no longer function.

For this reason it is mandatory to keep clcomdES running at all times.

3.7.2 User administration

Most of the application require that user information is consistent across the cluster nodes (user ID, group membership, group ID) in order that users can log in to surviving nodes without experiencing problems.

This is particularly important in a failover (takeover) situation. It is imperative that the application user be able to access the shared files from any required node in the cluster. This usually means that the application related UID and GID must be the same on all nodes.

In preparation for a cluster configuration, it is important that this be considered and corrected, otherwise you may experience service problems during a failover.

Once HACMP is installed, it contains facilities to let you manage AIX 5L user and group accounts across an HACMP cluster. It also provides a utility to authorize specified users to change their own password across nodes in an HACMP cluster.

Attention: If you manage user accounts with a utility such as Network Information Service (NIS), PSSP user management, or Distributed Computing Environment (DCE) Manager, do NOT use HACMP user management. Using HACMP user management in this environment might cause serious system inconsistencies in the user authentication databases.

3.7.3 HACMP group

During the installation of HACMP, the hacmp group will be created if it does not already exist. During creation, HACMP will simply pick the next available GID for the hacmp group.

Note: If you prefer to control the GID of the hacmp group, we suggest that you create the hacmp group before installing the HACMP filesets.

3.7.4 HACMP IP ports

In addition to the ports identified in the /etc/services file, the following services also require ports, however these ports are selected at random when the processes start. At present there is no way to specify specific ports, just be aware of their presence. Typical ports are shown for illustration, but these ports can be altered if you need to do so.

- #clstrmgr 870/udp
- #clstrmgr 871/udp
- #hatsd 32789/udp
- #clinfo 32790/udp

3.7.5 Planning for HACMP File Collections

HACMP requires that certain files must be identical on all cluster nodes. These files include event scripts, application scripts, certain AIX 5L configuration files, and HACMP configuration files. The HACMP File Collections facility allows you to automatically synchronize these files among cluster nodes and warns you if there are any unexpected results (for example, if one or more files in a collection has been deleted or has a length of zero on one or more cluster nodes).

These file collections can be managed through smit menus. Through smit you can add, delete, and modify file collections to meet your needs.

Default HACMP File Collections

When you install HACMP, it sets up the following default file collections:

- Configuration_Files
- HACMP_Files

Configuration_Files

Configuration_Files is a container for the following essential system files:

- /etc/hosts
- /etc/services
- /etc/snmpd.conf
- /etc/snmpdv3.conf
- /etc/rc.net
- /etc/inetd.conf
- /usr/es/sbin/cluster/netmon.cf

- /usr/es/sbin/cluster/etc/clhosts
- /usr/es/sbin/cluster/etc/rhosts

You can alter the propagation options for this file collection, and you can also add and delete files to/from this file collection.

HACMP_Files

HACMP_Files is a container in which you typically find user-configurable files in the HACMP configuration such as application start/stop scripts, customized events, etc. This File Collection cannot be removed or modified, and the files in this File Collection cannot be removed, modified or added.

Note: For example, when you define an application server to HACMP (start, stop and optional monitoring scripts), HACMP will automatically include these files into the HACMP_Files collection.

3.8 Planning cluster networks

Network configuration is a key component in the cluster design.

In a typical clustering environment, clients access the applications via a TCP/IP network (usually Ethernet) using a service address. This service address will be made highly available by HACMP and will move between communication interfaces on the same network as required. HACMP sends heartbeat packets between all communication interfaces (adapters) on the network to determine the status of the adapter(s) and node(s) and take remedial action(s) as required.

In order to eliminate the TCP/IP network protocol as a single point of failure and prevent cluster partitioning, HACMP also utilizes non-IP point-to-point networks for heartbeating. This assists HACMP with identifying the failure boundary, such as a TCP/IP failure or a node failure.

In this section we will look at each network type and decide on the appropriate network connections and addresses.

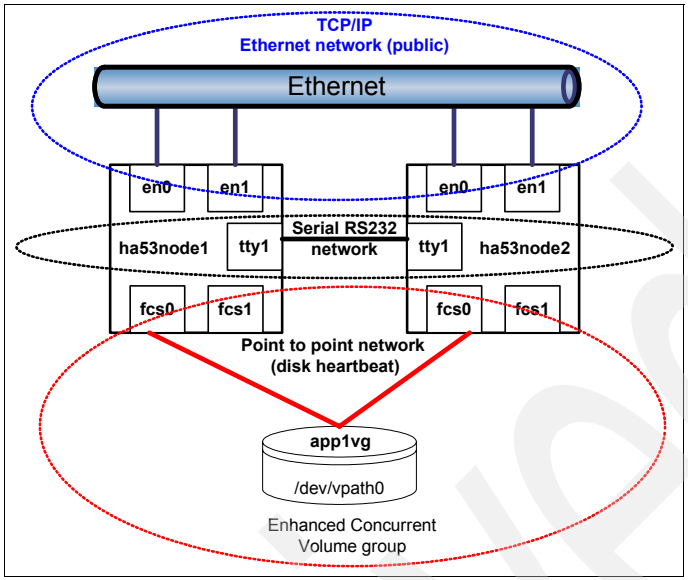

Figure 3-4 HACMP Cluster Networks

Figure 3-4 provides an overview of the networks used in a cluster.

An Ethernet network is used for public access and has multiple adapters connected from each node. This network will hold the base IP addresses, the persistent IP addresses, and the service IP addresses. You can have more than one network however, for simplicity, we are only going to use only one.

A serial RS232 network is shown. This is a point-to-point network with a direct connection using a cable between serial ports on each node.

A disk heartbeat network (also point-to-point) is shown as well. If you are using SAN disks, disk heartbeat is easy to implement because no additional hardware is required. Also, in a multi-path device configuration, using the *vpath device* allows us to take advantage of the capabilities of SDD software, as opposed to using a simple hdisk. That is, a hdisk has only one path to the device while a vpath typically has many paths. Multi-path devices may be configured whenever there are multiple disk adapter in a node, multiple storage adapters, or both.

All network connections are used by HACMP to monitor the status of the network, adapters, and nodes in the cluster.

In our example we will plan for an Ethernet and disk heartbeat network, but not an RS232 network.

3.8.1 Terminology

This section contains a quick recap of the various terminology used in discussions regarding HACMP networking.

IP labels

IP labels are simply names associated with IP addresses resolvable by the system (/etc/hosts, BIND etc.).

Service IP label / address

An IP Address or label over which a service is provided. Typically this is the address used by clients to access an application. It may be bound to a node or shared by nodes and is kept highly available by HACMP.

Persistent IP label / address

A node bound IP Alias that is managed by HACMP. That is, the persistent alias never moves to another node, sometimes referred to as node-bound.

Communication interface

A physical interface that supports the TCP/IP Protocol. For example and ethernet adapter. It is represented by its boot-time or base IP label.

Communication device

A physical device representing an end of a point-to-point non-IP network. For example /dev/tty1 or /dev/vpath0.

Communication adapter

A physical X25 adapter maintained highly available by HACMP.

Network Interface card (NIC)

A Network Interface Card (NIC) is simply a physical adapter used to provide access to an network, for example an ethernet adapter is referred to as a NIC.

3.8.2 General network considerations

This section contains a number of considerations to keep in mind when designing your network configuration.

Supported network types

HACMP allows internode communication with the following TCP/IP-based networks. It should be noted that Ethernet is the most common network in use.

- Ethernet
- Token-Ring
- Fiber Distributed Data Interchange (FDDI)
- ATM and ATM LAN Emulation
- SP Switch1 and SP Switch2
- Etherchannel (or 802.3ad Link Aggregation)

The following TCP/IP-based networks are NOT supported:

- Virtual IP Address (VIPA) facility of AIX 5L
- Serial Optical Channel Converter (SOCC)
- SLIP
- FC Switch (FCS)
- IBM HPS (High Performance Switch)
- 802_ether
- IP V6.

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

- Serial RS232
- Disk heartbeat (over an enhanced concurrent mode disk)
- Target Mode SSA (almost legacy)
- Target Mode SCSI (legacy)

Network connections

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

HACMP also requires all of the communication interfaces for a given cluster network be defined on the same physical network, route packets, and receive responses from each other without interference by any network equipment.

Do not place intelligent switches, routers, or other network equipment that do not transparently pass UDP broadcasts and other packets between all cluster nodes.

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.

Figure 3-5 on page 167 illustrates a physical Ethernet configuration, showing dual ethernet adapters on each node connected across two switches but all configured in the same physical network (VLAN). This is sometimes referred to as being in the same MAC “collision domain”.

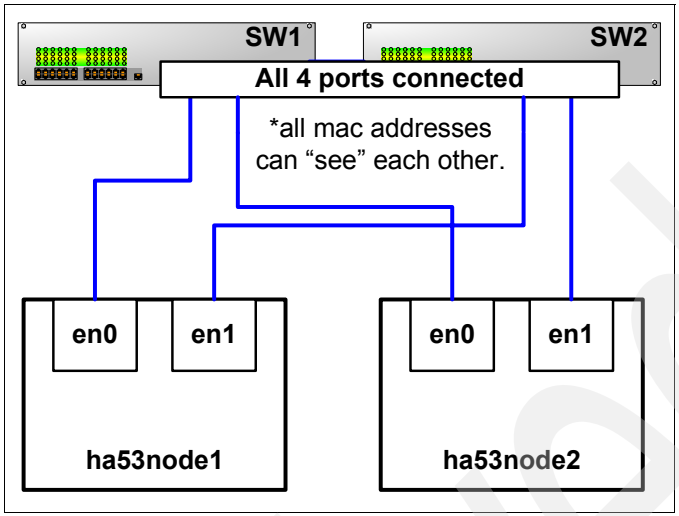

Figure 3-5 Ethernet Switch Connections

Etherchannel

HACMP supports the use of Etherchannel (or Link Aggregation) for connection to an Ethernet network. Etherchannel may be useful if you find that you want to use a number of ethernet adapters for both additional network bandwidth and failover, but also want to keep the HACMP configuration simple. With Etherchannel, you can simply specify the Etherchannel interface as the communication interface, any Ethernet failures, with the exception of the Ethernet network itself, can be handled without HACMP being aware or involved.

EtherChannel is a network port aggregation technology that allows several Ethernet adapters to be bond together to form a single pseudo Ethernet device. For example, ent0 and ent1 can be aggregated into an EtherChannel adapter called ent3; interface en3 would then be configured with an IP address. The system considers these aggregated adapters as one adapter. Therefore, IP is configured over them as over any Ethernet adapter. In addition, all adapters in the EtherChannel are given the same hardware (MAC) address, so they are treated by remote systems as though they were one adapter.

The main benefit of EtherChannel is that it uses the network bandwidth of all of adapters. If an adapter fails, network traffic is automatically sent on the next available adapter without disruption to existing user connections. The adapter is automatically returned to service when it recovers.

In addition to the aggregation feature, a backup adapter can also be assigned. This adapter is configured as part of the Etherchannel but remains inactive until all primary adapters fail. This is referred to as Network Interface Backup (NIB).

Consider the following when planning an Etherchannel,

- The primary (aggregated) links must go to the same switch.
- The Network switch must be configured to identify which ports are Etherchannelled.
- The Network Interface Backup Interface should go to a separate switch.
- Mixing adapters of different speeds is not supported.

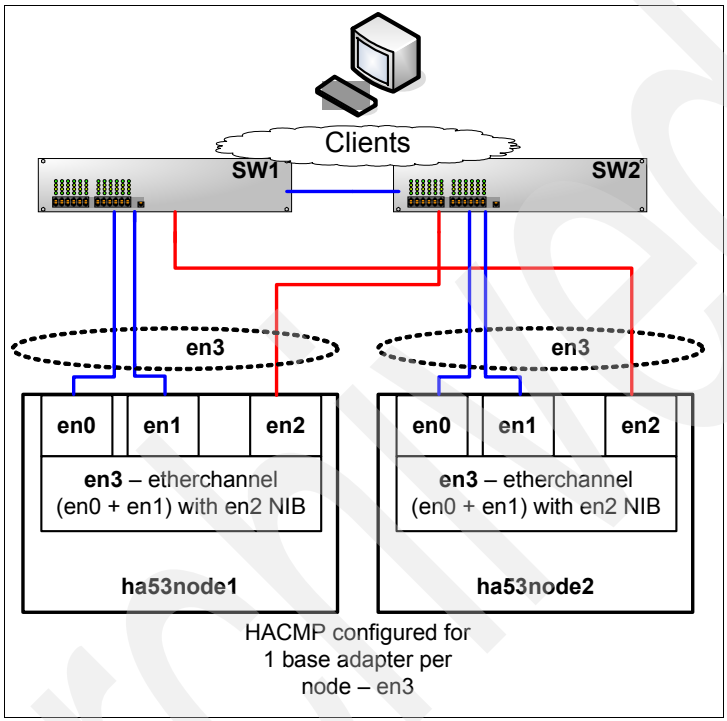

Figure 3-6 Etherchannel with NIB

Figure 3-6 illustrates a simple Etherchannel configuration with 2 adapters going to one switch and the backup going to a separate switch. In this configuration, HACMP would be configured to use only one base adapter per node, en3. In our example, the base, persistent, and service IP addresses would all reside on the same adapter until we failover to another node.

The following Figure 3-7 on page 169 illustrates a slightly more complex configuration showing each node with 2 Etherchannels, one to each switch. In this configuration, HACMP uses the etherchannel adapters as the base adapters.

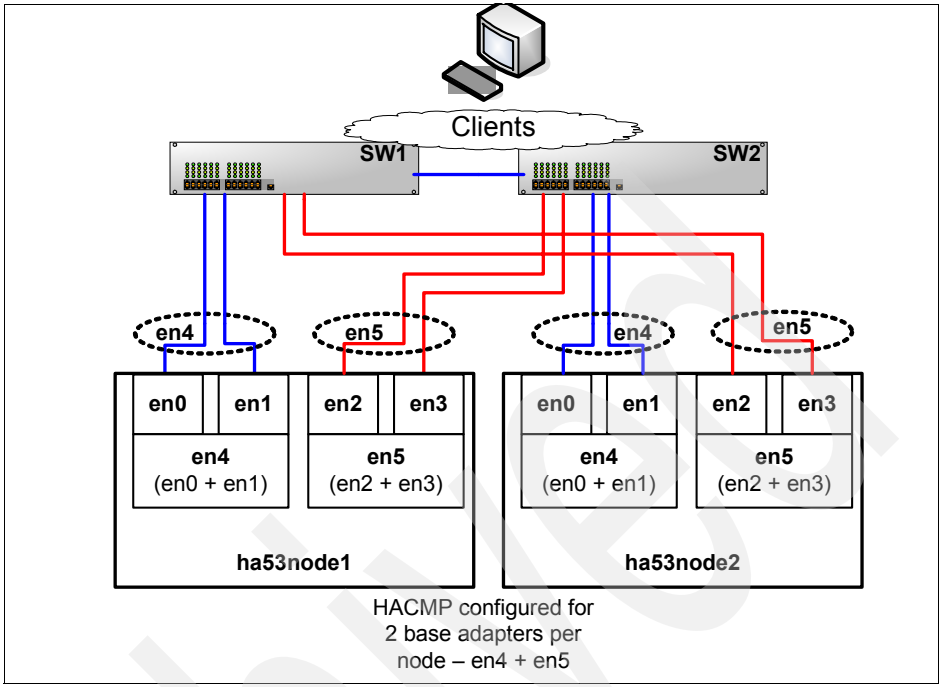

Figure 3-7 Multiple Etherchannel configuration

Note: An etherchannel can consist of 1 to 8 primary adapters with only 1 backup adapter per etherchannel.

Given this, it is quite possible to have only 1 adapter as the primary and 1 adapter as the backup if you want to handle adapter or switch failures without involving HACMP.

Hostnames and node names

Typically, the hostname is the same as the HACMP node name. If you use the Standard configuration path in HACMP, HACMP retrieves the hostname from a node and uses the it as the node name. In the Extended configuration path, you can specify the node name.

In a case where an application requires that the AIX 5L TCP/IP hostname attribute moves with an application to another node at failover, use pre- and post-event scripts to change the hostname to correspond to the service IP label when the resource group that contains this application moves over to another node.

Attention: If you plan to use DLPAR, the AIX hostname, the Cluster node name, and the HMC LPAR name must all match.

We recommend that you try to avoid this situation as it will limit your failover options. For example, you cannot change the hostname for a failover node if there is an application already running on it. In this case, you would need to configure a standby node.

/etc/hosts

An IP address and its associated label (name) must be present in the /etc/hosts file. We recommend that you choose one of the cluster nodes to perform all changes to this file and then use ftp or the HACMP file collections to propagate the /etc/hosts file to the other nodes.

Note: We strongly recommend that you test the direct and reverse name resolution on all nodes in the cluster and the associated Hardware Control Points (HMCs). All these must resolve names identically, otherwise you may run into security issues and other name resolution related problems.

IP aliases

An IP alias is an IP address configured onto a NIC in addition to the base IP address of the NIC. The use of IP aliases is an AIX 5L function that is supported by HACMP. AIX 5L supports multiple IP aliases on a NIC, each on the same or different subnets.

Note: AIX 5L allows IP aliases with *different subnet masks* to be configured for an interface, however this function is not yet supported in HACMP.

Persistent IP addresses (aliases)

A primary reason for using a persistent alias is to provide access the node with HACMP services down. This is a routable address and is available as long as the node is up. You need to configure this alias through HACMP. When HACMP starts, it checks to see if the alias is available, if it isn't, HACMP configures it on an available adapter on the designated network. If the alias is already available, HACMP leaves it alone.

Important: If the persistent IP address exists on the node, it MUST be an alias, NOT the base address of an adapter.

A persistent alias:

- Always stays on the same node (is node-bound)
- Co-exists with other IP labels present on an interface
- Does not require installing an additional physical interface on that node
- Is not part of any resource group.

We recommend that you configure the persistent alias through AIX (“smitty inetalias”) before configuring HACMP. Then use the persistent alias as the communication path to the node, not the base adapters. This allows you the freedom to further change the base IP’s, if required (be sure you check /usr/es/sbin/cluster/etc/rhosts file if you change the base adapter address). Once HACMP is configured, add the persistent alias to the HACMP configuration.

Note: The persistent IP address will be assigned by HACMP on one communication interface which is part of a HACMP defined network.

Figure 3-8 illustrates the concept of the persistent address. Note that this is simply another IP address configured on one of the base interfaces.The **`netstat`** command will show it as an additional IP address on an adapter.

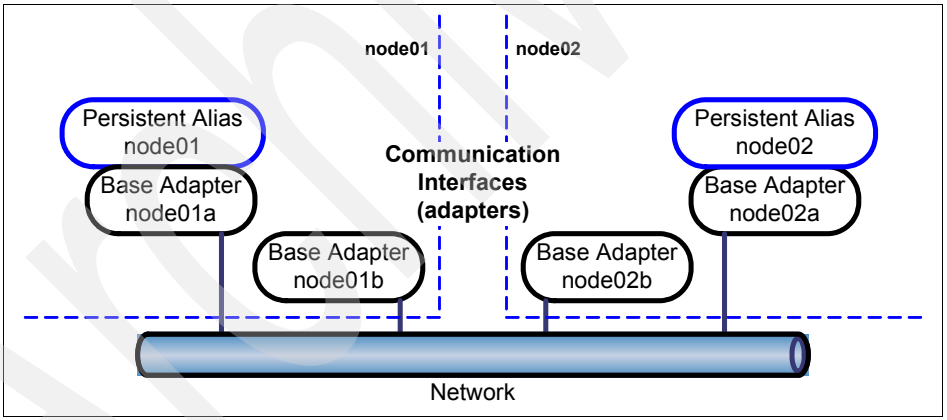

Figure 3-8 Persistent Aliases

Subnetting

Subnet requirements will vary depending upon the configuration chosen (IP Address Takeover -IPAT- via replacement or alias), however in HACMP, the subnet masks for all HACMP defined communication interfaces must be the same.

Fundamentally, for IPAT via replacement,

- Base and service IP addresses on the primary adapter must be on the same subnet.
- All base IP addresses on the secondary adapters must be on separate subnets (different from each other and from the primary adapter).

> **Note:** The base adapter addresses are also known as “boot” IP addresses if you are not using heartbeat monitoring over IP aliases (further explained in this section).

For IPAT via aliases,

- All base IP addresses on a node must be on separate subnets (if heartbeat monitoring over IP aliases is not used).
- All Service IP addresses must be on a separate subnet from any of the base subnets.
- The service IP addresses can all be in the same or different subnets.
- The persistent IP address can be in the same or different subnet from the service IP address.
- If you choose to use heartbeat monitoring over IP aliases, then the base IP addresses can be on the same or different subnets as they are not monitored by HACMP, only the HACMP supplied aliases are monitored.

Default gateway (route) considerations

Depending on your IP network configuration, during the manipulation of the interfaces by HACMP, you may find yourself loosing your default route.

If you tie your default route to one of the base address subnets and that adapter fails, your default route will be lost.

To prevent this situation we recommend you use a persistent address and tie the default route to this subnet. The persistent address will be active as long as the node is active and therefore so will the default route.

If you choose not to do this, then you will have to create a post event script to re-establish the default route if this becomes an issue.

Arp cache updating

During manufacturing, every Network Interface Card (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. Most systems maintain a list that contains recently used IP addresses and their corresponding MAC addresses called an Address Resolution Protocol (ARP) cache. Since HACMP can move IP addresses between NICs, some client ARP cache entries may become inaccurate.

After a cluster event, HACMP nodes and network devices that support promiscuous mode automatically update their ARP caches. Clients and network appliances that do not support promiscuous mode continue to have incorrect entries. You can manage these updates in one of two ways:

- Use alternate hardware addresses. Configure HACMP to move both the IP address and the MAC address (works only with IPAT via replacement).
- Update the ARP cache through use of ping_client_list entries in clinfo.rc.

HACMP in a switched network

If VLANs are used, all interfaces defined to HACMP on a given network must be on the same VLAN. That is, all adapters in the same network are connected to the same physical network and can communicate between each other ("see" each other's MAC addresses).

Note: NOT all adapters have to contain addresses that are routable outside the VLAN. Only the service and persistent addresses need to be routable. The base adapter addresses and any aliases used for heartbeating do not need to be routed outside the VLAN as they are not known to the client side.

Ensure that the switch provides a timely response to ARP requests. For many brands of switches, this means turning **off** the following functions:

- the spanning tree algorithm,
- portfast,
- uplinkfast,
- backbonefast.
- If it is necessary to have spanning tree turned on, then portfast should also be turned on.

Ethernet media speed settings

For Fast Ethernet adapters, as the media speed negotiation may cause problems in certain adapter-switch combinations, we recommend that you not use autonegotiaton, but rather set the media to run at the desired values for speed and duplex.

3.8.3 IP Address takeover planning

IP address Takeover (IPAT) is the mechanism used by HACMP to move service addresses between communication interfaces.

Two methods can be used, IPAT via replacement and IPAT via aliases. Your network configuration will depend on which method you use to have HACMP manipulate the interfaces.

For any new installation, we recommend the use of IP Address Takeover (IPAT) via aliases as this is easy to implement and more flexible than IPAT via replacement. You can have multiple service addresses on the same adapter at any given time, and there are some time savings during failovers because HACMP simply has to add an alias rather then reconfiguring the base IP address of an adapter which is much faster.

Some configurations will require the use of heartbeating over aliases. For example, if both local base adapters are on the same subnet, or if all base adapters on all nodes are on separate subnets.

Each option will be looked at in detail in the following section. Our example will use IPAT via Aliases and heartbeating over aliases.

IP Address Takeover (IPAT) via IP replacement

This is the more traditional way of configuring HACMP networks. HACMP will replace the boot address with the service address when it starts.

For a two node cluster, at least one subnet per communication interface per node is required (same subnet mask for all subnets). For a cluster with multiple communication interfaces per node you need:

- Base and service addresses for the primary communication interface are on the same subnet.
- All secondary communication interfaces must have their base IP addresses on separate subnets (from each other and from the primary one).

IPAT via replacement has the advantage of allowing you to do Hardware Address Takeover (HWAT) in conjuntion with IPAT via replacement. This feature allows you to move the (locally administered) MAC address of the adapter holding the service IP address along with the IP address to a standby adapter. This avoids the need for the client side ARP cache to be updated in case an IP address swap occurs.

Figure 3-9 on page 175 illustrates the status of the network adapters before and after HACMP starts on the nodes. Notice that HACMP replaces the boot/base address with the service address when it starts. Failover is to secondary (a.k.a. standby) adapter(s), where again, the base (standby) address is replaced by the service address. The number of service addresses is restricted to the number of spare adapters defined on the same HACMP network.

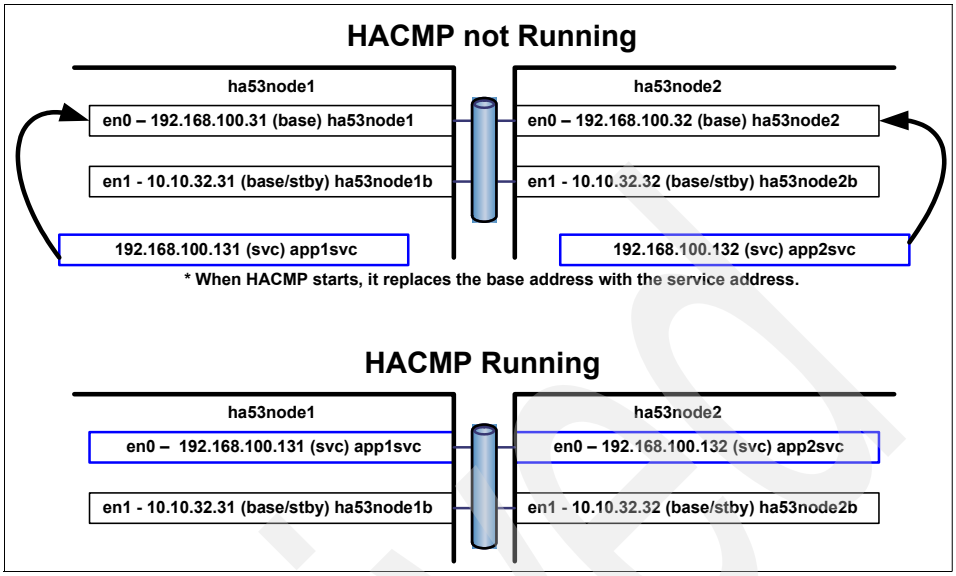

Figure 3-9 IPAT via replacement

IP Address Takeover (IPAT) via aliasing

This is the newer method to assign service addresses and is more flexible than IPAT via replacement. Using IPAT via aliasing, you can have multiple IP addresses assigned to a single communication 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 aliasing is not supported on ATM networks.

When HACMP starts, it simply configures a service alias on top of existing base IP address of an available adapter.

The following requirements must be considered in order to use IPAT via aliases:

- Subnet requirements:
  - Each base adapter must be on a separate subnet to allow for heartbeating. The base addresses do not have to be routable outside of the cluster.

Note: This restriction is lifted if heartbeat monitoring over aliases is used.

  - The service addresses reside on a separate subnet from any of the base subnets. There can be multiple service addresses and they can all be on the same subnet or different ones.
  - The persistent alias can be in the same or different subnet as the service.
  - The subnet masks must all be the same.
- Multiple service labels can coexist as aliases on a given interface.
- Hardware Address Takeover (HWAT) cannot be configured.

We recommend that you use a persistent alias and include it in the same subnet as your default route. This typically means that the persistent address is included in the same subnet as the service addresses. The persistent alias can be used to access the node when HACMP is down as well as overcome the default route issue.

You can configure a distribution preference for the placement of service IP labels that are configured in HACMP V5.3. The placement of the alias is configurable through smit menus as follows:

- Anti-collocation

  This is the default. HACMP distributes all service IP aliases across all available communication interfaces using a “least loaded” selection process.

- Collocation

  HACMP allocates all service IP label aliases on the same communication interface (NIC).

- Anti-collocation with persistent

  HACMP distributes all service IP label aliases across all active communication 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 addresses (persistent and service) must be allocated on the same communication interface. 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.

Figure 3-10 illustrates the status of the network adapters before and after HACMP starts on the nodes. Notice that the base addresses never change. The service and persistent aliases are added to the base adapters by HACMP. The persistent addresses are always available, while the service aliases are added and removed when HACMP starts and stops. Failover is accomplished by moving the service alias to another available communication interface. In our example, only the 192.168.100/24 network is routable outside the cluster.

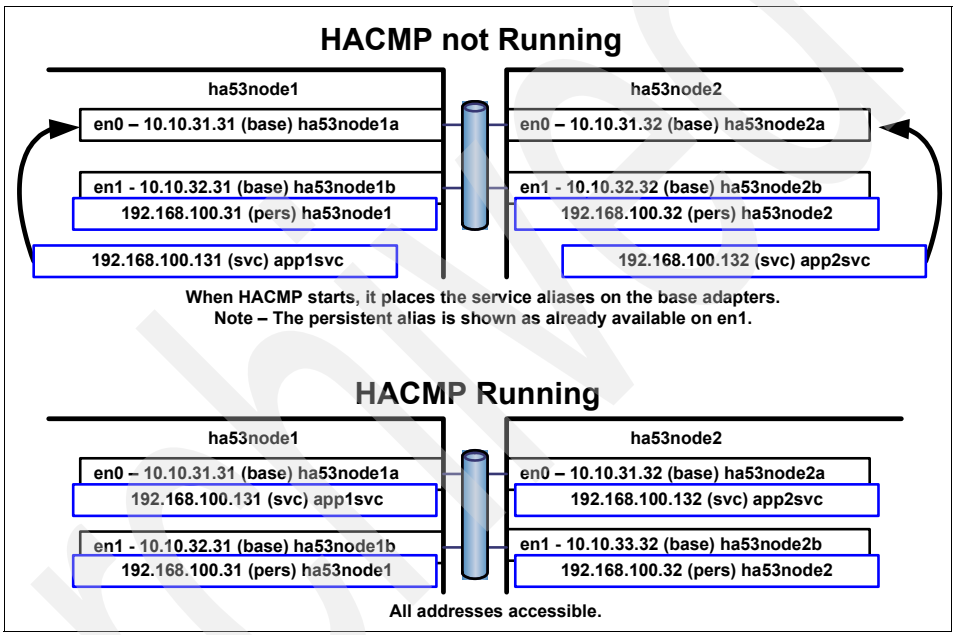

Figure 3-10 IPAT via aliases

3.8.4 Heartbeating over aliases

HACMP requires a separate subnet for each base adapter to be monitored. If you have a common two ethernet adapter per node configuration, you require 2 subnets. If you have 3 adapters, you require 3 subnets. These subnets do not have to be routable outside of the cluster network(s).

To provide the means to monitor these adapters (without changing the base adapter address) and alleviate any subnetting concerns, HACMP provides the heartbeating over alias feature. This method does not require you to make any changes to your existing base addresses. HACMP will simply ignore the base addresses and add its own set of aliases to do the heartbeating.

When using heartbeating over IP aliases, the IP addresses used at boot time can reside on the same subnet or different ones; however, an IP address used at boot time must reside on a subnet that does not include service IP labels. We found that if all addresses (base and service) fall in the same subnet, you will experience routing issues due to the AIX route striping feature.

To set up heartbeat over IP aliases, configure an “IP Address Offset for Heartbeating over IP Aliases” as part of the HACMP network configuration. The IP addresses used for heartbeat monitoring are calculated and assigned by HACMP using this offset value. The subnet mask is the same as that used for the service and non-service addresses.

For example, you could use 1.1.1.1 as the IP Address Offset. If you had a network with two NICs on each node, and a subnet mask of 255.255.255.0, you would end up with the following heartbeat IP aliases:

- node01
  - en0 - 1.1.1.1
  - en1 - 1.1.2.1
- node02
  - en0 - 1.1.1.2
  - en1 - 1.1.2.2

Heartbeat alias IP addresses are added by HACMP when it starts on the node and then removed again when it stops. These IP alias addresses are only used for heartbeat messages. They do not need to be routed and should not be used for any other traffic. The subnet mask is the same as that used for the service and non-service.

Figure 3-11 on page 179 illustrates the status of the network adapters before and after HACMP starts on the nodes. Notice that the base addresses never change. In addition to the service and persistent aliases being added to the base adapters by HACMP, the heartbeat aliases are also added. These are removed when HACMP is stopped, along with the service alias. In our example, only the 192.168.100/24 network is routable outside the cluster.

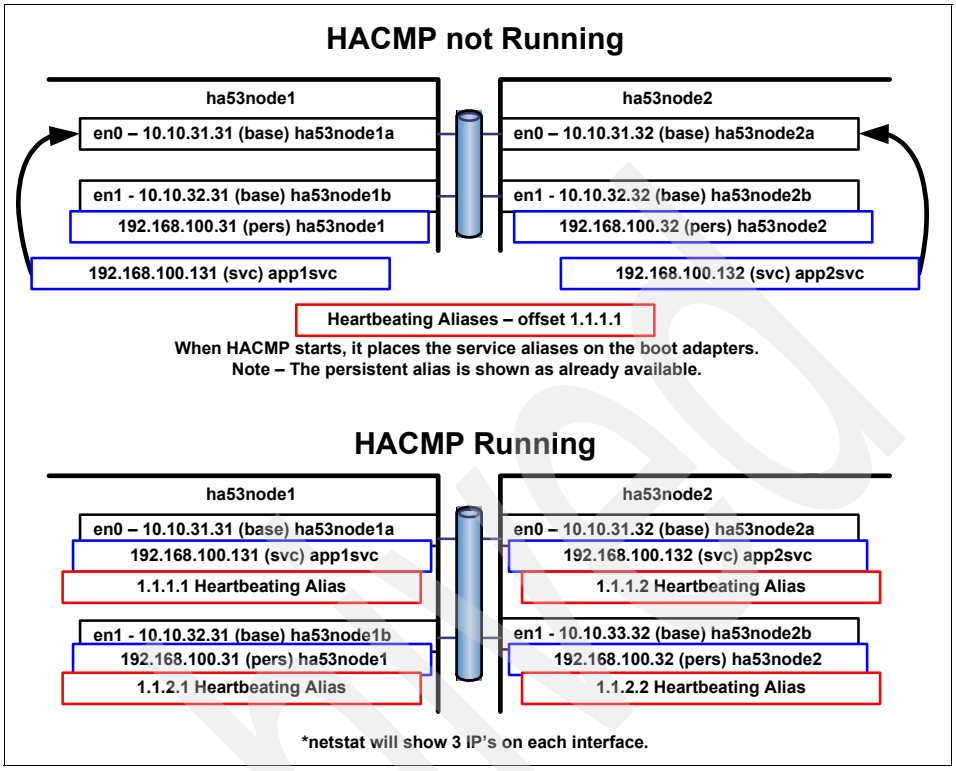

Figure 3-11 Heartbeating over Aliases

A **"`netstat -i`"** command will show three IP addresses on each adapter while HACMP is running.

3.8.5 Non-IP network planning

Point-to-point networks play an important role in ensuring the high availability of the cluster. It is not safe enough to depend on a single TCP/IP network to ensure the cluster availability and prevent cluster partitioning. This is why it is important that you create point-to-point networks. In the case of larger clusters, make sure you create adequate paths between all nodes in the clusters.

The objective of a serial, or point-to-point network topology is to provide enough paths between the cluster nodes in order for RSCT to make a proper diagnosis of the severity of the cluster failure.

Cluster partitioning

Partitioning, also called node isolation (or “split brain”), occurs when an HACMP node stops receiving all heartbeat traffic from another node (on all available networks), and assumes that the other node has failed.

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

The best protection against this situation is to provide more networks, both TCP/IP and point-to-point, in order to allow HACMP to diagnose the severity of the problem. Remember that HACMP (RSCT specifically) sends and receives heartbeat messages across all available networks - the more networks, the better able HACMP is to determine if the problem is with a node or a network.

The following set of four figures illustrate how to add networks to the cluster in order to provide better protection against partitioning.

Figure 3-12 on page 181 illustrates a four node cluster with a single Ethernet connection to each server. Since there is only one network, if any link is lost, part of the cluster will be partitioned. The example shows a break between the two ethernet switches, resulting in the two nodes on the left being partitioned from the two on the right. In this case, problems may arise due to nodes trying to acquire resources from active nodes.

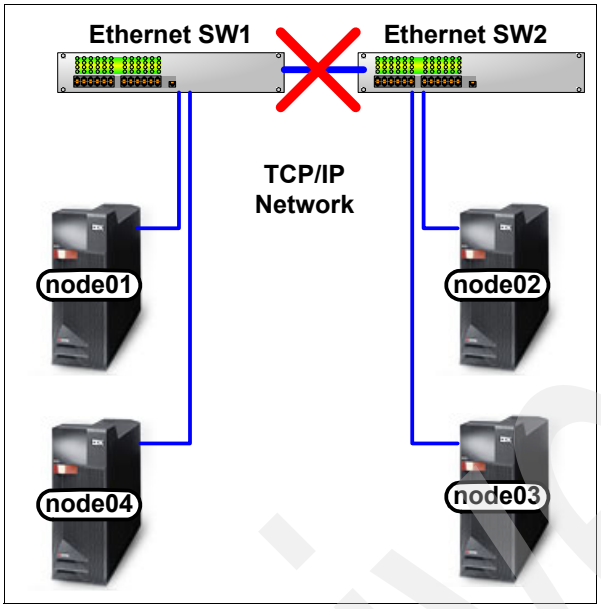

Figure 3-12 Partitioned Cluster

The following Figure 3-13 on page 182, is a bit more realistic, showing dual Ethernet connections from each node. Each Ethernet adapter is connected to a separate switch. In this case, we would require two failures such as both switches failing, or both ethernet connections to a node failing, in order to result in a partitioned cluster. However, the TCP/IP network itself remains a single point of failure.

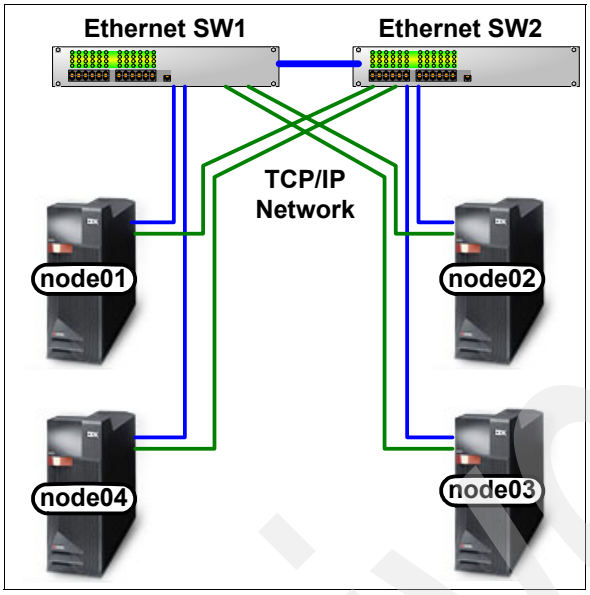

Figure 3-13 Two Ethernet non-partitioned cluster

Figure 3-14 on page 183 is our recommended configuration. We have the dual Ethernet connections going to multiple Ethernet switches, and we add a point-to-point loop network. The loop network has each node connected to its immediate neighbors. One connection can be lost and RSCT will still be able to connect to all surviving nodes. It would take a dual failure on this node, as well as the TCP/IP network to fail before the cluster would end up in a partitioned configuration.

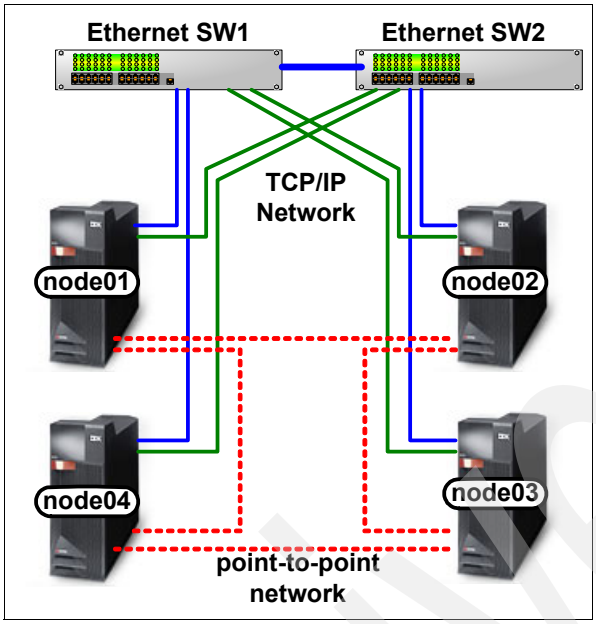

Figure 3-14 Ethernet and point-to-point loop network configuration

For a more robust and reliable configuration, consider implementing a star configuration. In this configuration, in addition to the TCP/IP network, each node is connected to all other cluster nodes by point-to-point networks. This allows for the failure of multiple nodes and RSCT can still communicate between any surviving nodes. This configuration is illustrated in the following diagram, Figure 3-15 on page 184.

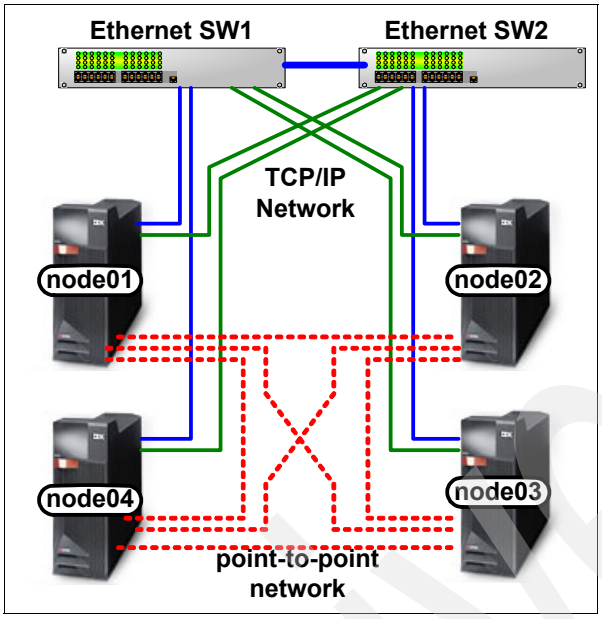

Figure 3-15 Ethernet and point-to-point star network configuration

3.8.6 Planning RS232 serial networks

An RS232 network contains one serial port on each node connected via a serial cable. If you have multiple nodes, you will need multiple serial ports and cables.

If you elect to use an RS232 serial network, consider the following,

- Some pSeries servers have restrictions on using the onboard (integrated) serial ports, some ports are unavailable, and some ports have to be assigned as a group (specially in a LPAR environment).
- If there are no serial ports available, and your planned HACMP configuration for that node uses an RS232 network, you will require a PCI serial adapter per cluster node (LPARs).
- All RS232 networks defined to HACMP are automatically configured to run the serial ports at 38,400Baud. Depending on the length of the serial cable, 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.

The cable needed to connect two serial ports has to be wired as a full NULL modem cable, and is not supplied by default with the hardware. Figure 3-16 on

page 185 illustrates the null modem wiring. The actual cable connectors will depend on your hardware, and most likey be a DB9, DB25, or RJ50 connector.

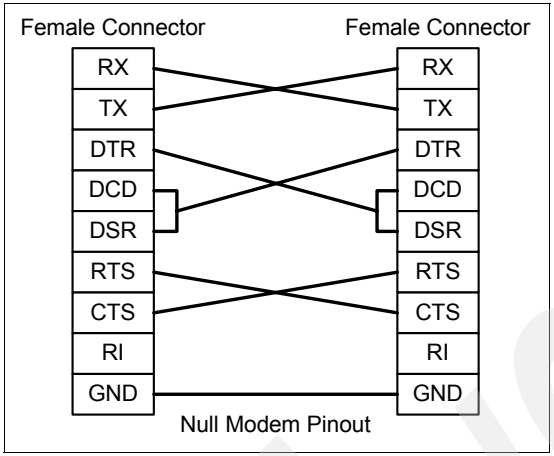

Figure 3-16 NULL modem cable wiring

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

3.8.7 Planning disk heartbeating

Heartbeating over disk provides another type of non-IP point-to-point network for failure detection. In prior versions of HACMP, you could configure non-IP heartbeat over SCSI or SSA disks by configuring Target Mode SCSI (TMSCSI) and Target Mode SSA (TMSSA) point-to-point networks.

Starting with HACMP 5.1, you can also configure a point-to-point, non-IP disk heartbeat connection using any shared disk which is part of an enhanced concurrent mode (ECM) volume group.

In a disk heartbeat 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. While a disk heartbeat network connects only two nodes, in clusters with more than two nodes, multiple disks can be used for heartbeating.

If you are using SAN disk for your shared disk, consider using disk heartbeating for the following reasons,

- You can use any existing shared disk (incl. SAN-attached disks).
- No additional hardware or cables are required.

In order to take advantage of disk heartbeating, you require SAN disks that are accessible by both nodes and are part of an enhanced concurrent volume group.

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 heartbeat network in a cluster contains:

- Two nodes, each with a SAN adapter. A node may be a member of any number of one disk heartbeat networks.
- An enhanced concurrent mode disk. A single disk can only participate 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.
- The disk used for heartbeating should not be overly busy as HACMP expects the writes to occur within certain intervals. If you choose to use a disk that has significant I/O load, increase the value for the timeout parameter for the disk heartbeat network. It is generally recommended that you use a disk that does not experience more than 60 seeks/second.
- When Subsystem Device Driver (device driver for DS8XXX series) 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) in order to take advantage of the multipath software.
- If a shared volume group is mirrored, at least one disk in each mirror should be used for disk heartbeating.
- The recommendation for the disk heartbeat network is to have one LUN (disk) per pair of nodes per disk enclosure.

Figure 3-17 on page 187 illustrates the basic components found in a disk heartbeat network.

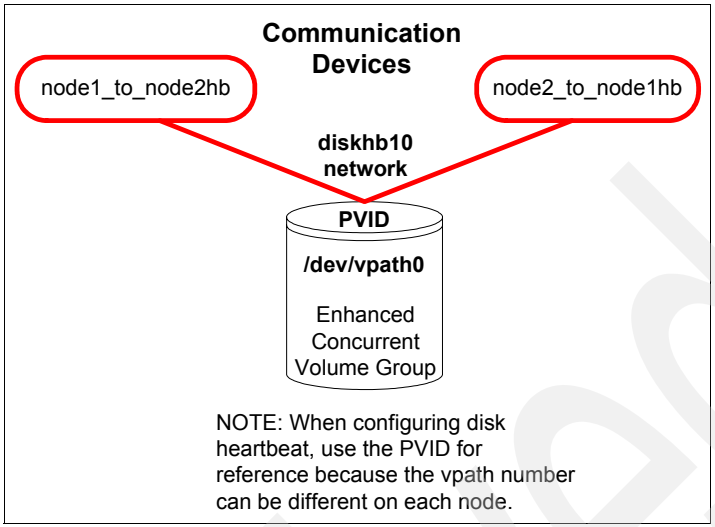

Figure 3-17 Disk Heartbeating Network

Note that the vpath number may appear differently from each node due to the AIX ordering of disks. Therefore check the PVID to ensure you have selected the same disk from either node.

It is generally recommended that you run HACMP discovery and pick the appropriate disks from the picklist provided.

3.8.8 Additional network planning considerations

In addition to configuring the network topology, there are two other topics to be considered during cluster design;

- HACMP with Domain Name Service (DNS) and Network Information Services (NIS)
- HACMP network modules

Each of these are discussed in this section.

HACMP with DNS and NIS

To ensure that cluster events complete successfully and quickly, HACMP disables NIS or DNS hostname resolution during service IP label swapping by setting the following AIX 5L environment variable: NSORDER = local. Therefore, the /etc/hosts file of each cluster node must contain all HACMP defined IP labels for all cluster nodes.

Once the swap completes, DNS access is restored.

We suggest that the following entry be made in the /etc/netsvc.conf file to assure that the /etc/hosts file is read before a DNS lookup is attempted.

- hosts = local, bind4

Network modules

Each supported cluster network has a corresponding RSCT network module (a.k.a. network interface module - NIM) that monitors the heartbeat traffic over the cluster network. The network modules maintain a connection to each other in a cluster through which cluster managers on all cluster nodes send messages to each other (keep-alive messages).

Currently, HACMP passes the corresponding tuning parameters to the RSCT network modules to support communication over the following types of networks:

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

Failure detection rate

The failure detection rate 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
  - For network type Ether the following apply,
    - fast = 10 seconds (5 x 1 x 2)
    - normal = 20 seconds (10 x 1 x 2)
    - slow = 48 seconds (12 x 2 x 2)
- Change the actual components cycle or hbrate.
  - You can use the smit menu “Change a Cluster Network Module using Custom Values”

The preset values are calculated for each type of network to give reasonable results. You may want to consider changing the failure detection rate to:

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

You can find out the network sensitivity (a.k.a. failure detection rate) from the topology services as shown in Example 3-1:

Example 3-1 Network sensitivity for Ether type network

```
p630n01-# lssrc -ls topsvcs
.................. Omitted lines.....................
NIM's PID: 19978
net_ether_01_1 [1]    3    1 S 10.10.31.31     10.10.31.31
net_ether_01_1 [1] en0          0x42d56868      0x42d56872
HB Interval = 1.000 secs. Sensitivity = 10 missed beats
.................. Omitted lines.....................
```

/usr/sbin/cluster/netmon.cf

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 a particular adapter failure. For these situations, RSCT uses the netmon.cf file.

RSCT topology services scans the netmon.cf configuration during cluster startup. 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.

This file can contain up to 30 addresses or labels and the following guidelines apply.

- 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.

3.8.9 Complete the network planning worksheets

The following worksheets capture the necessary network information.

The first worksheet (Table 3-6 on page 190) captures the specifications for the Ethernet network found in our example.

Table 3-6 Cluster Ethernet Networks

HACMP CLUSTER WORKSHEET - PART 4 of 11 CLUSTER ETHERNET NETWORKS					DATE: July 2005
NETWORK NAME	**NETWORK TYPE**	**NETMASK**	**NODE NAMES**	**IPAT VIA IP ALIASES**	**IP Address Offset for Heartbeating over IP Aliases**
ether10	ethernet (public)	255.255.255.0	node01, node02	enable	1.1.1.1
COMMENTS	*NOTE: IP Address offset will add IP Aliases to each Ethernet Interface when HACMP Starts. These aliases are then used for heartbeating and the base adapter addresses are not monitored. Select default Failure Detection rate (ether = normal = 20seconds)				

Next Table 3-7 documents the point-to-point network(s) found in the cluster. Our example will only use a disk heartbeat network but we have included an RS232 network as an example.

Table 3-7 Point to Point Networks

HACMP CLUSTER WORKSHEET - PART 5 of 11 CLUSTER POINT TO POINT AND SERIAL NETWORKS					DATE: July 2005
NETWORK NAME	**NETWORK TYPE**	**NODE NAMES**	**Device**	**INTERFACE NAME**	**ADAPTER LABEL**
serial10	serial (private)	node01, node02	NA	/dev/tty0 /dev/tty0	node01_tty1 node02_tty1
diskhb10	diskhb	node01, node02	vpath0 vpath0	NA	node1_to_node2hb node2_to_node1hb
COMMENTS	*NOTE: RS232, target mode SCSI, target mode SSA, and disk heartbeat links do not use the TCP/IP protocol and do not require a netmask or an IP address. The serial10 network will not be configured - only the diskhb10 network.				

Now that the networks have been recorded, document the interfaces and IP addresses used by HACMP, as shown in Table 3-8 on page 191:

Table 3-8 Cluster Communication Interfaces and IP addresses

HACMP CLUSTER WORKSHEET - PART 6 of 11 INTERFACES AND IP ADDRESSES					**DATE: July 2005**
node01					
IP Label	**IP Alias Dist. Preference**	**NETWORK INTERFACE**	**NETWORK NAME**	**INTERFACE FUNCTION**	**IP ADDRESS /MASK**
node01a	NA	en0	ether10	base (non-service)	10.10.31.31 255.255.255.0
node01b	NA	en1	ether10	base (non-service)	10.10.32.31 255.255.255.0
ha53node1	Anti-collocation (default)	NA	ether10	persistent	192.168.100.31 255.255.255.0
app1svc	Anti-collocation (default)	NA	ether10	service	192.168.100.131 255.255.255.0
node02					
IP Label	**IP AliasDist. Preference**	**NETWORK INTERFACE**	**NETWORK NAME**	**INTERFACE FUNCTION**	**IP ADDRESS /MASK**
node02a	NA	en0	ether10	base (non-service)	10.10.31.32 255.255.255.0
node02b	NA	en1	ether10	base (non-service)	10.10.32.32 255.255.255.0
ha53node2	Anti-collocation (default)	NA	ether10	persistent	192.168.100.32 255.255.255.0
app2svc	Anti-collocation (default)	NA	ether10	service	192.168.100.132 255.255.255.0
COMMENTS	Each Node contains 2 Base adapters, each in their own subnet. Each node also contains a Persistent (Node Bound) address and a Service Address. IPAT via Aliases is used as well as Heartbeat over Aliases (starting range = 1.1.1.1)				

3.9 Planning storage requirements

When planning cluster storage you must consider the following:

- Physical disks
  - Ensure any disk solution is highly available. This can be accomplished through mirroring, RAID, and redundant hardware.
  - Internal disks. Typically this is the location of rootvg.
  - External disks. This must be the location of the application data.
- LVM components
  - All shared storage has unique logical volume and filesystem names
  - Major numbers are unique.
  - Is mirroring of data required?

3.9.1 Internal disks

Internal node disk typically contains rootvg and perhaps the application binaries. We recommend that the internal disk be mirrored for higher availability, that is, plan to prevent a node failover due to a simple internal disk failure.

3.9.2 Shared disks

Application data resides on the external disk in order to be accessible by all required nodes. These are referred to as the shared disks.

Important: All shared disks must be "zoned" to any cluster nodes requiring access to the specific volumes. That is, the shared disk must be able to be varied on and accessed by any node that has to run a specific application.

We recommend that you verify the shared volume group can be manually varied on to each node before asking HACMP to manage it.

In an HACMP cluster, shared disks are 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, in order to restore service to clients, another cluster node in the resource group node list acquires ownership of the shared disks and restarts applications.

Typically, depending on the number of disks in a resource group and the disk takeover method, a takeover may take from 30 to 300 seconds.

HACMP supports the following IBM 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.

OEM disk may be supported but you have to validate support for these disk subsystems with the equipment manufacturer.

When working with a shared volume group:

- Do not include an internal disk in a shared volume group, because it will not be accessible by other nodes.
- Do not activate (vary on) the shared volume groups in an HACMP cluster at system boot. Use cluster event scripts to do this. Ensure that the automatic varyon attribute in the AIX 5L ODM is set to No for shared volume groups listed within a resource group. You can use the cluster verification utility to correct this attribute for you.

Important: If you define a volume group to HACMP, do not manage it manually on any node outside of HACMP while HACMP is running. 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.

3.9.3 Sample disk configuration

Figure 3-18 on page 194 illustrates the various disk configurations that may be used in a cluster. Note that the internal disks are mirrored (rootvg), and there are multiple fiber channel adapters (HBA's) in each node, each connected to a separate SAN switch (for redundancy).

PVID's should be used to verify the appropriate disk as it is quite possible that the vpath number may be different on each node due to AIX device ordering.

The shared disks are shown zoned to both nodes. Zoning is done through the SAN Storage Manager software.

Important: You should always follow the storage configuration rules (zoning, LUN masking) appropriate for your environment. This is especially important when HACMP nodes share the same SAN and storage subsystems with other servers.

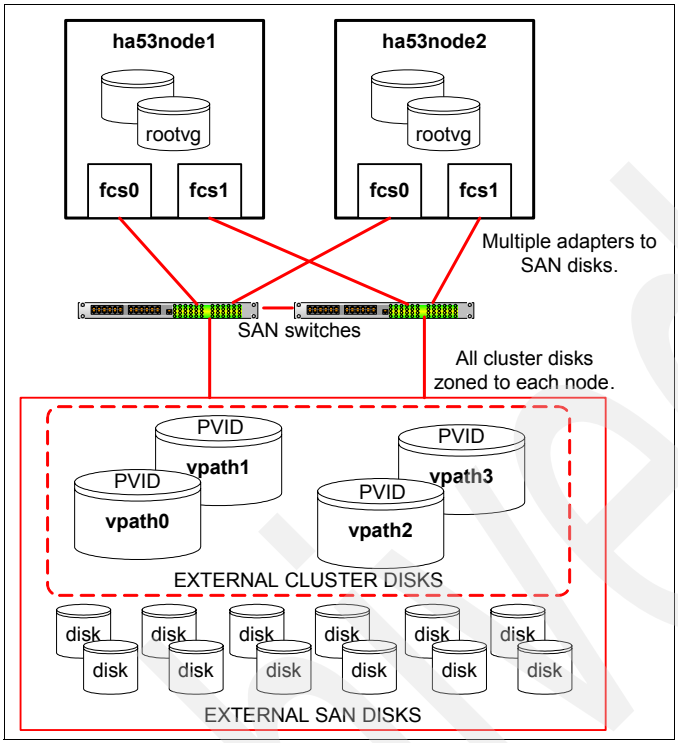

Figure 3-18 Physical Disk Configuration

3.9.4 Enhanced Concurrent Mode (ECM) volume groups

Any disk supported by HACMP for attachment to multiple nodes can be used to create an enhanced concurrent mode volume group, and used in either concurrent or non-concurrent environments:

- **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 then 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, the LVM allows access to the volume group on all nodes. However, it restricts the higher-level connections, such as JFS mounts and NFS mounts, on all nodes, and allows them only on the node that currently owns the volume group in HACMP.

ECM volume groups are available in AIX 5.x as a replacement for classic HACMP concurrent option. In new versions of AIX (5.2 and up) you can only create enhanced concurrent VGs. Although you can still use the “old” 32bit concurrent VGs, you must analyze the restrictions that come with maintaining such VGs in your cluster. For more detail see *High Availability Cluster Multi-Processing for AIX 5L Planning and Installation Guide*, SC23-4861-06.

Note: Although you can define enhanced concurrent mode volume groups, this DOES NOT necessarily mean that you are going to use them for concurrent access, i.e., you can still define and use these VGs as normal shared file system access. However, you must NOT define file systems on VGs that are intended for concurrent access.

3.9.5 Shared logical volumes

Planning for shared logical volumes is all about data availability. Making your data highly available throught the use of mirroring or RAID is a key requirement. Remember that HACMP relies on LVM and storage mechanisms (RAID) to protect against disk failures, therefore it is imperative that you make the disk infrastructure highly available.

Consider the following guidelines when planning shared LVM components:

- Logical volume copies or RAID arrays protect against loss of data from physical disk failure.
- All operating system files should reside in the root volume group (rootvg) and all user data should reside outside that group.
- 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 for the volume groups, you need to use the super strict disk allocation policy for mirrored physical volumes.
- When LVM mirroring, 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 (for example, using a quorum buster disk/LUN) or disable quorum.
- Keep in mind the cluster configurations that you have designed. A node whose resources are not taken over should not own critical volume groups.
- Ensure regular backups are scheduled.

Once you have established a highly available disk infrastructure you must consider the following items as well when designing your shared volume groups.

- All shared volume groups have unique logical volume and filesystem names. This includes the jfs/jfs2 log files.
- Do *not* use in-line logs with shared JFS2 file systems.
- Major numbers for each volume group are unique (especially if you plan to use NFS).

Figure 3-19 outlines the basic components found in the external storage. Notice all logical volumes and filesystem names are unique, as is the major number for each volume group. The data is made highly available through the use of SAN disk and redundant paths to the devices.

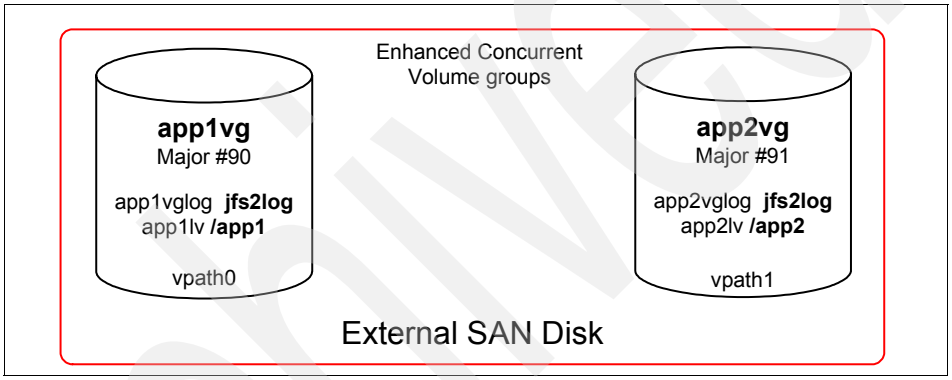

Figure 3-19 External Disk

3.9.6 Fast disk takeover

HACMP automatically detects node failures and initiates a disk takeover as part of a resource group takeover process. Traditional disk takover process implies breaking the SCSI (or SSA) disk reservation before varying on the volume group on the takeover node. This may be a lengthy process, especially if there are numerous disks in that volume group.

Starting in HACMP 5.1, as AIX provides support for enhanced concurrent volume groups, it is possible to reduce the disk takeover (and thus, the resource group) time by using the fast disk takeover option. If a VG used for shared file system access has been defined in enhanced concurrent mode, HACMP automatically detects this and varies on the VG on all nodes part of that RG. This eliminates the need for breaking the hardware disk reservation in case of a takeover. Fast disk takeover requires:

- AIX 5L v.5.2 and up

- HACMP 5.1 and up with the bos.clvm.enh AIX fileset 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, you can convert these volume groups to enhanced concurrent volume groups after upgrading your HACMP software.

The actual fast disk takeover 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.

3.9.7 Complete the storage planning worksheets

The following worksheets contain the required information about the shared volume groups. Combined, they will give you a good idea of the shared disk configuration.

Document the shared Volume Groups and physical disks in the following Table 3-9:

Table 3-9 Shared Disks

HACMP CLUSTER WORKSHEET - PART 7 of 11 SHARED DISKS					**DATE: July 2005**
node01			**node02**		
VGNAME	**VPATHS**	**HDISK**	**HDISK**	**VPATHS**	**VGNAME**
app1vg	vpath0	hdisk0, hdisk1, hdisk2, hdisk3	hdisk0, hdisk1, hdisk2, hdisk3	vpath0	
	vpath1	hdisk4, hdisk5, hdisk6, hdisk7	hdisk4, hdisk5, hdisk6, hdisk7	vpath1	app2vg
COMMENTS	All disks are seen by both nodes. app1vg normally resides on node01, app2vg normally resides on node02.				

Record the shared Volume Group details as shown in Table 3-10 on page 198:

Table 3-10 Shared Volume Groups

HACMP CLUSTER WORKSHEET - PART 8 of 11 SHARED VOLUME GROUPS (NON-CONCURRENT)		DATE: July 2005
RESOURCE GROUP	**VOLUME GROUP 1**	**VOLUME GROUP 1**
C10RG1	app1vg Major Number = 90 log = app1vglog Logical Volume 1 = app1lv1 Filesystem 1 = /app1 (20GB)	NA
C10RG2	app2vg Major Number = 91 log = app2vglog Logical Volume 1 = app2lv1 Filesystem 1 = /app2 (20GB)	NA
COMMENTS	Create the shared Volume Group on the first node and then import on the second node. #importvg -y app1vg -V 90 vpath0 (may have to make the pv available with chdev -l vpath0 -a pv=yes) #chvg -an app1vg (set vg to not auto vary on) #mount /app1 (ensure the filesystem mounts) #umount /app1 #varyoffvg app1vg (leave VG offline in order for HACMP to manage)	

3.10 Application planning

Virtually any application that runs on a standalone AIX server can be integrated into an HACMP cluster, as they are not aware of the underlining HACMP functionality. That is, HACMP basically starts and stops them.

When planning for an application to be highly available, be sure you understand the resources required by the application and the location of these resources within the cluster. This will enable you to provide a solution that allows them to be handled correctly by HACMP if a node fails.

You must thoroughly understand how the application behaves in a single-node and multi-node environment. We recommend that as part of preparing the application for HACMP, you test the execution of the application manually on both nodes before turning it over to HACMP to manage. Do not make assumptions about the application's behavior under failover conditions

Note: The key prerequisite to making an application highly available is that it first must run correctly in standalone mode on each node it may reside.

We recommend that you ensure the application runs on all required nodes properly before configuring it to be managed by HACMP.

You need to analyze and address the following aspects:

- application code - binaries, scripts, links, configuration files etc.
- environment variables - any environment variable that needs to be passed to the application for proper execution.
- application data
- networking setup - IP addresses, hostname
- application licensing
- application defined system users

When planning an application to be protected in an HACMP cluster, consider the following,

- The application is compatible with the version of AIX used.
- The application is compatible with the shared storage solution as this is where its data will reside.
- 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 server.
- 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, such as a database and appserver, HACMP provides an easy to use smit menu to allow you to specify this relationship.
- Write robust scripts to both start and stop the application on the cluster nodes. The startup script must be able to recover the application from an abnormal termination. Ensure that they run properly in a single-node environment before including in HACMP.
- 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.
- Verify that the application uses a proprietary locking mechanism if you need concurrent access.

3.10.1 Application servers

In HACMP an application server is simply a set of scripts used to start and stop an application.

Configure your application server by creating a name to be used by HACMP and associating a start and a stop script.

Once you have created an application server, you associate it with a resource group (RG). HACMP then uses this information to control the application.

3.10.2 Application monitoring

HACMP can monitor your application by one of two methods,

- 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 such as a script that you define.

Starting with HACMP 5.2 you can have multiple monitors for an application.

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

- You can configure multiple application monitors, each with unique names, and associate them with one or more application servers.
- The monitor method must be an executable program, such as 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.
- By default, the monitoring method logs messages to the /tmp/clappmond.application_monitor_name.monitor.log file. Also, by default, 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.

Important: As the monitoring process is time sensitive, ALWAYS 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

3.10.3 Availability analysis tool

The application availability analysis tool can be used 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.

3.10.4 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 verification ensures the correctness and consistency of certain aspects of your Fast Connect Services, or Workload Manager configuration.

3.10.5 Complete the application planning worksheets

The following worksheets capture the required information for each application.

Update the application worksheet to include all required information, as shown in Table 3-11:

Table 3-11 Application Worksheet

HACMP CLUSTER WORKSHEET - PART 9 of 11 **APPLICATION WORKSHEET**				**DATE: July 2005**
APP1				
ITEM	**DIRECTORY**	**FILESYSTEM**	**LOCATION**	**SHARING**
EXECUTABLE FILES	/app1/bin	/app1	SAN Storage	Shared
CONFIGURATION FILES	/app1/conf	/app1	SAN Storage	Shared
DATA FILES	/app1/data	/app1	SAN Storage	Shared
LOG FILES	/app1/logs	/app1	SAN Storage	Shared

HACMP CLUSTER WORKSHEET - PART 9 of 11 APPLICATION WORKSHEET				DATE: July 2005
START SCRIPT	/cluster/local/app1/start.sh	/	rootvg	Not Shared (must reside on both nodes)
STOP SCRIPT	/cluster/local/app1/stop.sh	/	rootvg	Not Shared (must reside on both nodes)
FAILOVER STRATEGY	Failover to node02.			
NORMAL START COMMANDS AND PROCEDURES	Ensure that the APP1 server is running.			
VERIFICATION COMMANDS AND PROCEDURES	Run the following command and ensure APP1 is active. If not, send notification.			
NORMAL START COMMANDS AND PROCEDURES	Ensure APP1 stops properly.			
NODE REINTEGRATION	Must be reintegrated during scheduled maintenance window to minimize client disruption.			
APP2				
ITEM	**DIRECTORY**	**FILESYSTEM**	**LOCATION**	**SHARING**
EXECUTABLE FILES	/app2/bin	/app2	SAN Storage	Shared
CONFIGURATION FILES	/app2/conf	/app2	SAN Storage	Shared
DATA FILES	/app2/data	/app2	SAN Storage	Shared
LOG FILES	/app2/logs	/app2	SAN Storage	Shared
START SCRIPT	/cluster/local/app2/start.sh	/	rootvg	Not Shared (must reside on both nodes)
STOP SCRIPT	/cluster/local/app2/stop.sh	/	rootvg	Not Shared (must reside on both nodes)

HACMP CLUSTER WORKSHEET - PART 9 of 11 APPLICATION WORKSHEET				DATE: July 2005
FAILOVER STRATEGY	Failover to node01.			
NORMAL START COMMANDS AND PROCEDURES	Ensure that the APP2 server is running.			
VERIFICATION COMMANDS AND PROCEDURES	Run the following command and ensure APP2 is active. If not, send notification.			
NORMAL START COMMANDS AND PROCEDURES	Ensure APP2 stops properly.			
NODE REINTEGRATION	Must be reintegrated during scheduled maintenance window to minimize client disruption.			
COMMENTS	Summary of Applications.			

Update the application monitoring worksheet to include all the information required for the application monitoring tools (Table 3-12).

Table 3-12 Application Monitoring Worksheet

HACMP CLUSTER WORKSHEET - PART 10 of 11 APPLICATION MONITORING	DATE: July 2005
APP1	
Can this Application Be Monitored with Process Monitor?	Yes
Processes to Monitor	app1
Process Owner	root
Instance Count	1
Stabilization Interval	30
Restart Count	3
Restart Interval	95

HACMP CLUSTER WORKSHEET - PART 10 of 11 APPLICATION MONITORING	DATE: July 2005
Action on Application Failure	Fallover
Notify Method	/usr/es/sbin/cluster/events/notify_app1
Cleanup Method	/usr/es/sbin/cluster/events/stop_app1
Restart Method	/usr/es/sbin/cluster/events/start_app1
APP2	
Can this Application Be Monitored with Process Monitor?	Yes
Processes to Monitor	app2
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_app2
Cleanup Method	/usr/es/sbin/cluster/events/stop_app2
Restart Method	/usr/es/sbin/cluster/events/start_app2

3.11 Planning for resource groups

HACMP manages resources through the use of resource groups.

Each resource group is handled as a unit that may contain the following types of resources: IP labels, applications, filesystems and volume groups. Each resource group has preferences that define when and how it will be acquired or released. 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 a reintegrating node.

The following rules and restrictions apply to resources and resource groups

- To be kept highly available by HACMP, a cluster resource must be part of a resource group. If you want a resource to be kept separate, you can 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.
- We recommend that you put the application server along with the resources it requires in the same resource group (unless otherwise needed).
- If you include the same node in participating nodelists for more than one resource group, make sure that the node can sustain all resource groups simultaneously.

Figure 3-20 simplifies the relationship between applications, volume groups, and service addresses and how they all combine to form resource groups. The grouping is such that we have an application with its shared disk storage and service IP lable all together in a single resource group. By doing so, everything needed by the application will be available when HACMP activates the resource group.

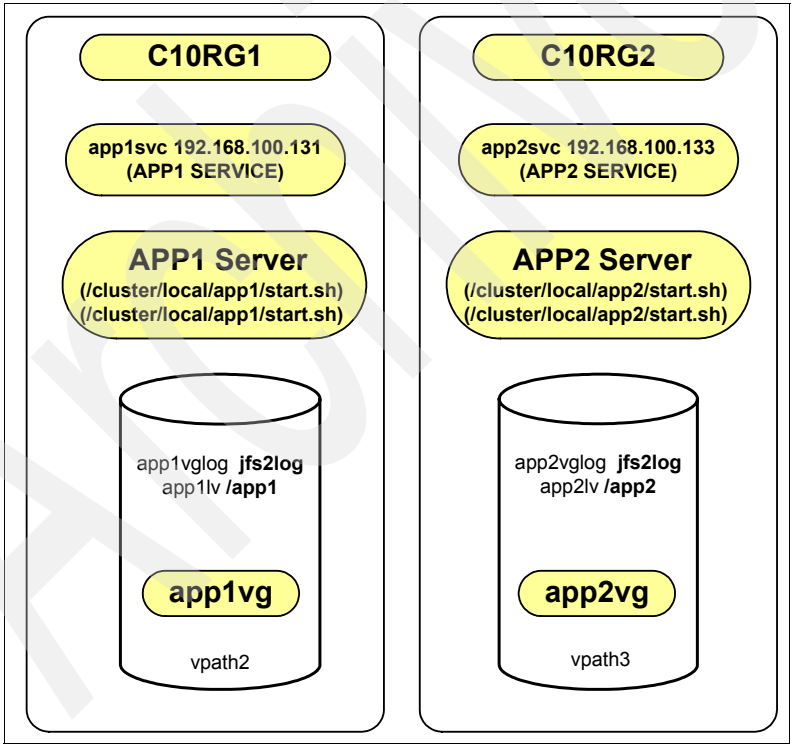

Figure 3-20 Resource Groups

Once you have decided what components are to be grouped into a resource group, you must plan the behavior of the resource group.

The following Table 3-13 summarizes the basic startup, fallover, and fallback behaviors you can configure for resource groups in HACMP 5.3.

Table 3-13 Resource group behavior

Startup Behavior	Fallover Behavior	Fallback Behavior
Online on home node only (OHNO) for the resource group.	-Fallover to next priority node in the list -Fallover using Dynamic Node Priority	-Never fall back -Fall back to higher priority node in the list
Online using node distribution policy.	-Fallover to next priority node in the list -Fallover using Dynamic Node Priority	Never fall back
Online on first available node (OFAN).	-Fallover to next priority node in the list -Fallover using Dynamic Node Priority -Bring offline (on error node only)	-Never fall back -Fall back to higher priority node in the list
Online on all available nodes.	Bring offline (on error node only)	Never fall back

3.11.1 Resource group attributes

Startup settling time

Settling time only applies to Online on First Available Node (OFAN) resource groups and lets HACMP wait for a set amount of time before activating a resource group. After the settling time, HACMP will then activate the resource group on the highest available priority node. Use this attribute to ensure that resource groups do not bounce among nodes, as nodes with increasing priority for the resource group are brought online.

If the node that is starting is a home node for this resource group, the settling time period is skipped and HACMP immediately attempts to acquire the resource group on this node.

Attention: This is a cluster wide setting and will be set for all OFAN volume groups.

Dynamic Node Priority (DNP) policy

Setting a dynamic node priority policy allows you select the takeover node based on specific performance criteria. This uses an 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.

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

The delayed fallback timer lets a resource group fall back to a higher priority node at a time that you specify. The resource group that has a delayed fallback timer configured and that currently resides on a non-home node falls back to the higher priority node at the specified time.

Resource group dependencies

HACMP 5.3 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/child dependencies so that related applications in different resource groups are processed in the proper order
- Location dependencies so that certain applications in different resource groups stay online together on a node or on a site, or stay online on different nodes.

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

Dependencies between resource groups offer a predictable and reliable way of building clusters with multi-tiered applications.

IPAT method 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.

There is no IPAT with concurrent resource groups.

A resource group may include multiple service IP labels. When a resource group configured with IPAT via IP Aliases is moved, all service labels in the resource group are moved as aliases to an available network interface.

Planning for Workload Manager (WLM)

WLM allows users to set targets 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.

3.11.2 Complete the planning worksheet

The resource group worksheet captures all the required planning information for the resource groups (see Table 3-14).

Table 3-14 Resource Groups Worksheets

HACMP CLUSTER WORKSHEET - PART 11 of 11 RESOURCE GROUPS)		**DATE: July 2005**
RESOURCE NAME	**C10RG1**	**C10RG2**
Inter-Site Management Policy	ignore	ignore
Participating Node Names	node01 node02	node02 node01
Startup Policy	Online on Home Node Only (OHNO)	Online on Home Node Only (OHNO)

HACMP CLUSTER WORKSHEET - PART 11 of 11 RESOURCE GROUPS)		DATE: July 2005
Fallover Policy	Fallover to Next Priority Node in List (FONP)	Fallover to Next Priority Node in List (FONP)
Fallback Policy	Fallback to Higher Priority Node (FBHP)	Fallback to Higher Priority Node (FBHP)
Delayed Fallback Timer		
Settling Time		
Runtime Policies		
Dynamic Node Priority Policy		
Processing Order (Parallel, Serial, or Customized)		
Service IP Label	app1svc	app2svc
Application Servers	app1	app2
Volume Groups	app1vg	app2vg
Filesystems	/app1	/app2
Filesystem Consistency Check	fsck	fsck
Filesystems Recovery Method	sequential	sequential
Filesystems or Directories to Export		
Filesystems or Directories to NFS mount		
Network for NFS mount	ether10	ether10
Primary Workload Manager Class		
Auto Import Volume Groups	false	false
Filesystems Mounted before IP Configured.	false	false
COMMENTS	Overview of the 2 Resource Groups.	

3.12 Detailed cluster design

Pulling it all together, using the information collected during the preceding cluster planning and documented in the Planning Worksheets, we can now build an easy to read, detailed cluster diagram. Figure 3-21 on page 211 contains a detailed cluster diagram for our example. This diagram is useful to use as an aid when configuring the cluster and diagnosing problems.

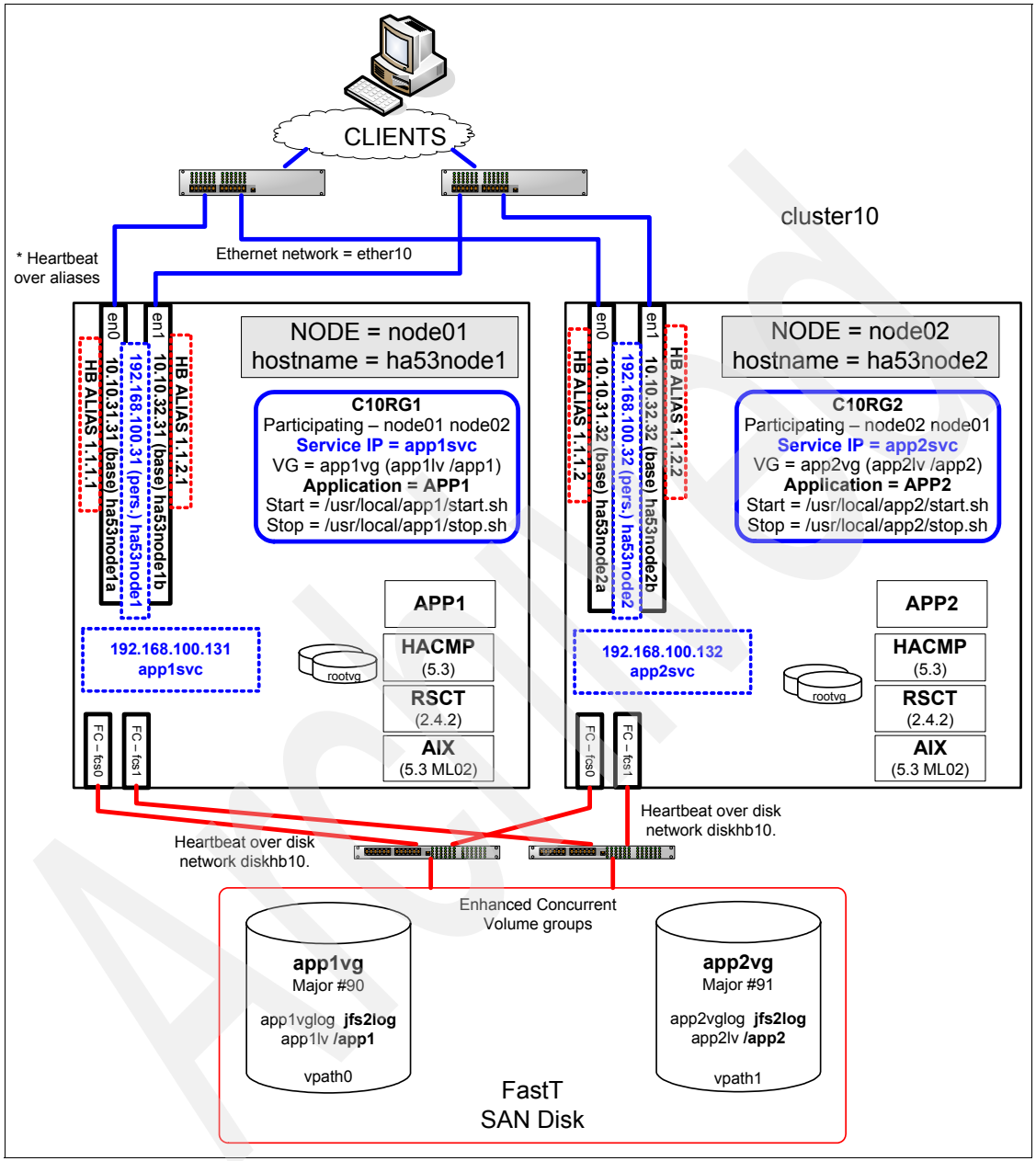

Figure 3-21 Detailed cluster design

3.13 Develop a cluster test plan

Just as important to planning and configuring your HACMP cluster is developing an appropriate test plan to validate the cluster under failure situations. That is, will the cluster handle failures as expected. You must test, or validate, the cluster recovery before the cluster becomes part of your production environment.

3.13.1 Custom test plan

As with previous releases of HACMP, you should develop a local set of tests to verify the integrity of the cluster. This typically involves unplugging network cables, downing interfaces, and shutting down cluster nodes to verify cluster recovery. This is still a useful exercise as you have the opportunity to simulate failures and watch the cluster behavior. If something does not respond correctly, or as expected, stop the tests and investigate the problem. Once all tests complete successfully, the cluster can be moved to production.

Table 3-15 outlines a sample test plan that can be used to test our cluster.

Table 3-15 Sample Test Plan

Cluster Test Plan			
Test #	**Test Description**	**Comments**	**Results**
1	Start HACMP on node01.	node01 starts and acquires the C10RG1 resource group.	
2	Start HACMP on node02.	node02 starts and acquires the C10RG2 resource group.	
3	Perform a graceful stop without takeover on node01.	Resource Group C10RG1 goes offline.	
4	Start HACMP on node01.	node01 starts and acquires the C10RG1 resource group.	
5	Perform a graceful stop with takeover on node01.	Resource Group C10RG1 moves to node02.	
6	Start HACMP on node01	node01 starts and requires the C10RG1 resource group.	
7	Fail (unplug) the service interface on node01.	The service IP moves to the second base adapter.	

Cluster Test Plan			
Test #	**Test Description**	**Comments**	**Results**
8	Reconnect the service interface on node01.	The service IP remains on the second base adapter.	
9	Fail (unplug) the service interface on node01 (now on the second adapter).	The service IP (and persistent) moves to the first base adapter.	
10	On node01 issue a “halt -q” to force down the operating system.	node01 halts - resource group C10RG1 moves to node02.	
11	Reboot node01 and restart HACMP.	node01 reboots. Once HACMP starts node01 requires C10RG1	
12	Perform a graceful stop without takeover on node02.	Resource Group C10RG2 goes offline.	
13	Start HACMP on node02.	node02 starts and acquires the C10RG2 resource group.	
14	Perform a graceful stop with takeover on node02.	Resource Group C10RG2 moves to node01.	
15	Start HACMP on node02	node02 starts and requires the C10RG2 resource group.	
16	Fail (unplug) the service interface on node02.	The service IP moves to the second base adapter.	
17	Reconnect the service interface on node02.	The service IP remains on the second base adapter.	
18	Fail (unplug) the service interface on node02 (now on the second adapter).	The service IP (and persistent) moves to the first base adapter.	
19	On node02 issue a “halt -q” to force down the operating system.	node02 halts - resource group C10RG2 moves to node01.	
20	Reboot node02 and restart HACMP.	node02 reboots. Once HACMP starts node02 requires C10RG2	

3.13.2 Cluster Test Tool

To ease with the testing of the cluster, HACMP 5.2 and 5.3 include a Cluster Test Tool to help you test the functionality of a cluster before it becomes part of your production environment.

The Cluster Test Tool only runs on a cluster with HACMP 5.2 or later where the configuration has been verified and synchronized. The tool can run in two fashions,

- Automated testing
  - Use the automated test procedure (a predefined set of tests) supplied with the tool to perform basic cluster testing on any cluster. No setup is required. You simply run the test from SMIT and view test results from the Cluster Test Tool log file.
- Custom testing
  - If you are an experienced HACMP administrator and want to tailor cluster testing to your environment, you can create custom tests that can be run from SMIT. After you set up your custom test environment, you run the test procedure from SMIT and view test results in the Cluster Test Tool log file.

The Cluster Test Tool uses the HACMP Cluster Communications daemon to communicate between cluster nodes to protect the security of your HACMP cluster.

Automated testing

This test tool provides an automated method to quickly test the functionality of the cluster. It typically takes 30 to 60 minutes to run, depending on the cluster complexity, and will perform the following tests. You must have root access to perform these tests.

General cluster topology tests

The Cluster Test Tool runs the general topology tests in the following order:

- Start cluster services on all available nodes
- Stop cluster services gracefully on a node
- Restart cluster services on the node that was stopped
- Stop cluster services with takeover on another node
- Restart cluster services on the node that was stopped
- Forces cluster services to stop on another node
- Restart cluster services on the node that was stopped.

Resource group tests on non-concurrent resource groups

If the cluster includes one or more non-concurrent resource groups, the tool runs each of the following tests in the following order for each resource group:

- Bring a local network down on a node to produce a resource group fallover
- Recover the previously failed network
- Bring an application server down and recover from the application failure.

Resource group test on concurrent resource groups

If the cluster includes one or more resource groups that have a startup management policy of online on all available nodes (OAAN), the tool runs one test that brings an application server down and recovers from the application failure.

Catastrophic failure test

The tool runs one catastrophic failure test that stops the cluster manager on a randomly selected node that currently has at least one active resource group.

Note: If the tool terminates the cluster manager on the control node, you may need to reboot this node.

Running automated tests

As a general recommendation, it is useful to periodically validate your cluster configuration. For this purpose, two automation tools are available:

- Cluster automated test tool

  This is used to actually test the cluster

- Automatic cluster configuration verification

  This is a tool that periodically checks and advertises any configuration changes so the cluster administrator can take corrective actions (synchronize and re-test the cluster)

These tools can be used to implement a standard validation procedure. A manual test is not necessary after the initial test has been completed. However, as the automated cluster tool may take disruptive actions, you must schedule the usage of this tool in a periodic maintenance window.

The Cluster Test Tool runs a specified set of tests and randomly selects the nodes, networks, resource groups, and so forth for testing. The tool tests different cluster components during the course of the testing.

Important: Before you start running an automated test ensure that the cluster is not in service in a production environment

To run the automated test procedure:

- Enter **`smit hacmp`**

- In SMIT, select Initialization and Standard Configuration > HACMP Cluster Test Tool and press Enter.
- The “Are you sure” message appears. If you press Enter again, the automated test plan runs.

3.14 Developing an HACMP installation plan

Now that you’ve planned the configuration of the cluster and documented the design, prepare for your installation.

If you are implementing HACMP on existing servers, be sure to schedule an adequate maintenance window to allow for the installation, configuration, and testing of the cluster.

If this is a new installation, allow time to configure and test the basic cluster. Once the cluster is configured and tested, you can integrate the required applications during a scheduled maintenance window.

Referring back to Figure 3-1 on page 138, you can see that there is a preparation step before installing HACMP. This step is intended to ensure the infrastructure is ready for HACMP. This typically involves using your planning worksheets and cluster diagram to prepare the nodes for an HACMP 5.3 install.

- Ensure the node software and operating system prerequisites are installed
- Ensure network connectivity is properly configured.
- Ensure the Shared Disks are properly configured.
- Ensure that the chosen applications are able to run on either node.

The preparation step can take some time depending on the complexity of your environment and the number of resource groups and nodes that to be used. Take your time preparing the environment as there is no purpose in trying to install HACMP in an environment that is not ready. You will simply spend your time troubleshooting a poor installation. Remember, a well configured cluster is built upon solid infrastructure.

Once the cluster planning is complete and environment is prepared, the nodes are ready for HACMP to be installed.

The installation of HACMP code is straightforward. If using the install CDROM, simply use smit to install the required filesets. If using a software repository, you can NFS mount the directory and use smit to install from this directory.

Ensure you are licensed for any features you install, such as the Smart Assist and HACMP/XD.

Once you have installed the required filesets on all cluster nodes, use the planning worksheets to configure your cluster. Here you have a few tools available to use to configure the cluster.

- You can configure WebSmit at this point and use it to configure the cluster.
- For a two node cluster, you can use the Java based 2-node assistant.
- You can use an ascii screen and smit to perform the configuration.

You have a number of choices available to help with the configuration of the cluster. The next chapter will discuss each option in detail but basically you can,

- Use the 2-node assistant to configure the cluster. This will configure a basic two node cluster with a single resource group.
- Use the HACMP Standard Configuration smit panels to configure the cluster is an standard format.
- Use the HACMP Extended Configuration smit panels to manually configure the cluster.
- Use the *.haw file generated by the Online Planning Worksheets to apply to the cluster.
- Apply a cluster snapshot to configure the cluster.

Note: We recommend that when configuring the cluster you start by configuring the cluster (network) topology. Once the cluster topology is configured, verify and synchronize the cluster before moving forward with the resources (shared volume groups, service IP addresses, and applications).

Once the topology has been successfully verified and synchronized, you should start the cluster services and verify if everything is running as expected.

This will allow you to identify any networking issues before moving forward to continue configuring the cluster resources.

Once you have configured, verified, and synchronized the cluster, execute the automated cluster test tool to validate cluster functionality. Review the results of the test tool and if it was successful, execute any custom tests you want to perform to perform further verification.

Verify any error notification you have included.

After successful testing, take a mksysb of each node and a cluster snapshot from one of the cluster nodes.

The cluster should be ready for production.

Standard change and problem management processes now apply to maintain application availability.

3.15 Backup the cluster configuration

The primary tool for backing up the HACMP cluster is the cluster snapshot. Although the Online Planning Worksheet Cluster Definition file also captures the cluster configuration, it is less comprehensive as it does not include ODM entries.

The primary information saved in a cluster snapshot is the data stored in the HACMP Configuration Database classes (such as HACMPcluster, HACMPnode, HACMPnetwork, HACMPdaemons).This is the information used to recreate the cluster configuration when a cluster snapshot is applied.

The cluster snapshot does not save any user-customized scripts, applications, or other non-HACMP configuration parameters. For example, the names of application servers and the locations of their start and stop scripts are stored in the HACMPserver Configuration Database object class. However, the scripts themselves as well as any applications they may call are not saved.

The cluster snapshot utility stores the data it saves in two separate files:

- ODM Data File (.odm)
  - This file contains all the data stored in the HACMP Configuration Database object classes for the cluster. This file is given a user-defined basename with the.odm file extension. Because the Configuration Database information is largely the same on every cluster node, the cluster snapshot saves the values from only one node.
- Cluster State Information File (.info)
  - This file contains the output from standard AIX 5L and HACMP. This file is given the same user-defined basename with the.info file extension. By default, this file no longer contains cluster log information. Note that you can specify in SMIT that HACMP collect cluster logs in this file when cluster snapshot is created.

For a complete backup, take a mksysb of each cluster node as per standard practices. Pick one node to perform a cluster snapshot and save the snapshot to a safe location for disaster recovery purposes.

If you can, take the snapshot before taking the mksysb of the node so that it is included in the system backup.

Important: You can take a snapshot from any node in the cluster, even if HACMP is down. However, you can only apply a snapshot to a cluster if all nodes are running the same version of HACMP and all are available (HACMP can communicate between the nodes using clcomdES).

3.16 Documenting the cluster

It is important to document the cluster configuration in order to effectively manage the cluster. From managing cluster changes, to troubleshooting problems, a well documented cluster will result in better change control and quicker problem resolution.

We suggest that you maintain an accurate cluster diagram which can be used for change and problem management.

In addition, HACMP provides the tools to easily gather the Cluster configuration data through the use of the Online Planning Worksheets (OLPW).

This section discusses how to product a cluster definition file through smit and then use it to create a cluster configuration report via the OLPW tool. The resulting report is in html format and can be viewed using a Web browser.

The basic steps in creating a cluster report are,

- Export a cluster definition file from one of the cluster nodes using smit.
  - This will typically be saved as a *.haw file.
  - If you are using the OLPW on your workstation, ftp the definition file to your workstation.
- Use the OLPW to open an existing definition file.
- Use the OLPW to create a configuration report.
  - This will create an *.html file.
- Use your Web browser to view the file.
  - We recommend that you save the file to another server or workstation for disaster recovery purposes.

3.16.1 Exporting a cluster definition file using SMIT

You can create a cluster definition file from an active HACMP cluster and then open this file using the Online Planning Worksheets application.

To create a cluster definition file from SMIT:

- Enter smit hacmp
- Select Extended Configuration
  - Export Definition File for Online Planning Worksheets and press Enter (see Example 3-2).

Example 3-2 Export Definition File from SMIT

```
                        Extended Configuration

Move cursor to desired item and press Enter.
```

```
Discover HACMP-related Information from Configured Nodes
Extended Topology Configuration
Extended Resource Configuration
Extended Cluster Service Settings
Extended Event Configuration
Extended Performance Tuning Parameters Configuration
Security and Users Configuration
Snapshot Configuration
Export Definition File for Online Planning Worksheets

Extended Verification and Synchronization
HACMP Cluster Test Tool
```

- 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/cluster.haw**.
  - Cluster Notes
    - Any additional comments that pertains to your cluster. The information that you enter here will display in the Cluster Notes panel in Online Planning Worksheets.
- Open the cluster definition file in Online Planning Worksheets.

3.16.2 Create a cluster definition file from a snapshot using SMIT

You can also create a cluster definition file from an HACMP cluster snapshot and then open this file using the Online Planning Worksheets application.

To create a cluster definition file from a snapshot using SMIT:

- Enter smit hacmp
- Select Extended Configuration
  - Extended Configuration
    - Snapshot Configuration ->
    - Convert Existing Snapshot For Online Planning Worksheets
    - Select your previously created snapshot.
- Once the Cluster definition file is created, open it in Online Planning Worksheets.

3.16.3 Creating a configuration report

A configuration report enables you to record information about the state of your cluster configuration in an HTML format.

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
- Applications
- Networks
- NFS exports
- IP labels
- Application servers
- Global network
- Application monitors
- Sites
- Pagers or cell phones
- Disks
- Remote notifications
- Resource groups
- Tape resources
- Volume groups
- Resource group runtime policies
- Logical volumes
- Node summary
- File collections
- Cluster verification
- Cross-site LVM Mirroring

To create a configuration definition report:

- Select File > Create Report.
- 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.

Figure 3-22 shows a screen capture of the generated report. You can scroll down the report page for further details.

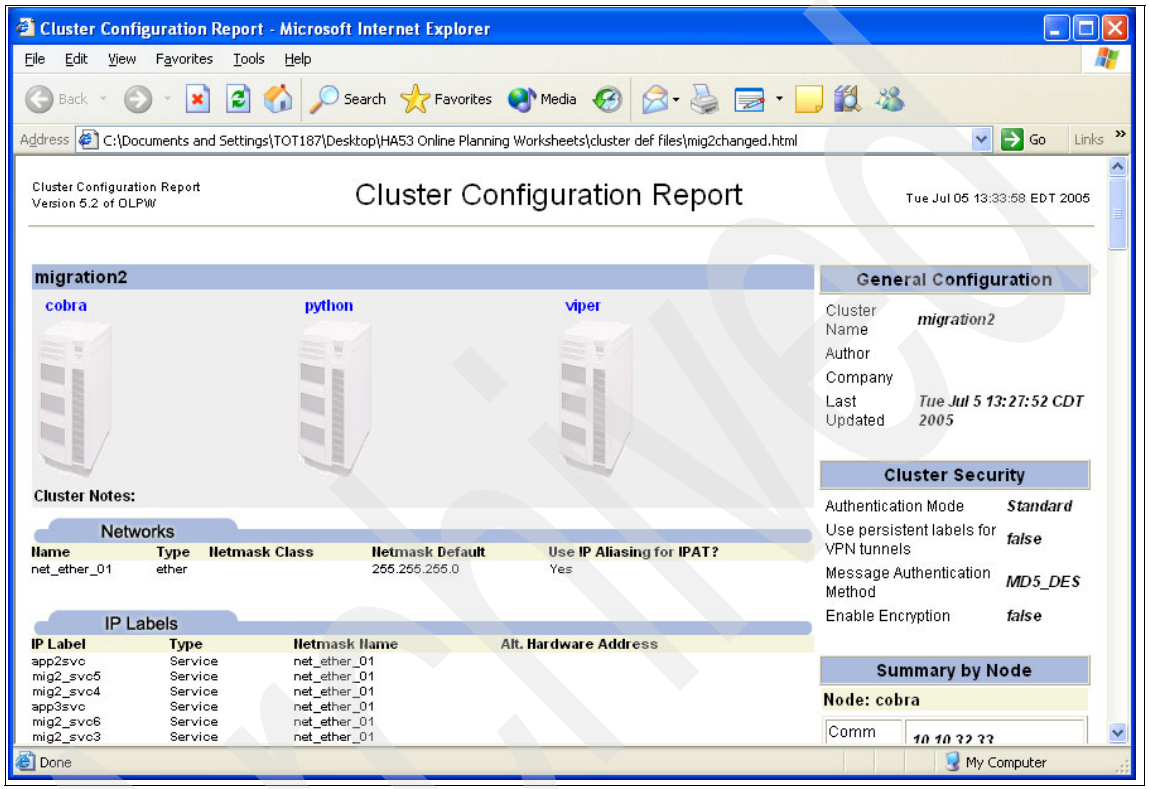

Figure 3-22 Sample Configuration Report

3.17 Change and problem management

Once the cluster is up and running, the job of managing change and problems begins.

Effective Change and Problem management processes are imperative to maintaining cluster availability. To be effective, you must have a current cluster configuration handy. You can use the OLPW tool to create an html version of the configuration and, as we also suggest, a current cluster diagram.

Any changes to the cluster should be fully investigated as to their effect on the the cluster functionality. Even changes that do not directly affect HACMP, such as the addition of additional non-hacmp workload, may affect the cluster. The changes should be planned, scheduled, documented, and the cluster tested once the change has been made.

To ease with implementing changes to the cluster, HACMP provides the Cluster Single Point of Control (C-SPOC) smit menus. Whenever possible, the C-SPOC menus should be used to make changes. Using C-SPOC, you can make changes from one node and the change will be propagated to the other cluster nodes.

Problems with the cluster should be quickly investigated and corrected. Since HACMPs primary job is to mask any errors from applications, it is quite possible that unless you have monitoring tools in place, you may be unaware of a failover. Ensure you make use of error notification to notify the appropriate staff of failures.

3.18 Planning tools

This section discusses the three main planning tools in greater detail. A sample cluster diagram and paper planning worksheets are provided.

3.18.1 Cluster diagram

Diagramming the HACMP cluster allows for a clear understanding of the behavior of the cluster and helps identify single points of failure. A sample two-node cluster diagram is provided in Figure 3-23 on page 224.

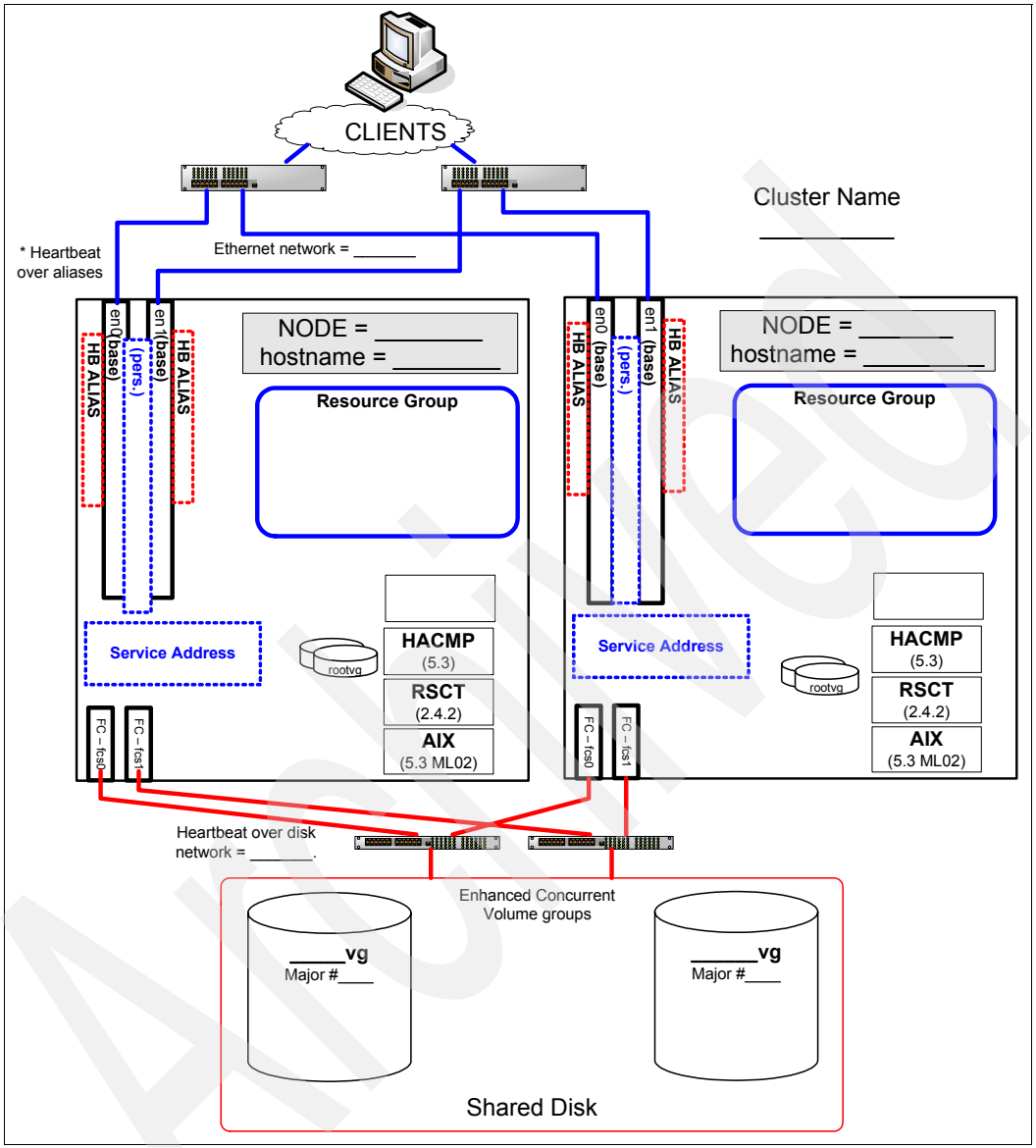

Figure 3-23 Sample Cluster Diagram

3.18.2 Online Planning Worksheets

The Online Planning Worksheets (OLPW) is a java-based version of the paper worksheets. Using this application, you can either import HACMP configuration information from an existing cluster and edit it as needed, or you can enter all

configuration information manually. The application saves your information as a cluster definition file that you can apply to configure your cluster. The application also validates your data to ensure that all required information has been entered.

The following steps illustrate an effective use of the OLPW tool, from planning, to implementing, to documenting your cluster.

- Prepare for cluster planning by familiarizing yourself with the HACMP concepts and your environment.
- Run the Online Planning Worksheet program on your workstation from the HACMP CD-ROM.
- Complete the Online Planning Worksheets and create a cluster definition file.
- Install HACMP on the cluster nodes.
- Copy the cluster definition file to one of the cluster nodes.
- Apply the cluster definition file to the cluster using the cl_opsconfig command.
- Take a cluster snapshot.
- Now document the cluster by generating a report which will create an html file that you can use for systems management.

Attention: The Online Planning Worksheets is a tool for configuring and recording a cluster. You still must have a good understanding of planning a cluster before using the tool.

Running the OLPW from CD-ROM

Running the Application from the CD-ROM on Windows

- Insert the HACMP CD-ROM into the appropriate drive.
- Navigate to the olpw/worksheets.bat file and run it.
- Note: Do not close the command window used to launch the application. Closing this window closes the application.

Running the Application from the CD-ROM on AIX 5L

On an AIX 5L system, to run the Online Planning Worksheets application from the HACMP CD-ROM:

- Ensure that the path to the JRE is set in your PATH environment variable as follows.
  - AIX 5.3 - /usr/java141/bin
  - AIX 5.2 - /usr/java131/bin
  - AIX 5.1 - /usr/java130/bin
- Mount the installation medium by using the following command:
  - mount -v cdrfs -p -r cd_location mount_directory
  - Where cd_location is the location of the CD, and mount_directory is the name of the directory to be mounted. For example:mount -v cdrfs -p -r /dev/cd0 /mnt
- Run the application by executing the following command:

– java -jar mount_directory/olpw/worksheets.jar
– where mount_directory is the directory specified above

Installing OLPW

Installing the Application on an AIX 5L System

You install the Online Planning Worksheets application from the HACMP software installation medium. The installable image for the application is:

- cluster.es.worksheets

The Online Planning Worksheets application is installed in the /usr/es/sbin/cluster/worksheets directory.

Running the OLPW application from an AIX 5L GUI

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.

Installing the OLPW application on a Windows system

To install the Online Planning Worksheets application on a Microsoft® Windows system:

- Install the Online Planning Worksheets from the HACMP installation medium on an AIX 5L system.
- Copy the worksheets.bat and worksheets.jar files to a directory of your choice on your Windows system.

Note: If you copy the files via FTP, be sure to specify the ASCII mode for the.bat file and binary for.jar.

Running the Application from a Windows Installation

Execute the worksheets.bat command from the command line, or from a file manager GUI double-click the worksheet.jar icon.

Understanding the main window

When you open the Online Planning Worksheets application, its main window displays, as shown in Figure 3-24 on page 227:

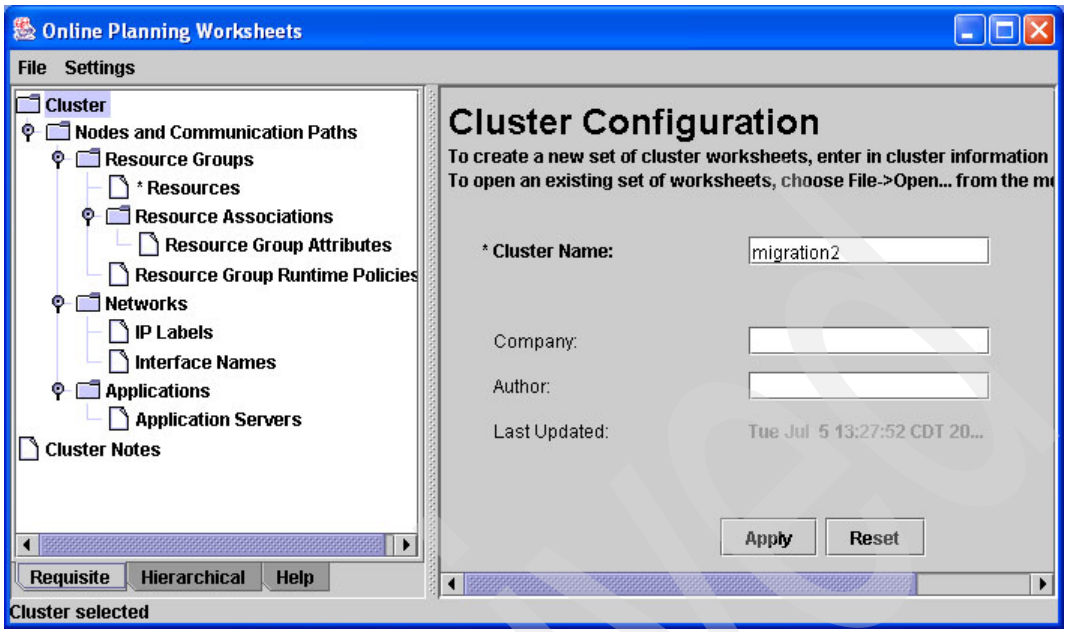

Figure 3-24 Online Planning Worksheets Main Menu

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. Your configuration information is entered on the right pane.

Creating a new cluster definition file

When you start planning your cluster, you can either fill in all information by hand or OLPW can read in your cluster configuration information and then fill in the rest of the information by hand.

To create a cluster definition file:

- Enter all data by hand using your planning information

  Or,

- Read in configuration information directly from your HACMP cluster as follows:
  - Use the smit menus to create a definition file.
  - Within the OLPW tool, Select File > Import HACMP Definition and select the cluster definition file.

  - 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.
  - Enter any additional data by hand
- Save the newly created 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 5.2 and 5.3.
- .xml
  - This extension may be used. It is supported in HACMP 5.3.
- .ws
  - This file format is supported in HACMP 5.1.0.1. For backwards compatibility,.ws files can be opened in HACMP 5.3; however, they must be saved with either the .xml or .haw file extension.

To open a cluster definition file, select File > Open. Only one cluster definition file can be opened at a time.

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 notes about the cluster configuration

As you plan your configuration, you can add notes to the cluster definition file

To add cluster notes:

- In the left pane in either the Requisite® view or the Hierarchical view, select Cluster Notes.
- In the Cluster Notes panel, enter the information you want to save.
- Push the Apply button.

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.

When you save a file, OLPW automatically validates the cluster definition unless automatic validation has been turned off.

Applying worksheet data to your HACMP cluster

After you complete the configuration panels in the OLPW application, you can save the file, and then apply it to a cluster node. If you use the Online Planning Worksheets application on a Windows system, you must first copy the cluster definition 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.
- 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:

- From the Online Planning Worksheets application, validate your cluster definition file.
- Create a report to document your cluster configuration.
- Save the file and exit the application. If your cluster configuration file resides on a Windows system, copy the file to an HACMP node.
- 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, performs a synchronization, and then a verify. 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
```

3.18.3 Paper planning worksheets

Detailed Paper Planning Worksheets are found in Appendix A of the **HACMP 5.3 Planning and Installation Guide.**

We have found that it is useful to tailor these worksheets into a format that fits your environment. To that end, we have included a set of tailored worksheet examples to help with the design of a simple cluster. These can be found in the Appendix A, “Paper planning worksheets” on page 695.

4

Cluster installation scenarios

In this chapter, the following topics are discussed:

- Preparing the cluster hardware and software
- Configuring WebSMIT
- General considerations about how to configure the cluster
- Standard configuration path - Two-Node Configuration Assistant
- Using Extended Configuration Path and C-SPOC

4.1 Basic steps to implement an HACMP cluster

In this section we present the general steps to follow while implementing an HACMP cluster. While the target configuration may differ slightly from implementation to implementation, the basic steps are the same, with certain sequence changes.

The basic steps for implementing a high availability cluster are:

1. Planning

   This step is perhaps the most important due to the fact that requires profound knowledge and understanding of your environment. A thorough planning is the key for a successful cluster implementation. For details and a planning methodology, see Chapter 3, “Planning” on page 135.

   Note: Beside the cluster configuration, the planning phase should also provide a cluster testing plan. This testing plan should be used in the final implementation phase, and also during periodic cluster validations.

2. Install and connect the hardware.

   In this step you should have your hardware environment prepared according to the configuration identified during the planning phase. The following should be performed:

   – Installing pSeries hardware (racks, power, Hardware Management console etc.)
   – Configuring the logical partitioning (where applicable)
   – Connecting machines to local networking environment
   – Connecting machines to storage (SAN)

3. Installing and configuring base operating system (AIX) and HACMP prerequisites

   In this step, the following tasks should be performed:

   – Installing base operating system, application and HACMP prerequisites (CDOM, Network Install Manager) according to local rules.
   – Configure local networking environment (TCP/IP configuration - interfaces, name resolution etc.)
   – Configure users, groups, authentication etc.

4. Configure shared storage

   Storage configuration may consist of (depending on the storage subsystem used):

   – Configure storage device drivers and multi-path extensions (if applicable)

- Configure physical-to-logical storage (RAID arrays, LUNs etc.) and storage protection
- Configure storage security (LUN masking, SAN zoning) - where applicable
- Configure the storage method for the application (file systems, raw logical volumes, or raw disks)

5. Installing and configuring application software

   In this step, the application software must be configured and tested to run as a standalone node. Perform also a manual movement and testing of the application on all nodes designated for application in the HA cluster.

   - Create and test the application start and stop scripts; make sure the application is able to recover from unexpected failures, and that the application start/stop scripts work as expected/desired on all nodes designated for running this application.
   - Create and test the application monitoring scripts (if desired) on all nodes designated to run the application

6. Install HACMP software and reboot each node
   - mandatory after applying HACMP fixes as well

7. Define the cluster and discover or manually define the cluster topology

   As HACMP provides various configuration tools, you can choose between an “standard” (easy) configuration, or the “extended” path (for more complicated configurations). Also you can choose between manually introducing all topology data, and using HACMP discovery which eases cluster configuration.

8. Synchronize the cluster topology and start the HACMP services (on all nodes).

   In this step we recommend that you verify and synchronize the cluster topology and start cluster services. Verifying and synchronizing at this step eases the subsequent implementation steps, as it is much easier to detect configuration errors and correct them in this phase, providing a sound cluster topology for further resource configuration

9. Configure cluster resources

   The following resources should be configured in this step:

   - Service IP addresses (labels)
   - Application servers (appl. start/stop scripts)
   - Application monitors (appl. monitoring scripts and actions)

10. Configure cluster resource groups and shared storage

Cluster resource groups are “containers” used for grouping resources that will be managed together by HACMP. Initially, the RGs are defined as empty containers.

- Define the RGs; synchronize the cluster
- Define the shared storage (VGs, file systems, OEM disk methods etc.)
- Populate RGs with Service IP labels(s), application server(s), VGs, appl. monitors

11. Synchronize the cluster

    As the HACMP topology ia already configured and HACMP services started, once you synchronize the cluster, the RGs will be brought online.

    Asses the cluster by checking the messages (console, /tmp/hacmp.out etc.)

12. Test the cluster

    Once the cluster is in “stable” state, you should test the cluster.

    Note: Although you can use the cluster automated test tool, we strongly recommend that you also perform a thorough manual testing of the cluster. Cluster automated test tool is specially useful

    - Document test and results
    - Update cluster documentation

4.2 Installing and configuring WebSMIT

Beside the “classic” configuration using System Management Interface Tool (SMIT), HACMP V5.2 and later also provide a Web interface for configuring your cluster. Although some preparation work is needed, using a Web interface to access the SMIT panels for HACMP is a keen method for configuring and maintaining your cluster.

Websmit also provides a GUI for monitoring a running HACMP cluster. You should keep in mind that WebSMIT is basically an interface to SMIT menus and cluster status, with some useful additions (like displaying the SMIT tree menu), and also an easy to understand graphics interface.

The basic steps to install and configure WebSMIT are:

- Installing the HACMP WebSMIT package
- Preparing the platform - installing Apache and prerequisites
- Configuring the Apache Web server for secure access
- Configuring WebSMIT and documentation

- Verifying and starting the WebSMIT pages
- Configuring and maintaining your HACMP cluster

4.2.1 Install the Apache Web server and prerequisites

This section describes how to install and configure a secure HTTP (Web) server (Apache using SSL) on your cluster nodes. You can chose to install Apache on all designated cluster nodes, or only on some of them. We recommend to perform this on all nodes in the cluster, thus you can perform cluster administration using WebSMIT from any node in the cluster.

Important: Despite the general perception that installing a Web server on a production server may pose security issues, WebSMIT configuration provides a SECURE way to administer the cluster. WebSMIT only provides access to HACMP SMIT menus (not to entire SMIT), and also provides authentication and encrypted traffic.

Before you start, you should check the latest README file in /usr/es/sbin/cluster/wsm directory on your cluster nodes.

Note: As the Apache and SSL are not IBM products, and they contain encryption software which is under US and other countries export regulation, and it is necessary to obtain the packages according to your country rules. IBM provides for download the cryptographic packages, but you MUST register on IBM Web site before access to download is granted.

Registration process may take up to 24 hours, depending on your geographic zone, thus you should prepare in advance.

The following filesets will be needed:

- rpm.rte
- expat-XXXX.ppc.rpm
- apache-XXXX.ppc.rpm
- mod_ssl-XXXX.ppc.rpm
- openssl-XXXX.ppc.rpm

Where “XXXX” should reflect the current version of the fileset, as pointed by the /usr/es/sbin/cluster/wsm/README file on your cluster nodes. At the time of this writing this redbook, we used the following versions:

```
expat-1.95.7-1.aix5.1.ppc.rpm
apache-1.3.31-1ssl.aix5.1.ppc.rpm
mod_ssl-2.8.19-1ssl.aix5.1.ppc.rpm
openssl-0.9.7d-1.aix5.1.ppc.rpm
```

Check if they are already installed on you system by the following command:

```
# rpm -qa
```

And check for the previous RPM package list.

For current AIX 5L installations, the RPM (Red Hat Package Manager) is installed by default. If RPM is not installed on your system, download it from:

```
ftp://ftp.software.ibm.com/aix/freeSoftware/aixtoolbox/INSTALLP/ppc/rpm.rte
```

And install it:

```
# installp -qacXgd rpm.rte
```

Download the Apache code and the prerequisites

With your browser, go to the following URL:

```
http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html
```

Use /tmp directory for saving the downloaded files. Download the RPM for the “expat” package, then click the link for “AIX Toolbox Cryptographic Content”, sign in, accept the license, and download the RPMS for openssl, apache, and mod_ssl.

Install the Apache and prerequisites

Assuming you have downloaded the packages in the /tmp directory, use rpm to install the four rpm files (the order of installing these packages is important), as shown in Example 4-1:

Example 4-1 Installing Apache and prerequisites using rpm

```
# cd /tmp
# rpm -ivh openssl-*.rpm
# rpm -ivh expat-*.rpm
# rpm -ivh apache-*.rpm
# rpm -ivh mod_ssl-*.rpm
```

An alternate method for installing the rpm packages is via SMIT (or **installp** command, as **installp** is able to handle rpm packages): using **smitty install_latest** fastpath.

4.2.2 Configuring WebSMIT

The HACMP software including the WebSMIT fileset have to be installed prior to this step. See “Install the Apache Web server and prerequisites” on page 235.

For configuring WebSMIT, we are going to use a configure the Apache Web server to serve the HACMP SMIT menus from a “Virtual Host”.

Also, for security reasons we use:

- user authentication (operating system user)
- Secure HTTP (HTTPS) protocol (to avoid clear text transmission — e.g., for the user password)
- a different port for the WebSMIT pages - 42267/TCP (to avoid any conflicts with other Web pages which may be accessible through the default ports 80, 443).

Note: Always check the README file in the directory /usr/es/sbin/cluster/wsm carefully. Also check for updated to his file on the IBM Web site.

Follow the README file and copy the Apache configuration file. This configuration assumes that you do not have any other Web server installation on your machine. If this is not true, you must follow local Web server configuration rules, and check with your local Web administrator before you actually configure WebSMIT and Apache.

Start by saving the original Apache configuration file (httpd.conf). Then copy the file /usr/es/sbin/cluster/wsm/httpd.conf.sample supplied with WebSMIT package to /etc/opt/freeware/apache/httpd.conf.

Modify the configuration file according to the README file and your environment (name resolution, path, IP configuration).

In Section 2 of, modify the ServerName directive according to the following Example 4-2:

Example 4-2 Changing the ServerName directive in httpd.conf file

```
.................. Omitted lines ..............
#
#ServerName localhost.your.domain

>>>>>>>>>>>>>>>> Change to : <<<<<<<<<<<<<<<<<<
#
#ServerName localhost.your.domain
```

```
ServerName ha53node1                          <------- Added this line
```

Next, in Section 3 of the httpd.conf file, we modify the VirtualHost stanza to serve the WebSMIT pages according to our local configuration. Add the entire section, as shown in Example 4-3:

Example 4-3 WebSMIT virtual host stanza

```
.................. Omitted lines ..............
### Section 3: Virtual Hosts

.................. Omitted lines ..............
##
## SSL Virtual Host Context
##

<VirtualHost _default_:443>
.................. Omitted lines ..............
<VirtualHost>

##### ------------------ Add from HERE ----------------------

</VirtualHost>
#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
#!!!!!  The following values have to be changed in order to reflect !!!!!
#!!!!!          the actual WebSMIT and HACMP configuration.         !!!!!
#!!!!!                                                              !!!!!
#!!!!!The lines start with the following entries:                   !!!!!
#!!!!! NameVirtualHost                                             !!!!!
#!!!!! <VirtualHost              >                                 !!!!!
#!!!!!                                                             !!!!!
#!!!!! ServerName                                                  !!!!!
#!!!!! ServerAdmin                                                 !!!!!
#!!!!!                                                             !!!!!
#!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!


NameVirtualHost 192.168.100.31:42267
<VirtualHost 192.168.100.31:42267>

#  General setup for the virtual host
DocumentRoot "/usr/es/sbin/cluster/wsm/htdocs/en_US"
ServerName ha53node1
ServerAdmin root@localhost
ErrorLog /usr/es/sbin/cluster/wsm/logs/error_log
TransferLog /usr/es/sbin/cluster/wsm/logs/access_log
```

```
#   SSL Engine Switch:
#   Enable/Disable SSL for this virtual host.
SSLEngine on

#   SSL Cipher Suite:
#   List the ciphers that the client is permitted to negotiate.
#   See the mod_ssl documentation for a complete list.
SSLCipherSuite ALL:!ADH:!EXPORT56:RC4+RSA:+HIGH:+MEDIUM:+LOW:+SSLv2:+EXP:+eNULL

#   Server Certificate:
#   Point SSLCertificateFile at a PEM encoded certificate.  If
#   the certificate is encrypted, then you will be prompted for a
#   pass phrase.  Note that a kill -HUP will prompt again. A test
#   certificate can be generated with `make certificate' under
#   built time. Keep in mind that if you've both a RSA and a DSA
#   certificate you can configure both in parallel (to also allow
#   the use of DSA ciphers, etc.)
SSLCertificateFile /etc/opt/freeware/apache/ssl.crt/server.crt
#SSLCertificateFile /etc/opt/freeware/apache/ssl.crt/server-dsa.crt

#   Server Private Key:
#   If the key is not combined with the certificate, use this
#   directive to point at the key file.  Keep in mind that if
#   you've both a RSA and a DSA private key you can configure
#   both in parallel (to also allow the use of DSA ciphers, etc.)
SSLCertificateKeyFile /etc/opt/freeware/apache/ssl.key/server.key
#SSLCertificateKeyFile /etc/opt/freeware/apache/ssl.key/server-dsa.key

#   Server Certificate Chain:
#   Point SSLCertificateChainFile at a file containing the
#   concatenation of PEM encoded CA certificates which form the
#   certificate chain for the server certificate. Alternatively
#   the referenced file can be the same as SSLCertificateFile
#   when the CA certificates are directly appended to the server
#   certificate for convinience.
#SSLCertificateChainFile /etc/opt/freeware/apache/ssl.crt/ca.crt

#   Certificate Authority (CA):
#   Set the CA certificate verification path where to find CA
#   certificates for client authentication or alternatively one
#   huge file containing all of them (file must be PEM encoded)
#   Note: Inside SSLCACertificatePath you need hash symlinks
#         to point to the certificate files. Use the provided
#         Makefile to update the hash symlinks after changes.
#SSLCACertificatePath /etc/opt/freeware/apache/ssl.crt
#SSLCACertificateFile /etc/opt/freeware/apache/ssl.crt/ca-bundle.crt

#   Certificate Revocation Lists (CRL):
#   Set the CA revocation path where to find CA CRLs for client
```

```
#   authentication or alternatively one huge file containing all
#   of them (file must be PEM encoded)
#   Note: Inside SSLCARevocationPath you need hash symlinks
#         to point to the certificate files. Use the provided
#         Makefile to update the hash symlinks after changes.
#SSLCARevocationPath /etc/opt/freeware/apache/ssl.crl
#SSLCARevocationFile /etc/opt/freeware/apache/ssl.crl/ca-bundle.crl

#   Client Authentication (Type):
#   Client certificate verification type and depth.  Types are
#   none, optional, require and optional_no_ca.  Depth is a
#   number which specifies how deeply to verify the certificate
#   issuer chain before deciding the certificate is not valid.
#SSLVerifyClient require
#SSLVerifyDepth  10

#   Access Control:
#   With SSLRequire you can do per-directory access control based
#   on arbitrary complex boolean expressions containing server
#   variable checks and other lookup directives.  The syntax is a
#   mixture between C and Perl.  See the mod_ssl documentation
#   for more details.
#<Location />
#SSLRequire (    %{SSL_CIPHER} !~ m/^(EXP|NULL)/ \
#            and %{SSL_CLIENT_S_DN_O} eq "Snake Oil, Ltd." \
#            and %{SSL_CLIENT_S_DN_OU} in {"Staff", "CA", "Dev"} \
#            and %{TIME_WDAY} >= 1 and %{TIME_WDAY} <= 5 \
#            and %{TIME_HOUR} >= 8 and %{TIME_HOUR} <= 20       ) \
#           or %{REMOTE_ADDR} =~ m/^192\.76\.162\.[0-9]+$/
#</Location>

#
# Aliases: Add here as many aliases as you need (with no limit). The format is
# Alias fakename realname
#
<IfModule mod_alias.c>

#
# Note that if you include a trailing / on fakename then the server will
# require it to be present in the URL.  So "/icons" isn't aliased in this
# example, only "/icons/".  If the fakename is slash-terminated, then the
# realname must also be slash terminated, and if the fakename omits the
# trailing slash, the realname must also omit it.

#
# ScriptAlias: This controls which directories contain server scripts.
# ScriptAliases are essentially the same as Aliases, except that
# documents in the realname directory are treated as applications and
# run by the server when requested rather than as documents sent to the client.
```

```
# The same rules about trailing "/" apply to ScriptAlias directives as to
# Alias.
#
    ScriptAlias /cgi-bin/ "/usr/es/sbin/cluster/wsm/cgi-bin/"

#
# "/opt/freeware/apache/cgi-bin" should be changed to whatever your ScriptAlias ed
# CGI directory exists, if you have that configured.
#
    <Directory "/usr/es/sbin/cluster/wsm/cgi-bin">
        AllowOverride AuthConfig
        Options None
        Order allow,deny
        <FilesMatch ".*\.cgi$|wsm_tree">
            Allow from all
        </FilesMatch>
    </Directory>

</IfModule>


#   SSL Engine Options:
#   Set various options for the SSL engine.
#   o FakeBasicAuth:
#     Translate the client X.509 into a Basic Authorisation.  This means that
#     the standard Auth/DBMAuth methods can be used for access control.  The
#     user name is the `one line' version of the client's X.509 certificate.
#     Note that no password is obtained from the user. Every entry in the user
#     file needs this password: `xxj31ZMTZzkVA'.
#   o ExportCertData:
#     This exports two additional environment variables: SSL_CLIENT_CERT and
#     SSL_SERVER_CERT. These contain the PEM-encoded certificates of the
#     server (always existing) and the client (only existing when client
#     authentication is used). This can be used to import the certificates
#     into CGI scripts.
#   o StdEnvVars:
#     This exports the standard SSL/TLS related `SSL_*' environment variables.
#     Per default this exportation is switched off for performance reasons,
#     because the extraction step is an expensive operation and is usually
#     useless for serving static content. So one usually enables the
#     exportation for CGI and SSI requests only.
#   o CompatEnvVars:
#     This exports obsolete environment variables for backward compatibility
#     to Apache-SSL 1.x, mod_ssl 2.0.x, Sioux 1.0 and Stronghold 2.x. Use this
#     to provide compatibility to existing CGI scripts.
#   o StrictRequire:
#     This denies access when "SSLRequireSSL" or "SSLRequire" applied even
#     under a "Satisfy any" situation, i.e. when it applies access is denied
#     and no other module can change it.
```

```
#   o OptRenegotiate:
#     This enables optimized SSL connection renegotiation handling when SSL
#     directives are used in per-directory context.
#SSLOptions +FakeBasicAuth +ExportCertData +CompatEnvVars +StrictRequire
<Files ~ "\.(cgi|shtml|phtml|php3?)$">
SSLOptions +StdEnvVars
</Files>
<Directory "/usr/es/sbin/cluster/wsm/cgi-bin">
SSLOptions +StdEnvVars
</Directory>

#   SSL Protocol Adjustments:
#   The safe and default but still SSL/TLS standard compliant shutdown
#   approach is that mod_ssl sends the close notify alert but doesn't wait for
#   the close notify alert from client. When you need a different shutdown
#   approach you can use one of the following variables:
#   o ssl-unclean-shutdown:
#     This forces an unclean shutdown when the connection is closed, i.e. no
#     SSL close notify alert is send or allowed to received.  This violates
#     the SSL/TLS standard but is needed for some brain-dead browsers. Use
#     this when you receive I/O errors because of the standard approach where
#     mod_ssl sends the close notify alert.
#   o ssl-accurate-shutdown:
#     This forces an accurate shutdown when the connection is closed, i.e. a
#     SSL close notify alert is send and mod_ssl waits for the close notify
#     alert of the client. This is 100% SSL/TLS standard compliant, but in
#     practice often causes hanging connections with brain-dead browsers. Use
#     this only for browsers where you know that their SSL implementation
#     works correctly.
#   Notice: Most problems of broken clients are also related to the HTTP
#   keep-alive facility, so you usually additionally want to disable
#   keep-alive for those clients, too. Use variable "nokeepalive" for this.
#   Similarly, one has to force some clients to use HTTP/1.0 to workaround
#   their broken HTTP/1.1 implementation. Use variables "downgrade-1.0" and
#   "force-response-1.0" for this.
SetEnvIf User-Agent ".*MSIE.*" \
 nokeepalive ssl-unclean-shutdown \
 downgrade-1.0 force-response-1.0

#   Per-Server Logging:
#   The home of a custom SSL log file. Use this when you want a
#   compact non-error SSL logfile on a virtual host basis.
CustomLog /usr/es/sbin/cluster/wsm/logs/ssl_request_log \
  "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
</VirtualHost>
```

Note: In our environment, 192.168.100.31 is the persistent IP address, and resolves to “ha53node1” in /etc/hosts.

Next, prepare the WebSMIT files and directories according to local platform, as shown in Example 4-4:

Example 4-4 Preparing WebSMIT files

```
# ha53node1_> cd /usr/es/sbin/cluster/wsm
# ha53node1_> chown nobody:sytem logs
# ha53node1_> chmod 775 logs
# ha53node1_> chmod 4511 cgi-bin/wsm_cmd_exec
# ha53node1_> ln -sf /usr/share/man/info/en_US/cluster/HAES \
> /usr/es/sbin/cluster/wsm/htdocs/en_US/HAES
```

Note: If you update the WebSMIT packages, you will have to re-run the commands shown in Example 4-4.

Check the /ust/es/sbin/cluster/wsm/wsm.conf file for the following options (shown in Example 4-5):

Example 4-5 WebSMIT configuration options

```
AUTHORIZED_PORT=42267
REDIRECT_TO_HTTPS=1
AUTHORIZED_USERS=root
REQUIRE_AUTHENTICATION=1
```

These configuration variables (default values shown here) tell us that the WebSMIT traffic is redirected to port 42267/TCP (as configured in the VirtualHost stanza for Apache), use HTTP secure communication, the allowed user to access HACMP SMIT menus via WebSMIT is “root”, and that the user is required to authenticate.

4.2.3 Starting the Apache Web server

Once configuration changes ar done, you can start the Apache Web server. We recommend to test the configuration each time before you start the server by running the following command:

```
# apachectl configtest
SYNTAX OK
```

Check for error messages. If you get a warning here the configuration should be ok, as we did not configure the main Web server (only a virtual one).

Start the Apache Web server with the SSL option using the following command:

```
#apachectl startssl
```

Note: If you fail to start the server with the SSL option (by simply running **`apachectl start`** command), you will not be able to access the WebSMIT pages!

To stop the Apache Web server do the following:

```
# apachectl stop
```

Every time any of the configuration files are updated the configuration should be tested and Apache needs to be stopped and started again for the changes to take effect.

4.2.4 Access WebSMIT pages with your browser

WebSMIT has been tested with, and fully supports Internet Explorer Version 5 or higher. Mozilla, Firefox, and other Gecko-based browsers are also supported. All Javascript- based features work, except maybe for Function Keys.
Javascript-based features will not work in Netscape Navigator V4 or lower.

There has been no testing on Opera or KHTML-based browsers.

You start WebSmit on your browser and point it to the following URL:

```
https://<yourIPaddress>:42267
```

You will see a page similar to the one shown in Figure 4-1 on page 245:

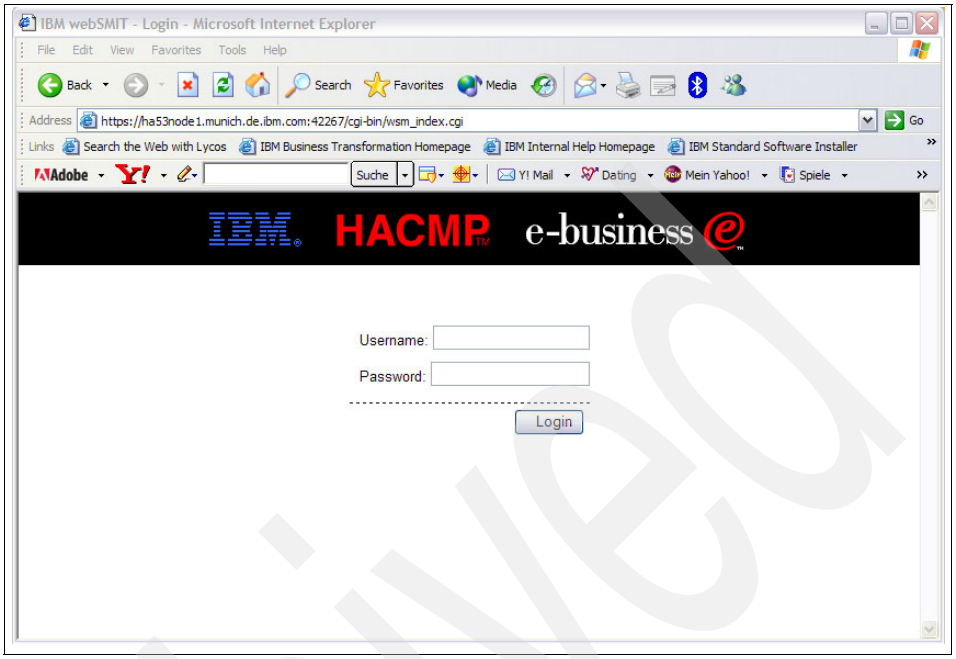

Figure 4-1 WebSMIT login page

By default you use the “root” user to access WebSMIT (as listed in /usr/es/sbin/cluster/wsm/wsm_smit.conf file). The user can be changed to any AIX user (user must exist and have a valid password).

Attention: This user will run WebSMIT with “root” authority even though the AIX user does not have the root authority.

Once users defined to AIX operating systems, you may add or change WebSMIT users in /usr/es/sbin/cluster/wsm/wsm_smit.conf file.

4.2.5 Introduction into WebSMIT

After you successfully login to WebSMIT, you will see a screen similar to the one displayed Figure 4-2 on page 246:

Figure 4-2 WebSMIT main menu

In the main part of the window you have three selectable items:

- **Cluster Status** will give you an overview of the cluster and its status. An example of a configured and running cluster is shown in Figure 4-3 on page 247.
- **Cluster Configuration and Management** gives you access to the cluster configuration tools. This is described in “WebSMIT menu: Cluster Configuration and Management” on page 247.
- **Online Documentation** gives you access to the HACMP Documentation Bookshelf. It contains all Manuals for HACMP 5.3.

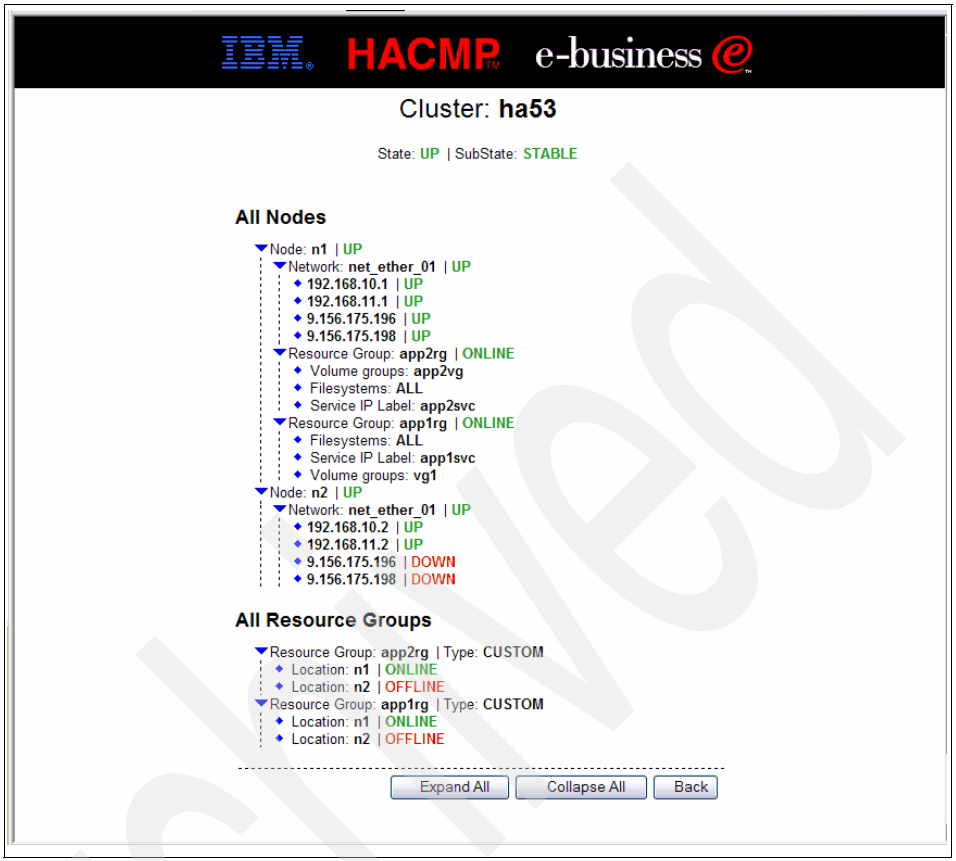

Figure 4-3 WebSMIT cluster status

Important: In the cluster status page you can also perform actions on cluster nodes, resources, and resource groups. Make sure you read the informational and help messages before you take any action.

4.2.6 WebSMIT menu: Cluster Configuration and Management

The Cluster Configuration and Management menu gives you access to the same menu structure and behavior as the SMIT with its HACMP menus. The WebSMIT menus have the following advantages compared to the SMIT menus:

- The *Treeview* gives you a complete structured overview over all menus and submenus. In addition, this is also usable as a navigation bar. You can click each item in the *Treeview*, and your main WebSMIT window will jump to this menu.

- While moving the mouse cursor over the menus in the main window pop-up windows show up. They contain context sensitive help text.
- At the bottom of the window you will always see the current fastpath. This is the same fastpath used in the SMIT menus.

WebSMIT menus are very easy to use. In addition to the normal SMIT functionality you can also go backward in the pages by:

  - The back button of the browser window
  - The backspace key
  - The F3 key
  - The F3 button on the bottom part of the page

The F1 key displayed in the bottom part of the page gives you the context sensitive help for the current page. Also, moving the mouse cursor around you will get context sensitive help for each action.

4.3 Configuring HACMP

This section presents a basic cluster configuration using various tools and menus provided. You have two ways to approach a Basic installation of HACMP:

- You can start with the two node configuration assistant as a basic configuration, and from this basic configuration, you can further adjust your configuration. This Scenario is discussed in 4.3.1, “Standard configuration path - Two-Node Configuration Assistant” on page 253.
- Or you can configure only the topology in first step, and configure all resources, resource groups etc. using the extended menus. This scenario is presented in 4.3.2, “Using Extended Configuration Path and C-SPOC” on page 260

Before you decide which way to go, make sure you have the performed the planning and the documentation for your cluster is ready to use. Refer to Chapter 3, “Planning” on page 135.

In this chapter we will configure the two scenarios according to the planning drawing shown in Figure 3-3 on page 144.

General considerations on the configuration method

When to use the Standard Configuration Path

Using the standard Configuration Path will give the opportunity to add the basic components to the HACMP Configuration Database (ODM) in a few simple steps. This configuration path significantly automates the discovery and

configuration information selection, and chooses default behaviors for networks and resource groups.

Prerequisites, Assumptions and defaults for the Standard Path

- HACMP software must be installed on all nodes of the cluster
- All network interfaces must be both physical and logical configured to AIX. You must be able to communicate from one node to each of the other nodes and vice versa.
- The HACMP discovery process runs on all server nodes, not just the local node.
- while you are using the Standard Configuration path and information that is required for configuration resides on remote nodes, HACMP automatically discovers the necessary cluster information for you.
  - Cluster discovery is run automatically while using the standard configuration path.

Restriction: You cannot select IP address takeover via replacement when using Standard Configuration. You have to modify this behavior in Extended Configuration.

- HACMP assumes all network interfaces on a physical network belong to the same HACMP network.
- Hostnames are used as node names
- HACMP uses IP aliasing as the default
- Resource group configuration with any of the policies for startup, fallover, and fallback, (without specifying fallback timer policies).
- The application start and stop scripts can be configured in the standard configuration path but the application monitoring scripts have to be implemented by using the extended configuration path.

When to use the Extended Configuration Path

In order to configure the less common cluster elements, or if connectivity to each of the cluster nodes is not available at configuration time, you can manually enter the information by using the Extended Configuration path.

Using the options under the Extended Configuration menu you can add the basic components of a cluster to the HACMP configuration Database, as well as additional types of behaviors and resources. Use the Extended Configuration path to customize the cluster for all the components, policies, and options that are not included in the Initialization and Standard Configuration menus.

Make sure you use the Extended Configuration path if you plan to:

- You do not want to be NodeName and hostname the same
- Use IPAT via IP Replacement.
- Add or change an IP-based network.
- Add or change a non-IP-based networks and devices.
- Configure a distribution preference for service IP addresses
- specifying fallback timer policies for resource groups
- Add or change any cluster configuration while one node is unavailable etc.

Persistent IP addresses

A good practice when configuring your cluster is to assign persistent IP addresses to all nodes. The persistent IP addresses are assigned as aliases on top of existing IP address to one interface on each node, and will uniquely identify a particular node.

The role of the persistent IP addresses is to be able to access the node regardless the status of the HACMP cluster. Once the cluster topology has been synchronized, as long as the node is up (even if HACMP is not running), the persistent IP is present on the node, allowing administrative access.

A possible scenario with persistent IP addresses is shown in Figure 4-4 on page 251.

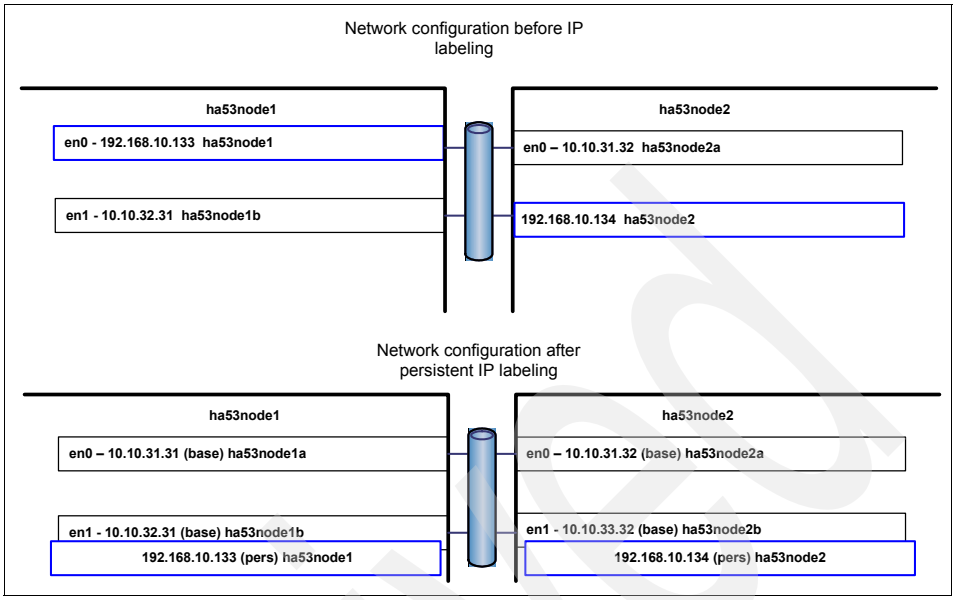

Figure 4-4 Persistent IP labeling

To change the basic adapter IP addresses you will have to be logged on to the nodes either at the system console or via telnet on an interface that is not going to be subject to this change. In this case, on node ha53node1 we have used interface en1 (IP label= ha53node1b) or on node ha53node2 use interface en0 (IP label= ha53node2a).

When you are logged onto the node check which interfaces are up.

```
# netstat -i
```

Look for the network interface which currently has the persistent IP label attached to it. Remove this persistent IP label:

```
#smitty
   ->Communications Applications and Services
      ->TCP/IP
         ->Further Configuration
            ->Network Interfaces
               ->Network Interface Selection
                  ->Change / Show Characteristics of a Network Interface
```

Select the interface you want to change. You will get a screen similar to the one shown in Figure 4-5 on page 252

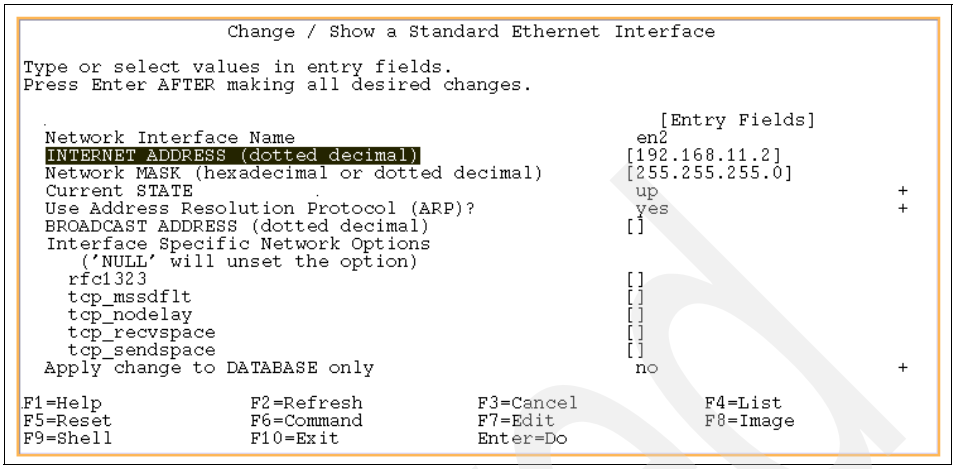

Figure 4-5 Changing the P address for an interface

In the SMIT screen shown in Figure 4-5, delete the values for INTERNET ADDRESS (dotted decimal) and NETWORK MASK (hexadecimal or dotted decimal). Set *Current STATE* to *down* or *detached*.

To be sure that the interface changed its status either to *down* or *detached* check with the following command:

```
#netstat -i
```

There is a good possibility the default route is lost after this change. We will reconfigure the routing later.

Enter the same SMIT screen as shown in Figure 4-5. You may use the SMIT fastpath **smitty chinet**. Change the interface and add its HACMP related base address with the INTERNET ADDRESS and its NETWORK MASK. Set the current State to **up**.

Add the persistent IP Label:

```
#smitty
   ->Communications Applications and Services
      ->TCP/IP
         ->Further Configuration
            ->Network Interfaces
               ->Network Interface Selection
                  ->Configure Aliases (select your IP version - we use IPV4)
                     ->Add an IPV4 Network Alias
```

select the Network Interface where this alias should be attached to. This will be most likely the interface where the second HACMP base address is. Enter the `INTERNET ADDRESS` and the `NETWORK MASK`.

Check if there is a default route:

```
#netstat -rn (or lsattr -El inet0)
```

If the default route is missing add it using:

```
# mkdev -l inet0
```

The HACMP related SMIT panels and its structure

Using WebSMIT provides an useful view of the HACMP related SMIT menu structure (tree). Every triangle marks at least one submenu. This is shown in Figure 4-6.

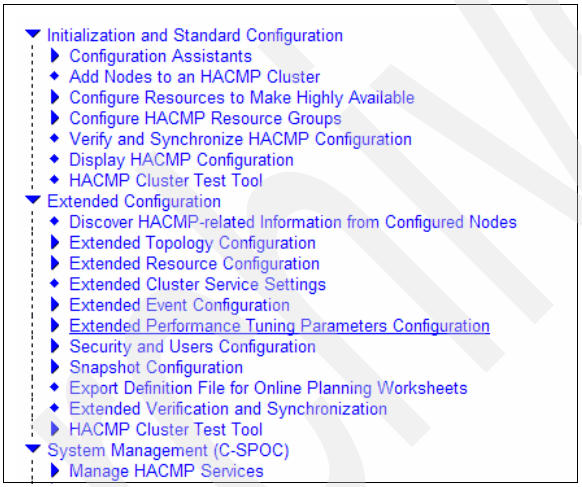

Figure 4-6 Websmit HACMP menu structure (extract)

4.3.1 Standard configuration path - Two-Node Configuration Assistant

We will setup a very simple cluster topology first by using WebSMIT. We will be using the “Two-Node Cluster Configuration Assistant” where we will have to answer five questions to get the basic cluster configured. Prior to this the Network Configuration, the LVM configuration and the Application start and stop scripts have to be available.

These five items are:

- Communication Path to Takeover Node
- Application Server Name
- Application Server Start Script
- Application Server Stop Script
- Service IP Label

Before we can do this we have to make sure that all the general assumptions of the standard Path meet our planed cluster. This is discussed in “General considerations on the configuration method” on page 248.

To ensure correct results when using the standard installation path with the Two-Node Configuration Assistant, the following preparations have to be done before you actually start:

Network configuration

While preparing the persistent IP Labels in “Persistent IP addresses” on page 250 you already configured the basic networks. For starting, the Two Node Configuration Assistant this is enough. Everything else will be added later.

Storage configuration

For the usage of the Two Node Configuration Assistant everything concerning Volume Groups, Logical Volumes and Filesystems have to be configured in advance.

Note: If you have already some volume groups (containing LVs and file systems) configured on the external disks, even though they may not be related to the application you want to integrate, the two node configuration assistant will discover and use these VGs as they are part of the cluster.

This is the procedure we recommend you to follow:

1. “Check the configuration” on page 255
2. “Create an enhanced Concurrent Volume Group” on page 255
3. “Create a log logical volume for this VG” on page 257
4. “Create as many logical volumes as you need” on page 258
5. “Create the file systems for each defined LV” on page 258
6. “Mount the Filesystems” on page 259
7. “Check there is no other Log Logical Volume” on page 259.
8. “Unmount all file systems in the HACMP related VGs” on page 259

9. “Varyoff the volume group” on page 260
10. “Import the volume group on the other node and verify” on page 260
11. “Mount all file systems” on page 260
12. Document unsupported commands (varyonvg -c -P xxxx)

Note: Steps from “Create an enhanced Concurrent Volume Group” on page 255 up to “Varyoff the volume group” on page 260 have to be done on one node only.

Steps “Import the volume group on the other node and verify” on page 260 up to step “Mount all file systems” on page 260 have to be done on all other nodes.

Check the configuration

In order to configure the shared LVM component you have to make sure that all nodes see all shared disks.

do this by running the following command on both nodes:

```
# lspv
```

compare the output on both nodes to be sure both sides see the same unassigned physical volumes (PV) with the same physical volume identifier (PVID).

If you do not see a PVID for an hdisk you have to run the following command on all the nodes prior to any further operation.

```
# chdev -l hdiskX -a pv=yes
```

Note: You have to have the same PVID attached to each shared disks. Otherwise you will not be able to create the LVM components.

Create an enhanced Concurrent Volume Group

We recommend to use only enhanced concurred Volume Groups as shared Volume Groups.

Note: Enhanced concurrent mode refers to the method of defining the VG as concurrent capable (enabling it for concurrent applications), even though this VG will be used in concurrent mode or not.

The enhanced Concurrent VG is usable for both shared (non-concurrent) and concurrent RGs. The VG can therefore be either in the Concurrent mode or in

the Shared mode. The conc. capable VG will be used in conc. RGs under the control of HACMP and RSCT. If the conc. capable VG is used in a non-conc. RG, it provides the following two features:

– You can do diskhb with any enhanced concurrent VG without having a complete disk reserved for diskhb.

– Because of being enhanced concurrent the VG is varied on all sides but active only on one node. This is used by HACMP for implementing Fast Disk Takeover. The passive varyon of the Volume Group gives the node the information about the actual status of the VG. This makes it very fast to change to an active varyon if a failure occurs. It is made sure that the VGs data are secure and integrated. Only one node can have the VG in the active state.

Note: It is not recommended to varyon an enhanced concurrent VG manually. You should always let HACMP coordinate this task.

To configure the volume group, use SMIT:

```
#smitty mkvg
```

This will give you the screen shown in Figure 4-7:

```
                        Add an Original Volume Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  VOLUME GROUP name                                  [app2vg]
  Physical partition SIZE in megabytes                                +
* PHYSICAL VOLUME names                              [hdisk3]         +
  Force the creation of a volume group?               no              +
  Activate volume group AUTOMATICALLY                 no              +
    at system restart?
  Volume Group MAJOR NUMBER                          [110]            +#
  Create VG Concurrent Capable?                       enhanced concurrent +



F1=Help          F2=Refresh       F3=Cancel        F4=List
F5=Reset         F6=Command       F7=Edit          F8=Image
F9=Shell         F10=Exit         Enter=Do
```

Figure 4-7 Defining a VG using SMIT

Make sure to set the field *Activate volume group AUTOMATICALLY at system restart* to **no.**

If you ar planning to use NFS to export directories residing in filesystems defined in this volume group, you must also make sure that the *Volume Group MAJOR NUMBER* is set to the same unique value on all nodes in the cluster.

The volume Group is not varied on right now. Vary it on now by:

```
# varyonvg app2vg
```

Create a log logical volume for this VG

We recommend to manually create a dedicated LV for JFS or JFS2 logs. This way you are able to choose the name and the placement of the logical volume.

Important: Inline logging for JFS2 shared file systems is NOT supported with HACMP.

To define the logical volume, use:

```
# smitty mklv
```

Choose the Volume Group you just created and varied on. In out example we use `app2vg`, as shown in Figure 4-8:

```
                                Add a Logical Volume

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                   [Entry Fields]
  Logical volume NAME                                  [app2loglv]
* VOLUME GROUP name                                     app2vg
* Number of LOGICAL PARTITIONS                        [1]                    #
  PHYSICAL VOLUME names                                [hdisk3]               +
  Logical volume TYPE                                  [jfs2log]              +
  POSITION on physical volume                           middle                +
  RANGE of physical volumes                             minimum               +
  MAXIMUM NUMBER of PHYSICAL VOLUMES                   []                      #
    to use for allocation
  Number of COPIES of each logical                      1                     +
    partition
  Mirror Write Consistency?                             active                +
  Allocate each logical partition copy                  yes                   +
[MORE...12]

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 4-8 Creating a log LV

For the *Number of Logical Partitions* **1** is usually enough. Make sure you set the *Logical volume TYPE* to **jfslog or jfs2log** according to your needs**.**

As *jfslog* is a special Logical Volume type this Logical Volume has to be formatted. Do this by the following command:

```
# logform /dev/app2loglv
```

Where *app2loglv* should be the name you considered for your Logical Volume Name in the previous step

You will be asked if you want to destroy the according device, answer yes.

If you do an

Create as many logical volumes as you need

Choose the same Volume Group again. In out example it is `app2vg`. Fill the following screen according to Figure 4-9

```
                                Add a Logical Volume

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                  [Entry Fields]
  Logical volume NAME                                 [app2lv]
* VOLUME GROUP name                                    app2vg
* Number of LOGICAL PARTITIONS                        [30]                    #
  PHYSICAL VOLUME names                               [hdisk3]                +
  Logical volume TYPE                                 [jfs2]                  +
  POSITION on physical volume                          middle                 +
  RANGE of physical volumes                            minimum                +
  MAXIMUM NUMBER of PHYSICAL VOLUMES                  []                       #
    to use for allocation
  Number of COPIES of each logical                     1                      +
    partition
  Mirror Write Consistency?                            active                 +
  Allocate each logical partition copy                 yes                    +
[MORE...12]

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 4-9 Creating an LV

Please make sure that the Logical Volume type is set to the value according to the one you have chosen for the Log Logical Volume. Repeat this for all file systems to be added to this shared VG.

Create the file systems for each defined LV

Use SMIT:

```
#smitty
   ->System Storage Management (Physical &Logical Storage)
     ->File Systems
        ->Add /Change / Show Delete File Systems
           ->Enhanced Journaled File Systems
              ->Add a Enhanced Journaled File Systems on a Previously Defined
                 Logical volume
```

Just create all file systems on their Logical Volumes as shown in Figure 4-10 on page 259:

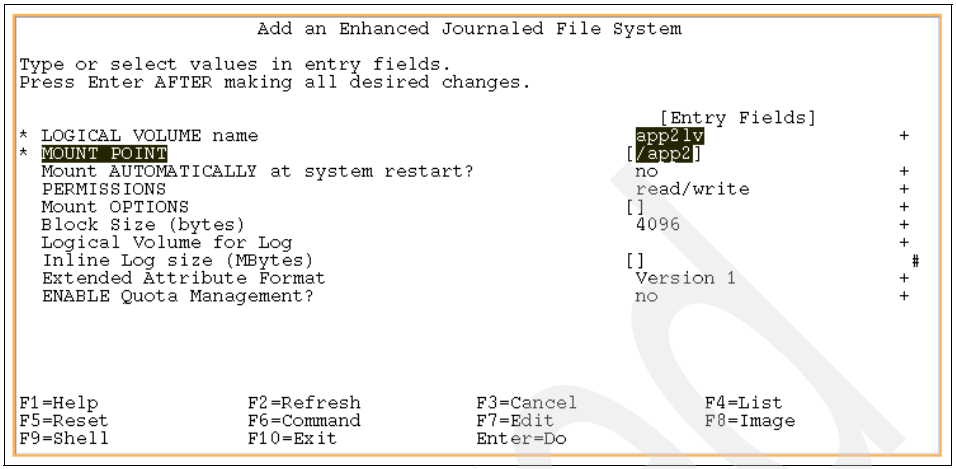

Figure 4-10 Creating file systems on previously defined LVs

Make sure that the value in field: *Mount AUTOMATICALLY at system restart* is set to **no**

Mount the Filesystems

You want to mount all the filesystems now by just doing

```
# mount /app2
```

Check there is no other Log Logical Volume

You have to make sure now that the Log Logical Volume you created is being used. An example is shown in Example 4-6:

Example 4-6 Checking the log LV

```
root@ha53node1:/>
root@ha53node1:/> mount /app2
root@ha53node1:/> lsvg -l app2vg
app2vg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
app2loglv           jfs2log    1     1     1    open/syncd    N/A
app2lv              jfs2       30    30    1    open/syncd    /app2
root@ha53node1:/>
```

Unmount all file systems in the HACMP related VGs

Unmount all you HACMP related Filesystems by:

```
# umount /app2
```

Varyoff the volume group

Varyoff the Volume group by:

```
#varryoffvg app2vg
```

Import the volume group on the other node and verify

You are going to import the Volume Group on the other node. Therefore you have to specify the VG major number you set on the first node. And you have to make sure, that you specify the correct and corresponding hdisk.

In our scenario the command in like follows:

```
# importvg -V 102 -y app2vg hdisk2
```

Next step it to vary the volume group on:

```
# varyonvg app2vg
```

Mount all file systems

To verify that everything is correct mount all filesystems and check again that the right LogLV is used.

```
# mount /app2
# lsvg -l app2vg
```

Varyoff all VGs:

```
# varyoffvg app2vg
```

Application preparation

In order to let HACMP take care about your applications a set of two to three scripts have to be prepared for each resource group.

- The start script will start the application
- The stop script will stop it

4.3.2 Using Extended Configuration Path and C-SPOC

In this scenario we will start configuring the cluster by discovering the cluster topology first and use the C-SPOC to do all the remaining configuration. This will change the list of steps which is shown in “Basic steps to implement an HACMP cluster” on page 232, like follows:

“Step 4: Configure shared storage “and “Step –: Create and test the application start and stop scripts; make sure the application is able to recover from unexpected failures, and that the application start/stop scripts work as expected/desired on all nodes designated for running this application. on page

233” will be done after “Step 7: Define the cluster and discover or manually define the cluster topology on page 233”

All further cluster configuration will be done via c-spoc while the cluster is running

Configuring the topology using Extended Configuration path

When using Extended Configuration path, you have the possibility to define granular options and certain configuration parameters not accessible via Standard Configuration path.

Define the cluster

```
#smitty
   ->Communications Applications and Services
      ->HACMP for AIX
         ->Extended Configuration
            ->Extended Topology Configuration
               ->Configure an HACMP Cluster
                  ->Add/Change/Show an HACMP Cluster
```

Choose a name for your cluster.

Add nodes to the cluster

After defining the cluster name, add the nodes and a communication path (IP address assigned to an available network interface) to each node:

```
#smitty
   ->Communications Applications and Services
      ->HACMP for AIX
         ->Extended Configuration
            ->Extended Topology Configuration
               ->Configure HACMP Nodes
```

Enter the Nodename and the communication path to this node here. In the extended configuration path you can select a different Node Name than the hostname. Repeat this for all nodes which will be in this cluster.

Add the Networks (IP, non-IP)

Using the Extended Path you have to create the Network and Network Interfaces.

```
#smitty
   ->Communications Applications and Services
      ->HACMP for AIX
         ->Extended Configuration
            ->Extended Topology Configuration
               ->Configure HACMP Networks
```

Select **Add a Network**. You will be asked to select a Network Type. Look if your type is listed under `Discovered IP-based Network Types, if not select it in the Pre-defined IP-based Network Types` Section. This will give you a screen shown in Figure 4-11

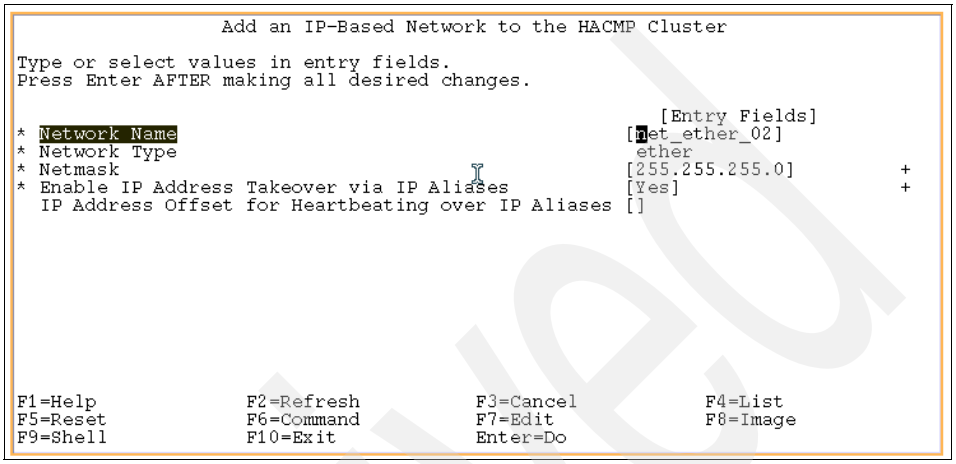

Figure 4-11 Adding an IP-based network

In this menu you can also specify not to use IP Address Takeover via IP Aliases. If you want to use IPAT via replacement, select **NO** here.

Repeat for all networks (IP, non-IP) you want to add.

Communication interfaces and devices

Next step is to Configure Communication Interfaces/Devices. Select the according menu item on the `Extended Topology Configuration` screen, select `Add Communication Interface/Devices` on the next screen, select `Add Pre-defined Communication Interfaces and Devices` and then `Communication Interfaces`. The previously created Network will show up for selection. Select it will give you the screen shown in Figure 4-12 on page 263.

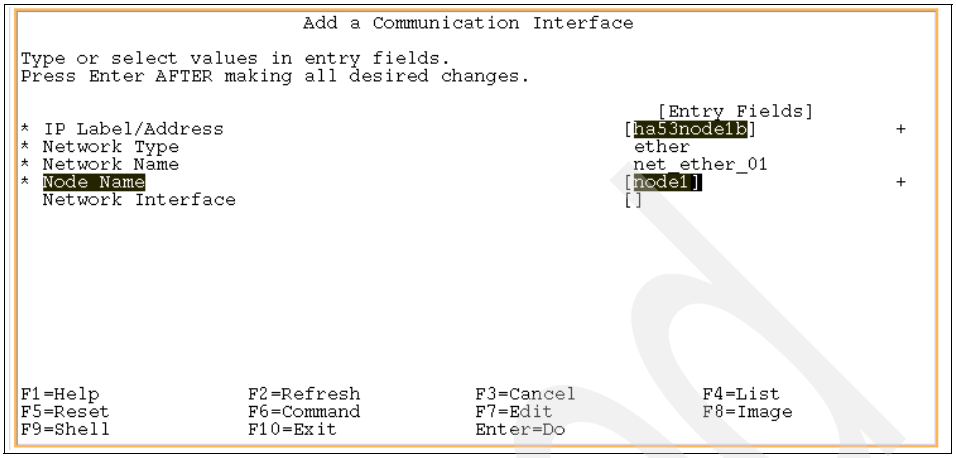

Figure 4-12 Adding communication interfaces

Repeat this step for all communication interfaces and devices you need to configure.

Discover HACMP related information from all nodes

Instead of entering all data manually, once you have defined the cluster and the nodes (providing a communication path to each node), you can run HACMP discovery:

```
#smitty
   ->Communications Applications and Services
      ->HACMP for AIX
         ->Extended Configuration
            ->Discover HACMP related Information from Configured Nodes
```

This will automatically discover the cluster for you (networks - IP and non-IP, communication interfaces and devices, shared volume groups).

The data discovered will be presented as a selection for the extended topology and resources menus.

Verify and synchronize the cluster

On the SMIT screen *"Extended Configuration"* you will find as well the next step to do.

```
->Extended Verification and Synchronization
```

This will give you the screen shown inFigure 4-13 on page 264.

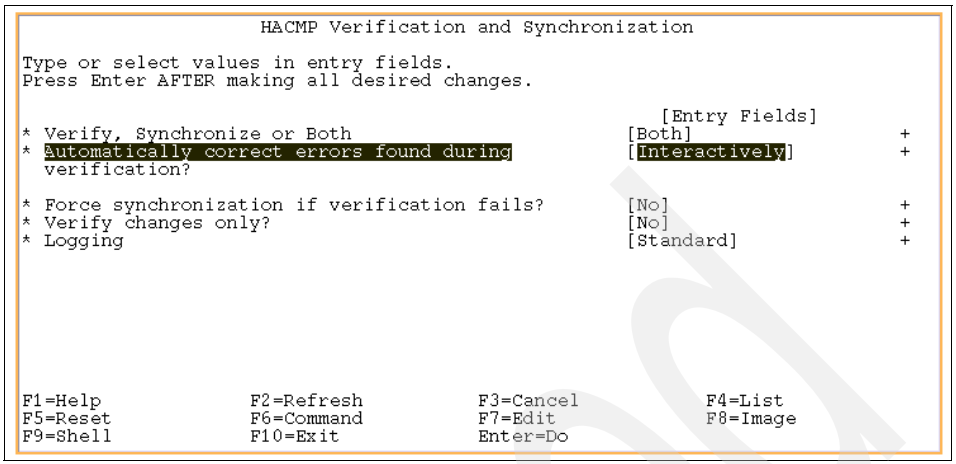

Figure 4-13 HACMP extended verification and synchronization

Select the values according to what you want to do, right here you should select the above shown values.

You car run Cluster Verification and Synchronization any time you want to. There are certain steps where it is required. There are for example:

- before you start cluster services, when you changed the configuration via Extended Path.

Note: Automatic error correction can only be performed if HACMP is stopped on all nodes in the cluster.

Configuring every thing else via C-SPOC

C-SPOC stands for Cluster Single Point Of Control, and is a powerful administration tool that allows cluster configuration changes from a single node.

Why use C-Spoc?

As a system administrator of an HACMP cluster, you may be called upon to perform any of the following tasks:

- LVM-related
  - Creating a new shared volume group
  - Extending, reducing, changing, or removing an existing volume group
  - Creating a new shared logical volume
  - Extending, reducing, changing, or removing an existing logical volume
  - Creating a new shared filesystem
  - Extending, changing, or removing an existing filesystem

– Adding, removing physical volumes

C-SPOC will avoid most of the common failures regarding the stable cluster information on all nodes.

The C-SPOC commands only operate on both shared and concurrent LVM components that are defined as part of an HACMP resource group. When you use SMIT HACMP C-SPOC, it executes the command on the node that owns the LVM component (the node that has it varied on).

The C-SPOC commands that modify LVM components require a resource group name as an argument. The LVM component that is the target of the command must be configured in the resource group specified. C-SPOC uses the resource group information to determine on which nodes it must execute the operation specified.

The only task you are allowed to do in C-SPOC without a related Resource Group is to generate an Volume Group.

Removing a Filesystem or Logical Volume

When removing a filesystem or logical volume using C-SPOC, the target filesystem or logicalvolume must not be configured as a resource in the resource group specified. You must remove the configuration for it from the resource group before removing the filesystem or logicalvolume.

Updating LVM Components in an HACMP Cluster

When you change the definition of a shared LVM component in a cluster, the operation updates the LVM data that describes the component on the local node and in the Volume Group Descriptor Area (VGDA) on the disks in the volume group. AIX 5L LVM enhancements allow all nodes in the cluster to be aware of changes to a volume group, logical volume, and filesystem, at the time the changes are made, rather than waiting for the information to be retrieved during a lazy update.

For any further information about C-SPOC refer to Chapter 9 in *High Availability Cluster Multi-Processing Administration Guide*, SC23-4862-06.

How to use C-SPOC

You will enter the Smitty C-Spoc Menu by the following SMIT Path:

```
#smitty
   ->Communications Applications and Services
      ->HACMP for AIX
         ->System Management (C-SPOC)
```

This will give you the screen shown in Figure 4-14 on page 266.

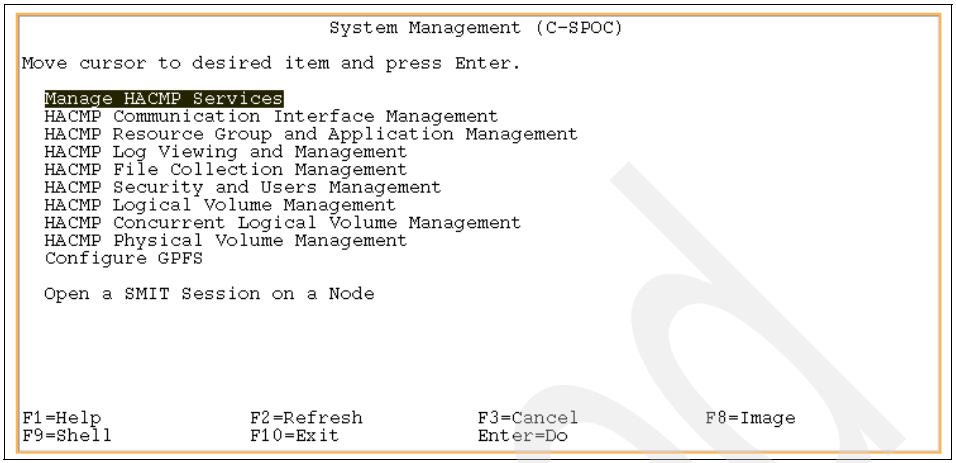

Figure 4-14 C-SPOC main menu

You can find more information about C-SPOC in 8.1, “CSPOC DP” on page 354.

5

Migrating a cluster to HACMP V5.3

This chapter describes the various ways to migrate a cluster to HACMP 5.3. There are important planning steps and considerations to acknowledge prior to undertaking a migration. Before starting, make sure that you are familiar with the use of the cluster snapshot utility. Also, be aware of the current configuration, how it is expected to behave and of the changes in each release that you upgrade to, since your end result may not work or behave the same way.

Attention: Always review the planning and installation manual for a list of documented steps if you are not familiar with the procedure.

5.1 Identifying the migration path

There are three migration paths to get your cluster to a higher release. Identifying which one to use can be easily determined by whether you can tolerate an interruption to the entire cluster or whether you need to maintain maximum availability.

- **Rolling Migration** (ES to ES): This method will maintain the resources online on at least one node while you upgrade each of cluster nodes one at a time. This will leave the cluster in a mixed state during the migration period.
- **Snapshot Method**: This method will require all cluster nodes to be inaccessible for a period of time and that all nodes have the new HACMP release installed before trying to convert your pre-migration snapshot and applying it.
- **Node by Node** (HAS to ES): This method is similar to a Rolling migration. However, as each node is upgraded, both versions of HACMP code will show as being installed at the same time, with the old daemons actually managing the cluster. Only when the last node in the cluster is upgraded and integrated into the cluster will the final conversion be done.

Tip: Note that in the past the concepts of a Rolling and Node-by-Node migration were used synonymously.

5.1.1 Supported migration paths

Table 5-1 Supported release upgrades to HACMP 5.3

Existing Version	Rolling Migration	Snapshot Conversion	Node-by-Node Migration
HACMP/ES 5.2	Yes	Yes	Not Applicable
HACMP/ES 5.1	Yes	Yes	Not Applicable
HACMP/ES 4.5	Yes	Yes	Not Applicable
HACMP 4.5	Not Applicable	Yes	Yes

5.2 Prerequisites

Before upgrading HACMP ensure that you are familiar with high-level concepts to low-level tasks, such as planning, maintenance and troubleshooting since many of the sections in this chapter will build on that knowledge.

Space requirements (specific for HAS to HAES migration):

There must be enough disk space to hold both HAS and HAES software during the migration process:

- Approximately 120 MB in the /usr directory.
- Approximately 1.2 MB in the / (root) directory.

The nodes must also have enough memory to run both HACMP sets of daemons simultaneously. There is a minimum of 64MB of RAM, however, 128MB of RAM is recommended (not including the application memory requirements).

Cluster software and RSCT prerequisites

Ensure that the same level of cluster software and RSCT filesets (including PTFs) are on all cluster nodes before starting a migration. Also, ensure that software installation is committed (not just applied).

To ensure that the software is already committed:

1. Run the **`lslpp -h cluster.*`** command
2. If the word APPLY displays under the action header, enter `smit install_commit` before installing the HACMP software

   SMIT displays the Commit Applied Software Updates (Remove Saved Files) panel.
3. Enter field values as follows:

   SOFTWARE name: From the picklist, select all cluster filesets.

   COMMIT old version if above version used it?: Set this field to yes.

   EXTEND file system if space needed?: Set this field to yes.

Understanding the configuration

Know your cluster configuration. Consider reviewing and understanding the overall configuration prior to the upgrade. If new to your configuration consider running **`cltopinfo`** or **`cldump`** to get an overview of the environment. The output of **`cllsif`** and **`clshowres`** will also give you a good sense of how the cluster is configured and its expected fallover behavior. This information is also useful in the event that you need to place a call to AIX Support.

- `/usr/es/sbin/cluster/utilities/cllsif`
- `/usr/es/sbin/cluster/utilities/clshowres`

The first output will show you the topology currently configured. It will help you easily identify the available networks and the associated labels and IPs and their current function. The second command will display each resource group and all

of the associated attributes. This output should help you determine the currently assigned resources and their designated fallover behavior.

Pre-migration recommendations and checks

The following are things that you should make a habit of doing before any migration in order to guarantee its success and to have a reliable backout plan:

- Always take a snapshot before commencing a migration. Remember to save it in multiple safe locations.
- Always have a system backup (**mksysb**) with the current configuration. Make sure that you have checked its contents and that it will restore your data.
- If the resources are available, consider creating an alternate disk installation on each cluster node prior to the migration. This is useful in the event you have to quickly revert back to the old configuration. To change back you would alter the bootlist to the alternate disk and reboot the migrated nodes.
- Ensure that the same level of cluster software and RSCT filesets (including PTFs) are on all cluster nodes before starting a migration.
- The / .rhosts file is only needed during the migration from versions prior to HACMP 5.1. Once the migration is completed, we recommend removing the .rhosts file in root (/) if no other applications need rsh for internode communication.

  Note: If you have application and/or pre- and post-event scripts that require remote command execution (based on AIX rsh), you still need to keep the ~/.rhosts file even after migration.

- Make sure that your cluster will fallover successfully before the migration, otherwise it will probably also not work after the upgrade.
- Check the state of the system and filesets installed:
  - Run **lppchk -v, -l, -c**
  - Run **instfix -i | grep ML**
  - Review **errpt -a** for any recent pertinent errors
  - Run **df -k** and check for full file systems
  - Run **lsps -s** and make sure that paging space is not full
  - Run **emgr -l** to check for any efixes loaded on the system before initiating the upgrade

5.3 Considerations

Note that various things have changed between earlier releases and HACMP 5.3. In some instances this will lead to modified cluster behavior. The following are things to keep in mind going into an upgrade:

- As of HACMP 5.2, the resource group types of `cascading`, `rotating` and `concurrent` were converted to custom resource groups that have corresponding start/fallover/fallback policies. If migrating from releases prior to HACMP 5.2 the migration will convert the resource groups to use the corresponding policies within each one. For a detailed explanation of the policies within custom resource groups refer back to Chapter 1.
- HACMP tunable values will be reset when you upgrade the cluster. These can include the following:
  - Network module tuning parameters, such as failure detection rate, grace period and heartbeat rate. These are reset to their installation-time default values.
  - Modifications to cluster events such as pre or post events will be reset. Resetting these changes does not remove any files or scripts that the customization used, but HACMP will lose any knowledge of them.
- Any changes to the default set of HACMP commands will be reset to the installation-time defaults.
- Once you begin a rolling migration you cluster will be running in a mixed mode until the last node is upgraded and reintegrated into the cluster. During this period of time do not attempt to make any changes to the cluster topology or resources.
  - Do not verify or synchronize the cluster.
  - Do not force down a node.
  - Do not attempt a DARE or C-SPOC operations, except for the Manage HACMP Services functions.
  - Do not use the `Problem Determination Tools > View Current State` function.
  - Do not use the `Extended Configuration > Snapshot Configuration > Add a Cluster Snapshot` option or run the clsnapshot command.
  - Do not use the `Problem Determination Tools > Recover from HACMP Script Failure` option or run the **`clruncmd`** command, except when running the command or SMIT option from the target node specified by the command.

Important: Do not leave the cluster in a hybrid state for an extended period of time to avoid accidentally invoking any of these operations.

- If upgrading from a release prior to HACMP 5.2 the HACMP installation creates the *hacmp* group on all nodes. The HACMP Configuration Database (ODM) classes have been updated so that they are owned by the root user and the *hacmp* group.
  - The permissions of 640 are set for most HACMP object classes. The `HACMPdisksubsystem` is an exception with 600.
  - All HACMP binaries intended for use by non-root users are installed with 2555 permissions. The **`setgid`** bit is turned on so that the program runs as *hacmp* group.
  - If you use programs that access the ODM directly they may need to be rewritten.

    Attention: Using the information retrieved directly from the ODM is for informational purposes only as the format within the stanzas may change with updates, and/or new versions.

    Thus hardcoding ODM queries within user defined applications is not supported and should be avoided.

  - If using the PSSP File Collections facility to maintain the consistency of /etc/group, the new *hacmp* group may be lost when next file synchronization occurs. To avoid this do the following:
    - Turn off the PSSP File collection synchronization of /etc/group
    - Include the *hacmp* group in the master /etc/group file and propagate the change to all cluster nodes.

    Note: For security reasons you should not expand the authority of the *hacmp* group.

- The dynamic node priority policies prior to HACMP 5.2 were based on RSCT Event Management resource variables. If your configuration includes an HACMP 5.1 resource group with dynamic node priority policy set for Fallover using Dynamic Node Priority, the migration will reset this setting. The **`cl_convert`** utility changes the fallover policy for that resource group to Fallover to Next Priority Node in the list when it completes.
  - During the migration the default node fallover policy is used.
  - Note that only 3 policies are supported with HACMP 5.3:

- cl_highest_free_mem
- cl_highest_idle_cpu
- cl_lowest_disk_busy

- In HACMP 5.3, the only resource group distribution policy is the node-based distribution policy. The network-based-distribution policy previously available was removed. When you migrate, this is converted to the node-based policy.
- User-defined cluster events may no longer work since `emsvcs` are no longer used by HACMP. To correct any problems rewrite them using **`rmcd`**.
- The Cluster Lock Manager (`cllockd`, or `cllockdES`) is no longer available in HACMP 5.3. The installation removes the Lock Manager files and definitions from a node. Consult with the application vendor about concurrent access support.
- Note that Enhanced Concurrent Mode is only supported on AIX 5L V5.1 and later. SSA concurrent mode is not supported on 64-bit kernels. If you have SSA disks in concurrent mode, you cannot run 64-bit kernels until you have converted all volume groups to enhanced concurrent mode.

5.4 General migration steps

Rolling migration

The rolling migration path is used if you need to maintain the application available during the migration process. The steps for this type of migration are:

1. Stop cluster services with takeover on first node to be migrated.
2. Upgrade the AIX and RSCT software (if necessary).
3. Upgrade the HACMP software (including latest PTFs).
4. Reboot.
5. Reintegrate the node into the cluster and repeat steps on next node.

In Figure 5-1 on page 274 you can view an example of the upgrade for an HACMP 4.5 cluster on AIX 5.1 ending up at HACMP 5.3 with AIX 5.2.

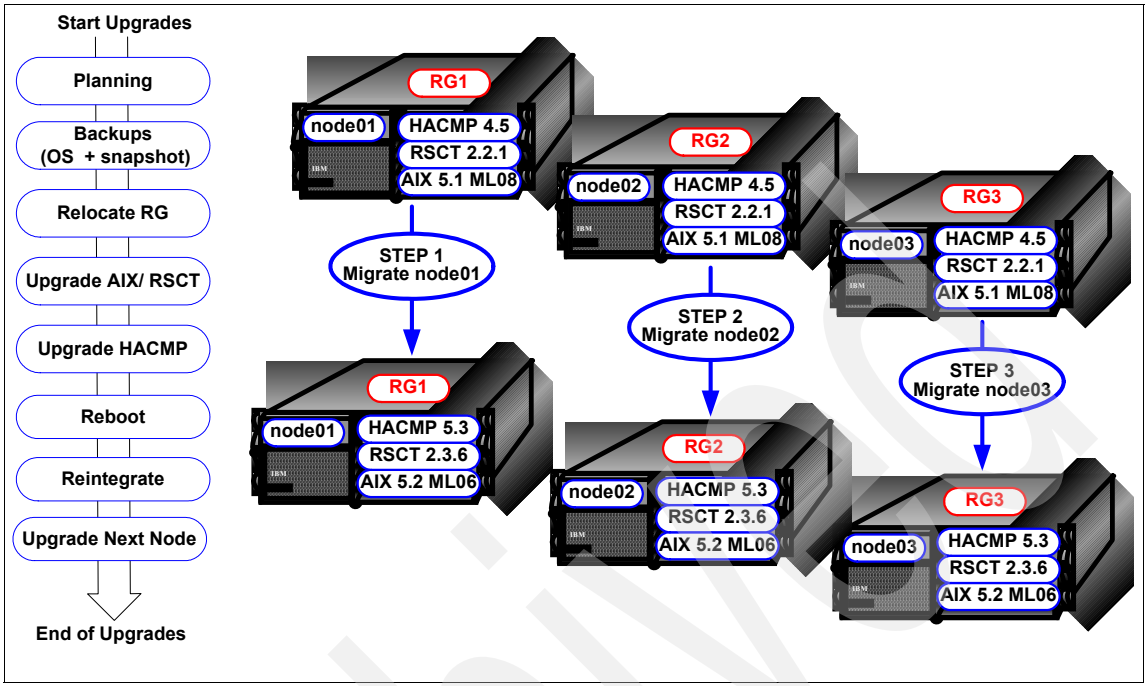

Figure 5-1 General rolling migration steps- HA 4.5 to HA 5.3

Snapshot migration

You also have the option to migrate a cluster via the snapshot method. Using this method you use a snapshot taken prior to the upgrade to restore the configuration. When the removal and reinstallation of all the nodes is finished you will convert and then apply the snapshot. However, the entire cluster will be unavailable during the entire process, and all nodes must be upgraded before the cluster is activated again.

1. Stop cluster services on all cluster nodes.
2. Upgrade the AIX and RSCT software on all nodes (if necessary).
3. Deinstall the current version of HACMP and install HACMP 5.3 on all nodes (including latest PTFs if available).
4. Reboot.
5. Convert the snapshot.
6. Apply the snapshot (this will push the configuration to all nodes).
7. Start cluster services on one node at a time.
8. Verify and synchronize the cluster.

Tip: If planning to use the snapshot method be aware that when you try to reapply it to the nodes the first communication path that HACMP will try to use on each node is the one specified in the HACMPnode object class.

```
#odmget HACMPnode
HACMPnode:
        name = "cobra"
        object = "COMMUNICATION_PATH"
        value = "10.10.32.33"
        node_id = 2
        node_handle = 2
        version = 8
```

This value is originally set whenever the nodes were first defined to the cluster and a `COMMUNICATION PATH` was specified. A best practice recommendation is to always set this path as the persistent IP address. If for any reason that IP is not available when you try to apply the snapshot manually set that alias on one of the interfaces.

The last file checked for communication is the `/usr/es/sbin/cluster/etc/rhosts`, which can be manually updated with all of the cluster IP addresses if necessary. The IPs will include: base, service and persistent IPs.

5.5 Scenarios tested

During the writing of this redbook we tested migrations to HACMP V5.3 from three different releases:

- HAES 4.5
- HAES 5.1
- HAES 5.2

For each of these we performed a rolling migration and a snapshot migration and documented the steps and results. Note that due to the move from HAS to HAES we selected to not cover the node-by-node migration path. If preparing for an upgrade from HAS 4.5 we would recommend the snapshot conversion method if possible.

We performed all installations and upgrades using a NIM server with the latest PTFs to date for HACMP, AIX, RSCT, and SDD.

5.5.1 Scenario 1 - AIX 5.1 and HAES 4.5

For our first migration test we selected to upgrade a three-node AIX 5.1 / HAES 4.5 cluster to AIX 5.2 and HA 5.3. We used p630 servers (7028-6C4) for our cluster configuration. Figure 5-2 presents the diagram of the configuration that we used in our test environment.

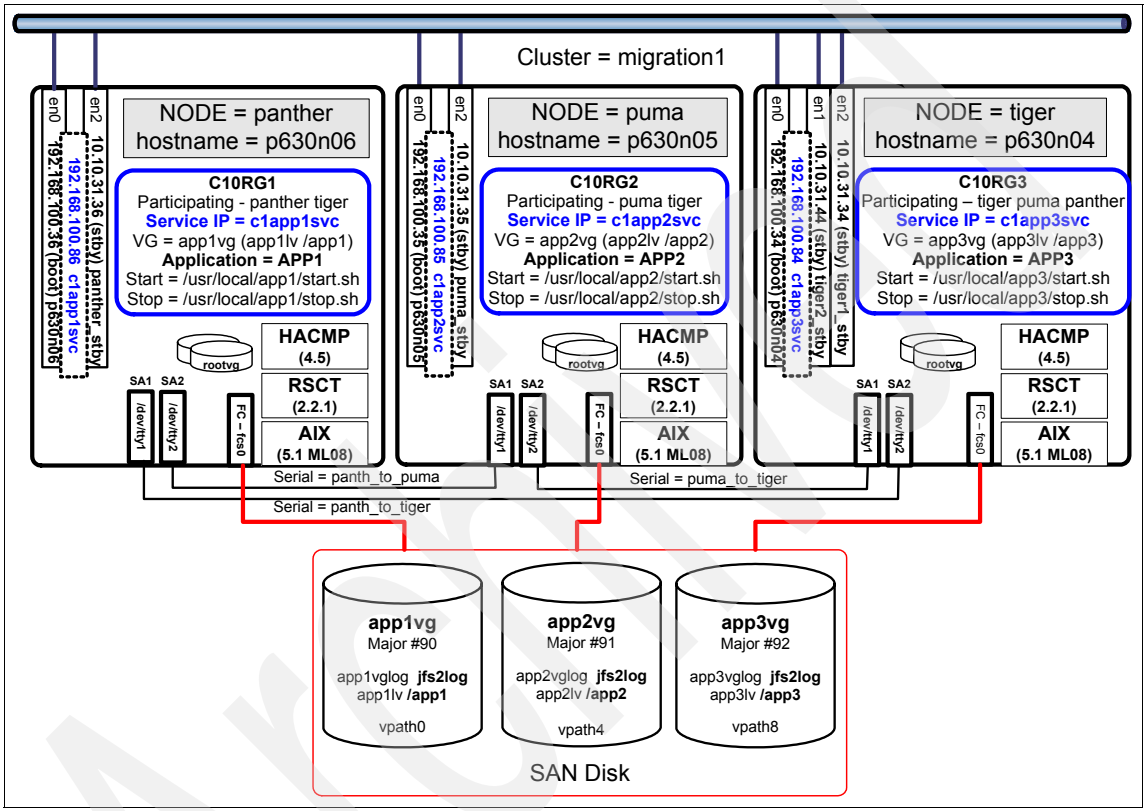

Figure 5-2 Scenario 1 - HA 4.5 to HA 5.3 migration environment

In our cluster each node managed its own resource group. Each resource group comprised of an application volume group, a service IP, and an application server. The topology was configured to use IPAT via Replacement in an effort to test the conversion to HACMP 5.3. We configured the rs232 networks utilizing the integrated ports on our nodes for our non-IP heartbeat traffic. Our storage consisted of switch attached ESS LUNs (2105-800) along with the host attachment script (ibm2105.rte) and the subsystem device driver.

Rolling migration steps - scenario 1

We began our scenario after testing that our 3-node cluster was stable and could fallover successfully. We selected to upgrade node **panther** first.

We took the following preliminary steps:

1. From the working cluster (all nodes active) we saved a snapshot.

> **Attention:** Do not save the snapshot to /tmp as the contents of /tmp are removed during the AIX migration. Consider saving it into another directory or another server.
>
> Remember to also save any application start/stop scripts to a safe directory.

2. Took a mksysb.
3. Created an alt_disk_install as a means to revert back to the previous configuration. This was done in an effort to speed up the retesting.

Migration scenario

The following are the steps we followed for our migration scenario:

1. Stopped HACMP on node **panther** graceful with takeover.
   - C1ORG1 moved to node **tiger.**

     Confirmed with /usr/es/sbin/cluster/utilities/clfindres
2. Initiated the AIX5.2/RSCT migration.
   - Applied latest AIX 5.2 fixes (ML06)
   - Updated and verified RSCT levels (rsct.basic.rte 2.3.6.2)
   - Removed and replaced SDD with the current AIX 5.2 specific driver:
     - **stopsrc -s sddsrv**
     - **rmdev -dl dpo -R**
     - Uninstalled 5.1 SDD with smitty remove (devices.sdd.51.rte 1.6.0.2)
     - Installed 5.2 SDD (devices.sdd.52.rte 1.6.0.0 & 1.6.0.2 PTF)
3. Ran smit update_all to update the HACMP filesets to 5.3.
4. Rebooted node **panther.**
   - Verified AIX install (lppchk -l / -c / -v, instfix, oslevel, errpt)
   - lslpp -l | grep cluster => All 4.5 gone (check for any missed filesets)
   - odmget HACMPcluster => version still showed 4.5 version [5]

5. Started hacmp on **panther**.
   - Our configuration had C1ORG1 set to fallback to the higher priority node upon reintegration, as a result the resource group returned to node **panther**.
   - **cldump** showed the proper location and status of all resources.

   Attention: Synchronizing the cluster while in this hybrid state would break the migration and leave you in an inconsistent state.

6. Repeated the steps for node **puma**.
7. Repeated the steps for node **tiger**.

When the last node entered the cluster the conversion of the ODMs was made and nodes showed to be at `cluster_version = 8` when looking at the `HACMPcluster` and `HACMPnode` object classes.

Rolling migration results - scenario 1

The AIX migration was the slowest portion of this upgrade. The reintegration of the last cluster node was successful and we were able to confirm that the different HACMP ODM stanzas were converted properly and that all nodes now showed to be at `cluster_version = 8`. Refer to Table 5-2 on page 292 for how to check HACMP release version numbers.

Upon completion we confirmed the status of the cluster by using the clstat utility and confirming the status of the nodes in the output of **lssrc -ls clstrmgrES**.

```
#lssrc -ls clstrmgrES
Current state: ST_STABLE
```

We utilized the cluster test tool upon completion to run a variety of tests in one operation and identified no problems. We simulated the migration test again and experienced no problems with the cluster functionality.

Snapshot migration - scenario 1

For the same 3-node cluster we also tested the snapshot conversion method. We used our alternate disk installation images to revert back to our previous environment, and restarted the cluster services on all nodes running HA 4.5. We proceeded our testing with the following steps:

1. Stopped HACMP on all nodes: panther, puma, tiger
2. Ran `smit remove` and deinstalled all `cluster.*` filesets from all of the nodes.
3. Migrated the AIX/RSCT code via NIM

4. Installed the HACMP packages with current PTF levels onto all nodes.
5. Rebooted all cluster nodes.
6. Copied the snapshot.odm file that we previously saved in the rolling migration scenario into `/usr/es/sbin/cluster/utilities/snapshot.odm`
7. Ran the following to convert the snapshot:

   ```
   #/usr/es/sbin/cluster/conversion/clconvert_snapshot -v 4.5 -s snapshot.odm
   ```

8. Applied the snapshot by following running `smit hacmp > Extended Configuration > Snapshot Configuration > Apply a Cluster Snapshot >` selected snapshot and pressing Enter.
9. Started cluster services one node at a time.

Snapshot migration results - scenario 1

The overall snapshot migration scenario was very quick. Although it did require a cluster wide outage, the conversion of the snapshot file took no more than a minute and applied very quickly across all 3 nodes.

When it completed we confirmed that the HACMP ODM stanzas were converted successfully by checking for the cluster version in the `HACMPcluster` and `HACMPnode` object classes. We also utilized cluster test tool upon completion and found no problems.

Tip: If planning to use the snapshot method be aware that when you try to reapply it to the nodes the first communication path that HACMP will try to use on each node is the one specified in the HACMPnode object class.

```
#odmget HACMPnode
HACMPnode:
        name = "cobra"
        object = "COMMUNICATION_PATH"
        value = "10.10.32.33"
        node_id = 2
        node_handle = 2
        version = 8
```

This value is originally set whenever the nodes were first defined to the cluster and a `COMMUNICATION PATH` was specified. A best practice recommendation is to always set this path as the persistent IP address. If for any reason that IP is not available when you try to apply the snapshot manually set that alias on one of the interfaces.

The last file checked for communication is the `/usr/es/sbin/cluster/etc/rhosts`, which can be manually updated with all of the cluster IP addresses if necessary. The IPs will include: base, service and persistent IPs.

5.5.2 Scenario 2 - AIX 5.2 and HA 5.1

Our second test scenario consisted of the migration of a 2-node AIX 5.2 / HA 5.1 cluster configured for mutual takeover, to AIX 5.3 / HA 5.3. We used p630 servers (7028-6C4) for our cluster configuration. Figure 5-3 is a diagram of the configuration that we used:

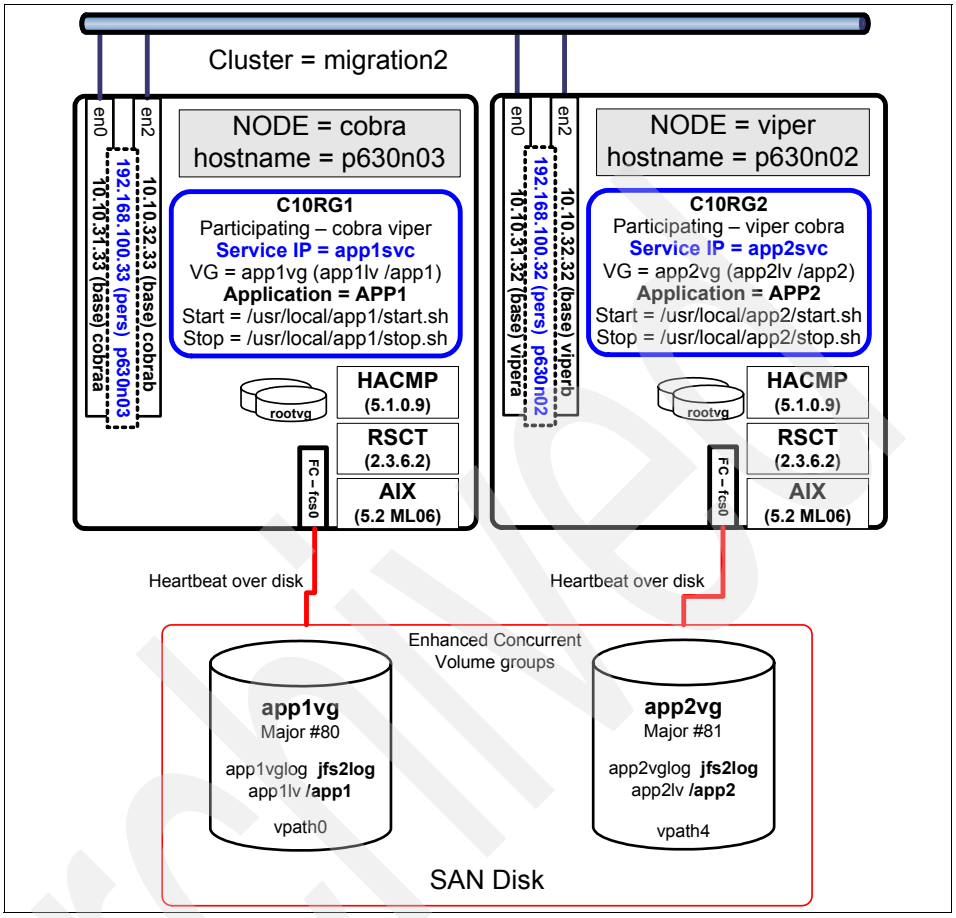

Figure 5-3 Scenario 2 - HA 5.1 to HA 5.3 migration environment

The topology was configured to use the default IP Aliasing behavior. In each node we configured persistent IP addresses in the same subnet as our service addresses. Our storage consisted of switch attached ESS LUNs (2105-800) along with the host attachment script (ibm2105.rte) and subsystem device driver. Each node was configured to host its own resource group, each with its own volume group, service IP address, and application server. We configured our volume groups as enhanced concurrent in order to setup our disk heartbeating non-IP network.

Rolling migration - scenario 2

We tested the cluster stability and fallover functionality before commencing our migration tests. The first migration method that we tested for this two node cluster was a rolling migration.

We took the following preliminary steps:

1. From the working cluster (all nodes active) we saved a snapshot.

Attention: Do not save the snapshot to /tmp as the contents of /tmp are removed during the AIX migration. Consider saving it into another directory or another server.

Save any application scripts to a safe directory.

2. Took a mksysb.
3. Created an alt_disk_install as a means to revert back to the previous configuration. This was done in an effort to speed up the retesting.

Migration scenario

We selected to upgrade node **cobra** first. To do this we took the following steps:

1. Stopped HACMP on node **cobra** graceful with takeover.
   - C1ORG1 moved to node **viper**

     Checked with **/usr/es/sbin/cluster/utilities/clRGinfo**
2. Initiated the AIX 5.3/ RSCT migration via NIM code along with latest fixes:
   - smit remove sdd - must remove and replace.
     - **stopsrc -s sddsrv**
     - **rmdev -dl dpo -R**
     - Uninstalled SDD 5.2 driver (devices.sdd.52.rte 1.6.0.2)
     - Installed SDD 5.3 driver (devices.sdd.53.rte 1.6.0.0 & 1.6.0.2)
3. Ran smit update_all to load the HA 5.3 filesets.
4. Rebooted node **cobra.**
5. Reintegrated node **cobra** into cluster by starting up cluster services.
   - Our resource policy was set to fallback to the higher priority node. As a result C1ORG1 moved back to node **cobra**.

Attention: During this stage the HACMP ODM classes on cobra had not been converted to reflect that it was running at the new release, however the filesets reflected the HA 5.3 level.

At this point the cluster is in a mixed mode or what we call hybrid state.

6. Stopped HACMP on node **viper** graceful with takeover.
   - Resource group C1ORG2 moved over to node **cobra**.

7. Repeated the AIX/RSCT and HACMP installation steps on the node.
8. Rebooted node **viper.**
9. Reintegrated node **viper** into the cluster by starting up cluster services.

Rolling migration results - scenario 2

The overall rolling migration test was a success. As in scenario 1, the AIX migration was the slowest portion of this upgrade. After it completed we tested the cluster fallover functionality and experienced no problems. The reintegration of the last cluster node was successful and we were able to confirm that the ODM changes were made reflecting the new release of HACMP.

For testing purposes we opted to disrupt the migration while the nodes were in a mixed mode. We halted node **cobra** while it was running HA 5.3 and was hosting both resource groups. At this time node **viper** was in the process of getting installed with HACMP 5.3. After rebooting **cobra** and restarting cluster services it acquired its own resource group (C1ORG1). We then manually brought viper's resource group online by going into:

```
#smit hacmp > System Management (C-SPOC) > HACMP Resource Group and Application
Management > Bring a Resource Group online > selected C1ORG2 and pressed Enter.
```

The resource group came online without any problems. After this we completed the AIX and HACMP upgrades on the second node and were able to successfully integrate into the cluster without any disruption.

We utilized the cluster test tool upon completion to run a variety of tests in one operation and identified no problems. We simulated the migration test again and experienced no problems with the cluster functionality.

Snapshot migration - scenario 2

For the same 2-node cluster we also tested the snapshot conversion method. We used our alternate disk installation images to revert back to our previous environment, and restarted the cluster services on both nodes running HA 5.1. We proceeded our testing with the following steps:

For this test we followed the following steps:

1. Stopped HACMP on all nodes: `cobra viper`
2. Ran `smit remove` and deinstalled all `cluster.*` filesets from all of the nodes.
3. Migrated the AIX/RSCT code via NIM
4. Installed HACMP packages with current PTF levels onto all nodes.
5. Rebooted all cluster nodes.

6. Copied the `snapshot.odm` file that we previously saved in the rolling migration scenario into /usr/es/sbin/cluster/utilities/snapshot.odm
7. Ran the following to convert the snapshot:

   ```
   #/usr/es/sbin/cluster/conversion/clconvert_snapshot -v 5.1 -s snapshot.odm
   ```

8. Applied the snapshot by following running `smit hacmp > Extended Configuration > Snapshot Configuration > Apply a Cluster Snapshot >` selected snapshot and pressing Enter.
9. Started cluster services one node at a time.

Snapshot migration results - scenario 2

The overall snapshot migration scenario was also a success. The conversion of the snapshot file took seconds and it also applied very quickly across both nodes. When it completed we confirmed that the HACMP ODM stanzas were converted successfully by checking for the cluster version in the `HACMPcluster` and `HACMPnode` object classes.

We utilized the cluster test tool upon completion to run multiple tests in one shot. No problems were discovered with the cluster functionality after the upgrade.

We find that if your environment can sustain a cluster wide outage this method is quick and reliable and avoids ever running nodes in the cluster in a mixed mode. Utilizing the snapshot to bring back your HACMP configuration after all nodes are upgraded will help you avoid the potential problems that could occur during a rolling migration. Refer back to “Considerations” on page 271 in this chapter for more details on potential problems.

5.5.3 Scenario 3 - AIX 5.2 and HA 5.2

Our third test scenario consisted of the migration of an AIX 5.2 / HA 5.2 cluster configured with two active nodes and one idle standby node, to AIX 5.3 / HA 5.3. For our configuration we used three equal p690 LPARs (7040-681). Below, on Figure 5-4 on page 285 is a diagram of the configuration used:

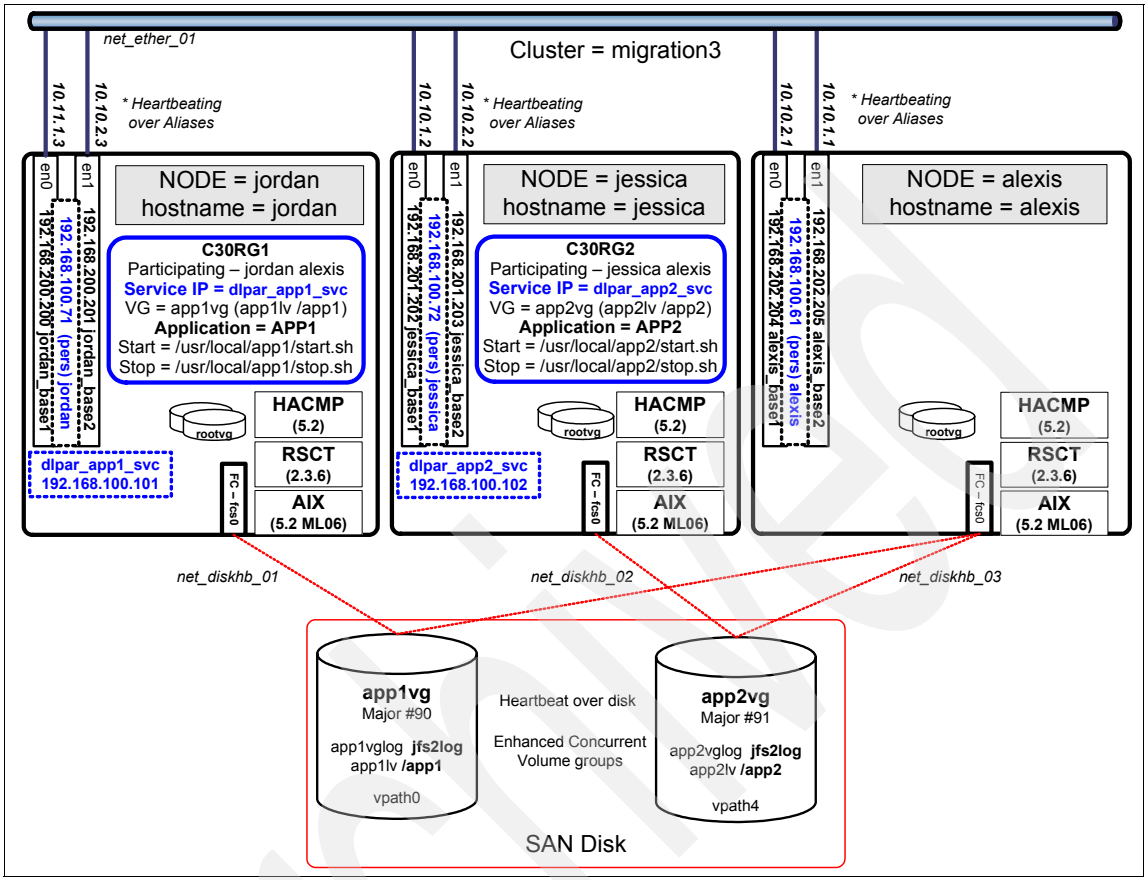

Figure 5-4 Scenario 3 - HA 5.2 to HA 5.3 migration environment

The topology was configured to use IP aliasing along with heartbeating over aliases. This was done in order to confirm that the upgrade would retain the 10.10.1.1 network offset and the corresponding aliases and that no problems were experienced with the cluster functionality thereafter.

Note: Remember that using Heartbeating over Aliases for your network topology the base addresses are not monitored by HACMP.

We configured the base addresses for each node in their own subnet and placed all service and persistent IPs on a different subnet range. This IP configuration was used to simulate a campus environment where the machines may be separated by some distance and may use different subnets and to also prove that the base addresses are not monitored by HACMP heartbeat traffic.

Our storage consisted of switch attached ESS LUNs (2105-800) using the host attachment script and the subsystem device driver for load-balancing and multipathing. We used enhanced concurrent volume groups in order to set up our three disk heartbeating non-IP networks.

Rolling migration - scenario 3

For this scenario we used a cluster on an environment the DLPAR functionality in order to ensure that everything still worked after the upgrade. We took the following preliminary steps:

1. From the working cluster (all nodes active) we saved a snapshot.

Attention: Do not save the snapshot to /tmp as the contents of /tmp are removed during the AIX migration. Consider saving it into another directory or another server.

Save any application scripts to a safe directory.

2. Took a mksysb.
3. Created an alt_disk_install as a means to revert back to the previous configuration. This was done in an effort to speed up the retesting.

Migration scenario

We proceeded to upgrade the standby node **alexis** first to minimize the amount of downtime on the nodes hosting resource groups. To accomplish this we did the following:

1. Stopped HACMP on node **alexis** with the graceful option.
2. Initiated the NIM install of AIX5.3/RSCT base code and latest fixes.
   - Removed and replaced SDD with current AIX 5.3 driver:
     - **stopsrc -s sddsrv**
     - **rmdev -dl dpo -R**
     - uninstall 5.2 SDD with `smitty remove` (`devices.sdd.52.rte 1.6.0.2`)
     - installed 5.3 SDD (`devices.sdd.53.rte 1.6.0.0 & 1.6.0.2 PTF`)
3. Ran `smit update_all` to load the HACMP 5.3 filesets.
4. Rebooted node **alexis**.
   - Verified AIX install (lppchk -l / -c / -v, instfix, oslevel, errpt)
   - lslpp -l | grep cluster => All 5.2 gone (check for any missed filesets)
   - `odmget HACMPcluster` => version still shows version = 7
5. Reintegrated node **alexis** by starting up cluster services.

Attention: Synchronizing the cluster while in this hybrid state would break the migration.

6. Stopped HACMP on node **jordan** graceful with takeover.
7. Repeated the steps 2-5 on node **jordan**.
8. Stopped HACMP on node **jessica** graceful with takeover.
9. Repeated the steps 2-5 on node **jessica**.

Rolling migration results - scenario 3

As in the two previous scenarios we did not see any problems with the HACMP portion of the rolling migration. When the last node entered the cluster the conversion of the ODM stanzas was made and all nodes now showed to be at `cluster_version = 8`, as we would expect. Refer to table 6-2 for HACMP release version numbers.

When we initially set up the topology for this cluster we configured the base, persistent and service IP addresses all into the same subnet. Although feasible, this topology poses certain problems in a NIM (Network Install Manager) environment, due to the fact that NIM is not able to handle the persistent aliases (NFS mount failures).

Thus we advise caution when implementing HACMP in a NIM environment (this is also valid for a Cluster Systems Management - CSM environment, which uses NIM for installing and updating software on the managed nodes). The following.Example 5-1 presents our hosts file configuration (which created problems with NIM):

Example 5-1 IP name resolution in our environment

```
_:>#  more /etc/hosts
192.168.100.200  jordan_base1
192.168.100.201  jordan_base2
192.168.100.202  jessica_base1
192.168.100.203  jessica_base2
192.168.100.204  alexis_base1
192.168.100.205  alexis_base2
192.168.100.71   p690_1_lpar1   #persistent IP jordan
192.168.100.72   p690_1_lpar2   #persistent IP jessica
192.168.100.61   p690_2_lpar1   #persistent IP alexis
192.168.100.101  dlpar_app1_svc
192.168.100.102  dlpar_app1_svc
```

Note: The configuration shown in Example 5-1is based on a class C network mask (`255.255.255.0`). As an alternative we changed our topology to the one in Figure 5-4 on page 285 and configured our base addresses in a separate subnet from the service and persistent IPs

Snapshot migration - scenario 3

For the same 3-node cluster we also tested the snapshot conversion method. We used our alternate disk installation images to revert back to our previous environment, and restarted the cluster services on both nodes running HACMP 5.2.

For this test we followed the following steps:

1. Stopped HACMP on all nodes: **`alexis jordan jessica`**
2. Ran `smit remove` and deinstalled all `cluster.*` filesets from all of the nodes.
3. Migrated the AIX/RSCT code via NIM.
4. Installed the HACMP packages with current PTF levels onto all nodes.
5. Rebooted all cluster nodes.
6. Copied the snapshot.odm file that we previously saved in the rolling migration scenario into `/usr/es/sbin/cluster/utilities/snapshot.odm`
7. Ran the following to convert the snapshot:

   ```
   #/usr/es/sbin/cluster/conversion/clconvert_snapshot -v 5.2 -s snapshot.odm
   ```

8. Applied the snapshot by following running `smit hacmp > Extended Configuration > Snapshot Configuration > Apply a Cluster Snapshot > selected snapshot` and pressing Enter.
9. Started cluster services one node at a time.

Snapshot migration results - scenario 3

The overall snapshot migration scenario was also a success. The conversion of the snapshot file took about one minute and it also applied very quickly across all 3 nodes. We confirmed that the ODM stanzas were converted successfully.

We found that if your environment can sustain a cluster wide outage this method is quick and reliable and avoids ever running nodes in the cluster in a mixed mode. Utilizing the snapshot to bring back your HACMP configuration after all the nodes are upgraded will help you avoid the potential problems that could occur during a rolling migration. Refer back to considerations section in this chapter for more details.

5.6 Post migration steps

The following are the recommended actions to take after a migration.

- Check for any HACMP filesets still installed from the previous release. It is easy to leave behind old documentation filesets, which although will not cause any problems should be upgraded as well.
- After completing either of the migration paths you should verify and synchronize the cluster configuration.
- Test the cluster fallover and recovery behavior after any migration. The cluster test tool gives you the ability to run multiple tests in one run and may be customized to include additional tests.
- Note that the ~/.rhosts file after 4.5 is no longer needed (nor used) by HACMP. If not required in your environment remember to remove it.

Post-migration tip:

We used a NIM server for our AIX/HACMP migrations. We used the persistent IP for each NIM client machine definition. As a result, we discovered that the installation set the persistent IP label as the base address for one of the interfaces and removed the base address. The hostname was also set to the machine name specified in the NIM client machine definition.

We corrected the changes by doing the following:

```
1. smitty chinet => to hardset the interface back to base address

2. Commented out the lines added in the /etc/rc.net file:
      /bin/hostname <hostname>
      /usr/sbin/ifconfig <en#> inet <IP> netmask 255.255.255.0

3. hostname <name> => set hostname back
```

If using NIM, make sure that you check for these changes otherwise you will experience mixed results within HACMP after starting cluster services.

This issue can be avoided by using NIM customization (provided that you write your own customization script).

5.7 Troubleshooting a failed migration

Migrating a cluster is an integral part of cluster administration. In order to complete a successful upgrade be sure to carefully plan and review your migration path. While we discovered no problems with our migration testing we

realize that should a problem occur it is useful to have steps documented that outline the way to return back to a stable state. This section was written with that in mind.

5.7.1 Backing out of a failed migration

Following the steps in the pre-migration checklist is the best way to avoid running into a migration problem. However, should you encounter a problem where the nodes fail to integrate into the cluster or the nodes halt at integration, consider retracing your migration steps and follow the suggestions below to revert back to your old configuration.

Attention: If you encounter a problem, depending on what stage of the migration you are in, you may not have to restore the entire cluster.

If the HACMP cluster migration that you are performing involves the upgrade of AIX and RSCT you cluster will be running mixed or in a hybrid state as soon as the first upgraded node is reintegrated into the cluster. Should this node or any remaining nodes in the cluster fail to integrate into the cluster there is a chance that your cluster migration will not be easily corrected and that you will need to revert back to your old configuration.

Once you are in this stage your options are limited to the following:

Option 1 - Power up the node and attempt to identify and troubleshoot the problem.

Option 2 - Revert back to your old configuration.

Option 3 - Deinstall the old HACMP software, install the new code and follow the snapshot migration path (assuming that you have a snapshot available).

Option 1 - Troubleshooting the migration failure

In the event that the node halted, power up the node. With the machine powered up and no HACMP active on it you can try and identify the cause of the clexit by reviewing the some of the different cluster and system logs. We recommend reviewing the following:

- Check the error report (**`errpt -a| more`**)

  Record the exact time of the failure based on the errors logged. Analyze any recent pertinent errors and try identify any abnormal daemon exits or CORE files generated.

- Check the `/usr/es/adm/cluster.log` file for any related cluster events.

Analyze the log for any cluster events taking place during the time of the failure. Remember to check the logs on the other cluster nodes for any possible differences.

- Check `/tmp/clstrmgr.debug` for any messages about the `clstrmgrES` exit.

  If the cluster manager exits abnormally a machine will typically halt. The majority of the time some type of an exit message will be logged at the end of this file. The message may give you or support representatives an idea as to the cause of the failure.

- Based on the **`errpt`** messages you may consider analyzing the group services log files in `/var/ha/log` to try and identify a problem.

  An analysis of the logs during the time frame that the problem occurred may help you identify the cause of an abnormal daemon exit or any other problems recorded.

Generally an in depth review of these log files should be performed by an experienced HACMP administrator or by an IBM software support representative. Keep in mind that the review of these logs during the migration process will prevent you from continuing and will delay the overall migration process.

Tip: If contacting IBM software support consider collecting a **`snap -e`** from the node that experienced the failure in order to expedite the analysis and resolution of your problem.

Option 2 - Reverting back to the old configuration

If your migration involves only the upgrade of the HACMP software and you experience a failure in the middle of a rolling migration, the quickest method to revert back to your previous configuration is to deinstall the HACMP filesets from the nodes that have been upgraded thus far. Then reinstall the old HACMP version onto those nodes and synchronize the configuration back from the remaining active node(s).

If you are upgrading the AIX and RSCT software along with HACMP reverting back to the old configuration will be a bit more difficult. You will need to stop HACMP on all upgraded nodes and revert them back to the previous version of AIX either via mksysb or alternate disk install taken prior to the start of the migration. After restoring the nodes back from your backup source you should perform a verification and synchronization from the currently active node(s). Upon a successful completion you can reintegrate the restored nodes back into the cluster.

After successfully restoring the cluster back to the original code version you should review your migration steps and ensure that you have reviewed and performed the steps discussed in the section about “Prerequisites” on page 268.

Option 3 - Deinstall HACMP and perform snapshot migration

In order to take advantage of this option you need to have a valid snapshot from when the cluster was at the version of HACMP that you are upgrading from. The steps involved are basically the same as the general snapshot migration steps covered in “General migration steps” on page 273.

1. Stop cluster services on all cluster nodes.
2. Upgrade the AIX and RSCT software on all nodes (if necessary).
3. Deinstall the current version of HACMP and install HACMP 5.3 on all nodes (including latest PTFs if available).
4. Reboot the nodes.
5. Convert the snapshot.
6. Apply the snapshot.
7. Start cluster services on one node at a time.
8. Verify and synchronize the cluster.

Note: The conversion to HACMP 5.3 was done automatically in all of our migration test scenarios and no problems were encountered. If you are having problems beyond those mentioned in this section consider contacting IBM Software Support for further assistance.

5.7.2 Review the cluster version in the HACMP ODM

To review the version of your cluster:

1. Run **`odmget HACMPcluster`** or **`odmget HACMPnode`**. It is very important to note that after the migration to HACMP 5.3 is completed the version level should be equal to 8.

Table 5-2 HACMP cluster version ODM stanzas

HACMP Version	In HACMPcluster	In HACMPnode
HACMP 5.3	cluster_version = 8	version = 8
HACMP 5.2	cluster_version = 7	version = 7
HACMP 5.1	cluster_version = 6	version = 6
HACMP 4.5	cluster_version = 5	version = 5

HACMP Version	In HACMPcluster	In HACMPnode
HACMP 4.4.1	cluster_version = 4	version = 4

If the version was not updated in a rolling migration after the last node integrated into the cluster check the `clconvert.log` for any migration problems reported. Only as a last resort should the ODM values be modified by an administrator or IBM software support personnel.

Attention: Although this is not supported, if you are still considering manually editing the ODM values to correct a discrepancy, you should call IBM software support before proceeding.

5.7.3 Troubleshooting stalled snapshot application

In some instances using the snapshot migration you may encounter that the verification fails and the snapshot fails to apply. If you apply a snapshot and see an error, review the log files and check to see if it can be corrected by the HACMP 5.3 verification utility. Be advised that even if the apply fails some of the configuration may be updated into the HACMP ODM classes.

If the error meets the criteria of a discrepancy that the auto-correct feature in a verification will resolve, you may continue the upgrade process by applying the snapshot with the forced option. Upon completion, run the cluster synchronization and verification process with the option `Automatically Correct Errors during the Cluster Verification` set to `Interactively`.

Attention: Only use the force option if you are sure that the error encountered can be automatically corrected.

You may see the following warnings and errors:

```
WARNING: “ The NFS mount/Filesystem specified for resource group rg1 is
using incorrect syntax for specifying an NFS cross mount: /mnt/fs1”.
```

```
ERROR: “ Disk Heartbeat Networks have been defined, but no Disk
Heartbeat Devices. You must configure one device for each node in order
for a Disk Heartbeat network to function”.
```

In these cases, apply the snapshot forcefully to continue the upgrade process to HACMP 5.3. Although the apply fails the cluster remains intact. In this instance force applying the snapshot is safe.

5.7.4 DARE error during synchronization

If after the migration completes you try to synchronize and see the following message:

```
cldare: Migration from HACMPversion to HACMP 5.3 Detected. cldare cannot be run
until migration has completed.
```

You should first check the `clconvert.log` for any failures and proceed to the following steps:

1. Enter `smit hacmp`.
2. Go to `Problem Determination Tools`.
3. Select `Restore HACMP Configuration Database from Active Configuration`.

If this still does not resolve the issue you may check for zero-length `/usr/es/sbin/cluster/.esmig` lock files on each of the cluster nodes. These files will normally be automatically removed when the last node integrates into the cluster.

Attention: Removing these files will remove the lock. However, if these files were not removed through the standard procedure, something else may have gone wrong and you should consider contacting IBM software support before proceeding.

5.7.5 Error: “config_too_long” during migration

If the cluster was in working order before starting your migration process it is unlikely that your cluster will enter recovery mode and encounter a `config_too_long` message. In the event that this occurs consider the following HACMP backup behavior:

Various files are saved in the `/usr/lpp/save.config` directory during the upgrade process, including:

```
/usr/lpp/save.config/usr/es/sbin/cluster/events/node_up.rp
/usr/lpp/save.config/usr/es/sbin/cluster/events/node_down.rp
```

If upgrading from HACMP/ES 4.5 the following event is also saved:

```
/usr/lpp/save.config/usr/es/sbin/cluster/events/rg_move.rp
```

If after the last node integrates into the cluster at the end of the migration the ODM stanzas are not automatically updated, you could potentially encounter a `config_too_long` since the processing within the cluster events will be unable to find the original path to these events /usr/es/sbin/cluster/events.

After checking the `clconvert.log` file for any migration failures a potential work-around is to remove the `/usr/lpp/save.config` portion out of the stanzas. This operation should only be performed as a last resort under the supervision of IBM support personnel.

Important: If these paths were not automatically corrected during the migration you may potentially have other things that failed to convert.

In this situation consider contacting IBM software support.

Part 3

Cluster scenarios and administration

Part 3 presents cluster administrative tasks and scenarios for modifying and maintaining an HACMP cluster.

The following topics are discussed:

- Scenario: Adding two nodes to a cluster
- Cluster maintenance
- Managing your cluster
- Cluster security

6

Scenario: Adding two nodes to a cluster

This chapter provides an example of how to change the cluster configuration from two nodes to four nodes. We show you step-by-step how to add new nodes and resource groups, and test the cluster:

- Description of the original two-node cluster configuration, using tables from HACMP planning worksheets
- New four-node cluster configuration, emphasizing the additional devices and resources
- Detailed description of the disk heartbeat configuration
- Detailed, step-by-step guide, of the online cluster reconfiguration

We assume that you already read how to plan, install and configure a HACMP cluster and understand Chapter 3, “Planning” on page 135 and Chapter 4, “Cluster installation scenarios” on page 231. Also, consult the HACMP administration manual when necessary.

In this chapter we do not describe each HACMP function, rather we focus on how you can change your cluster configuration online. We want to encourage you to safely modify your HACMP configuration.

6.1 Two-node configuration

Here we describe the configuration of our original, two-node cluster, using tables similar to HACMP cluster planning worksheet.

Figure 6-1 shows the cluster configuration. The main features of the two-node cluster:

- Two nodes.
- One public ethernet network (net_ether_01) with IP address takeover via aliases.
- Two base IP addresses and one service address per node.
- One disk heartbeat network (net_diskhb_01) for non IP connection.
- One application runs on each node in mutual takeover configuration.
- One shared volume group per node.
- Redundant Fibre Channel connections.

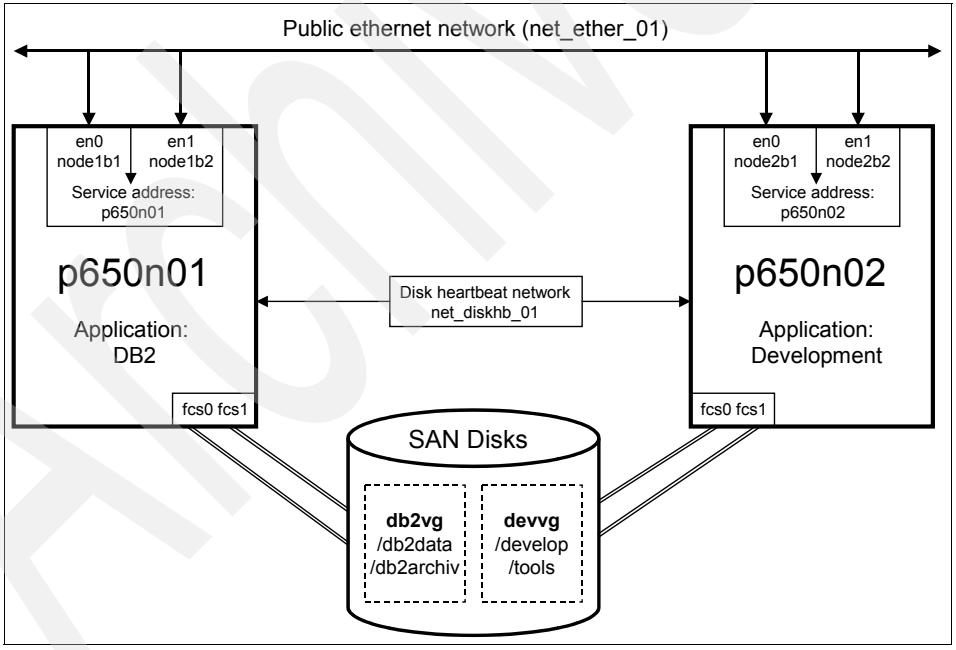

Figure 6-1 Two-node configuration

6.1.1 Cluster topology

The cluster topology configuration details:

Cluster name: clto24

Node1 name: p650n01

Node2 name: p650n02

We have one IP network and one non-IP network for disk heartbeat. We use IP address takeover via aliasing. Table 6-1 shows the network configuration.

Table 6-1 Network configuration

Network name	Network type	Netmask	IPAT via aliases
net_ether_01	ethernet (public)	255.255.255.0	enable
net_diskhb_01	disk heartbeat	N/A	N/A

Table 6-2 shows the network interface configuration. On each node we have two network adapters and each interface has one base IP address. We have two service addresses, one for each resource group.

Table 6-2 Network interfaces

IP Label	Network interface	Interface function	IP Address
node1b1	en0	base on p650n01	192.168.100.51
node1b2	en1	base on p650n01	192.168.145.51
p650n01		service	192.168.147.51
node2b1	en0	base on p650n02	192.168.100.52
node2b2	en1	base on p650n02	192.168.145.52
p650n02		service	192.168.147.52

The cluster has one disk heartbeat network, which includes one enhanced concurrent capable volume group. This disk is used only for heartbeat. See Table 6-3.

Table 6-3 Disk heartbeat network configuration

Network name	First node	Second node	Device name	VG name
net_diskhb_01	p650n01	p650n02	vpath4	c12vg

Note: For consistency, we have the same numbering of vpath devices on all nodes, that's why we have the same vpath device name on both nodes.

6.1.2 Cluster resources

The following pages describe the resources required to run our applications in HACMP.

Our cluster has two applications. Under normal circumstances, DB2 runs on p650n01 and Development runs on p650n02. They are set up in a traditional mutual takeover configuration. Each application has its own volume group: db2vg and devvg respectively.

Shared volume group and filesystem configuration

Table 6-4 shows the shared volume group configuration.

Table 6-4 Shared volume group configuration

Volume group name	Vpath on p650n01	Vpath on p650n02	Major number
db2vg	vpath0	vpath0	45
devvg	vpath1	vpath1	46

Table 6-5 shows the detailed file system configuration of db2vg volume group.

Table 6-5 Shared volume filesystem configuration 1.

Volume group name	db2vg
Filesystem #1	/db2data
Filesystem #1 logical volume	/dev/db2datalv
Filesystem #1 jfs log device	/dev/db2loglv
Filesysten2	/db2archiv
Filesystem #2 logical volume	/dev/db2arclv
Filesystem #2 jfs log device	/dev/db2loglv

Table 6-6 shows the file system configuration of devvg volume group.

Table 6-6 Shared volume filesystem configuration 2.

Volume group name	devvg
Filesystem #1	/develop
Filesystem #1 logical volume	/dev/devlv
Filesystem #1 jfs log device	Inline log

Volume group name	devvg
Filesysten2	/tools
Filesystem #2 logical volume	/dev/toolslv
Filesystem #2 jfs log device	inline log

Application servers

Our cluster has two application servers: DB2 and Development. Table 6-7 shows the start and stop scripts for the application servers.

Table 6-7 Application server configuration

Application server name	Start script	Stop script
DB2	/usr/ha/start.db2	/usr/ha/stop.db2
Development	/usr/ha/start.development	/usr/ha/stop.development

Resource groups

We created two resource groups. Each resource group contains one application server, one service IP address and the required volume group. Table 6-8 shows the detailed resource group configuration.

Table 6-8 Resource group configuration

Resource name	rg1	rg2
Participating node names	p650n01, p650n02	p650n02, p650n01
Startup policy	Online On Home Node Only	Online On Home Node Only
Fallover policy	Fallover To Next Priority Node In The List	Fallover To Next Priority Node In The List
Fallback policy	Fallback To Higher Priority Node In The List	Fallback To Higher Priority Node In The List
Dynamic node priority policy		
Processing order	Parallel	Parallel
Service IP Labels	p650n01	p650n02
Application servers	db2	development
Volume groups	db2vg	devvg

Resource name	rg1	rg2
Use forced varyon of volume groups, if necessary	false	false
Automatically import volume groups	false	false
Filesystems	ALL	ALL
Filesystem consistency check	fsck	fsck
Filesystem recovery method	sequential	sequential
Filesystems mounted before IP configured	false	false
Filesystems/Directories to Export		
Filesystems/Directories to NFS Mount		
Network For NFS Mount		
Primary Workload Manager Class		

6.2 Four-node configuration

The new four-node cluster configuration is discussed here. We like to add two new nodes and two new applications with their required resources to the cluster. We use the following information during the reconfiguration step as a reference.

Figure 6-2 on page 305 shows the new cluster configuration. The main features of the cluster:

- Four nodes.
- One public ethernet network (net_ether_01) with IP address takeover via aliases.
- Two base IP addresses and one service address per node.
- Four disk heartbeat networks for non IP connection.
- By default one application runs on each node.
- One shared volume group per node.

- Redundant Fibre Channel connections.

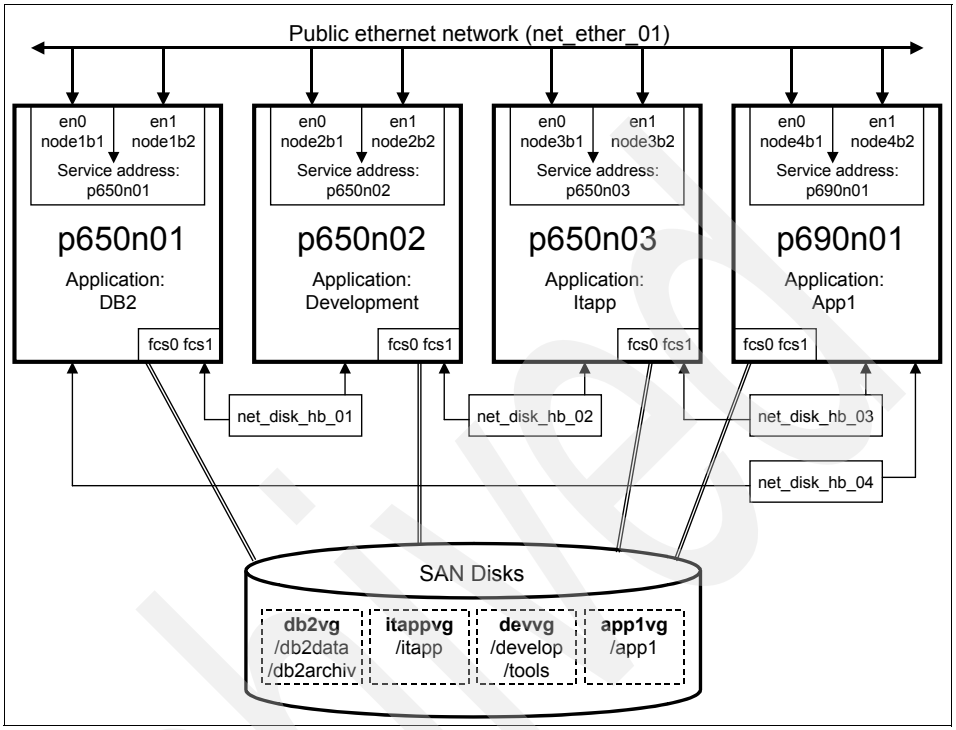

Figure 6-2 Four-node configuration

6.2.1 Cluster topology

The detailed cluster topology configuration:

Cluster name: clto24

Node1 name: p650n01

Node2 name: p650n02

Node3 name: p650n03

Node4 name: p690n01

See Table 6-9 on page 306 for the network configuration. The new cluster still has only one ethernet network, using IP address takeover via aliases. The change is that now we have four disk heartbeat networks. See 6.2.2, “Disk heartbeat configuration” on page 306 for the detailed disk heartbeat network configuration.

Table 6-9 Network configuration

Network name	Network type	Netmask	IPAT via aliases
net_ether_01	ethernet (public)	255.255.255.0	enable
net_diskhb_01	disk heartbeat	N/A	N/A
net_diskhb_02	disk heartbeat	N/A	N/A
net_diskhb_03	disk heartbeat	N/A	N/A
net_diskhb_04	disk heartbeat	N/A	N/A

The new nodes have similar IP network configuration likes the old ones: they have two interface cards with base addresses. We also have two new service addresses for the new resource groups. See Table 6-10.

Table 6-10 Network interfaces

IP Label	Network interface	Interface function	IP Address
node1b1	en0	base on p650n01	192.168.100.51
node1b2	en1	base on p650n01	192.168.145.51
p650n01		service	192.168.147.51
node2b1	en0	base on p650n02	192.168.100.52
node2b2	en1	base on p650n02	192.168.145.52
p650n02		service	192.168.147.52
node3b1	en0	base on p650n03	192.168.100.53
node3b2	en1	base on p650n03	192.168.145.53
p650n03		service	192.168.147.53
node4b1	en0	base on p690n01	192.168.100.63
node4b2	en1	base on p690n01	192.168.145.63
p690n01		service	192.168.147.63

6.2.2 Disk heartbeat configuration

The four-node cluster requires four disk heartbeat networks. Each network connects two nodes through a vpath device. The four networks consist a ring between the nodes. In case of an ethernet network failure the nodes can pass

heartbeat packets to each other through this ring: e.g: node1 can send data to node3. On Figure 6-3 we show you the disk heartbeat network topology.

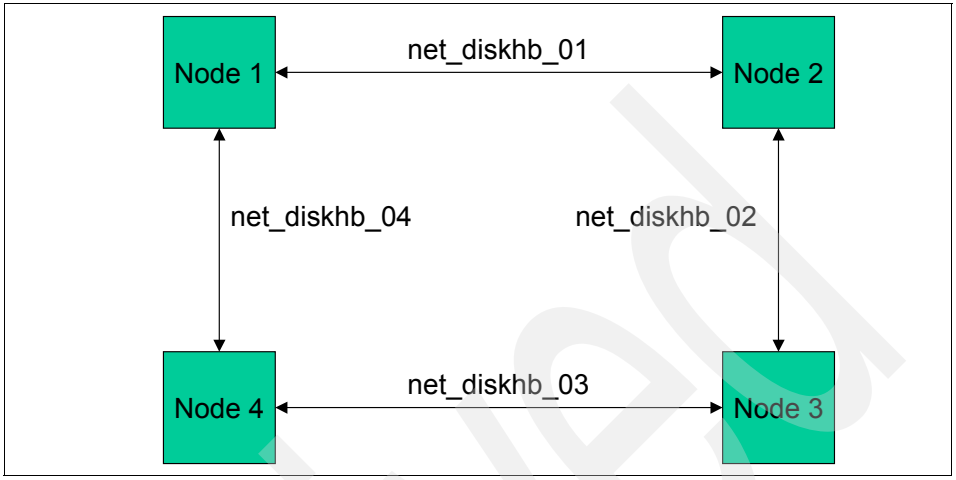

Figure 6-3 Disk heartbeat configuration

The detailed disk heartbeat network configuration can be seen on Table 6-11. Note: we have the same numbering of vpath devices on all nodes, that's why we have the same vpath device name on both node pair.

Table 6-11 Disk heartbeat network configuration

Network name	Node1 / device	Node2 / device	VG name
net_diskhb_01	p650n01 / vpath4	p650n02 / vpath4	c12vg
net_diskhb_02	p650n02 / vpath5	p650n03 / vpath5	c23vg
net_diskhb_03	p650n03 / vpath6	p690n01 / vpath6	c34vg
net_diskhb_04	p690n01 / vpath7	p650n01 / vpath7	c41vg

6.2.3 Resource group configuration

The new cluster has two new applications: Itapp runs on p650n03, its backup node is p690n01. App1's home node is p690n01 and it uses dynamic node priority for fallover to the least utilized node.

We do not change DB2 and Development application servers. Under normal circumstances DB2 runs on p650n01 and Development runs on p650n02. They are set up in a traditional mutual takeover configuration.

The following pages describe the detailed HACMP resource group configuration, including shared storage and resource group information.

Shared volume group and filesystem configuration

Table 6-12 shows the shared volume group configuration.

Table 6-12 Shared volume group configuration

Volume group name	p650n01	p650n02	p650n03	p690n01	Major number
db2vg	vpath0	vpath0	vpath0	vpath0	45
devvg	vpath1	vpath1	vpath1	vpath1	46
itappvg	vpath2	vpath2	vpath2	vpath2	47
app1vg	vpath3	vpath3	vpath3	vpath3	48

Table 6-13 shows the detailed file system configuration of db2vg volume group.

Table 6-13 Shared volume filesystem configuration 1.

Volume group name	db2vg
Filesystem #1	/db2data
Filesystem #1 logical volume	/dev/db2datalv
Filesystem #1 jfs log device	/dev/db2loglv
Filesysten2	/db2archiv
Filesystem #2 logical volume	/dev/db2arclv
Filesystem #2 jfs log device	/dev/db2loglv

Table 6-14 shows the file system configuration of devvg volume group.

Table 6-14 Shared volume filesystem configuration 2.

Volume group name	devvg
Filesystem #1	/develop
Filesystem #1 logical volume	/dev/devlv
Filesystem #1 jfs log device	Inline log
Filesysten2	/tools
Filesystem #2 logical volume	/dev/toolslv

Volume group name	devvg
Filesystem #2 jfs log device	inline log

Table 6-15 shows the file system configuration of itappvg volume group.

Table 6-15 Shared volume filesystem configuration 3.

Volume group name	itappvg
Filesystem #1	/itapp
Filesystem #1 logical volume	/dev/itapplv
Filesystem #1 jfs log device	Inline log

Table 6-16 shows the file system configuration of app1vg volume group.

Table 6-16 Shared volume filesystem configuration 4.

Volume group name	app1vg
Filesystem #1	/app1
Filesystem #1 logical volume	/dev/app1lv
Filesystem #1 jfs log device	Inline log

Application servers

Our cluster has two new application server: Itapp and App1. Table 6-17 shows the start and stop scripts for the application servers.

Table 6-17 Application server configuration

Application server name	Start script	Stop script
DB2	/usr/ha/start.db2	/usr/ha/stop.db2
Development	/usr/ha/start.development	/usr/ha/stop.development
Itapp	/usr/ha/start.itapp	/usr/ha/stop.itapp
App1	/usr/ha/start.app1	/usr/ha/stop.app1

Resource groups

Table 6-18 on page 310 shows the detailed resource group configuration for the original resource groups rg1 and rg2.

Table 6-18 Resource group configuration 1.

Resource name	rg1	rg2
Participating node names	p650n01, p650n02	p650n02, p650n01
Startup policy	Online On Home Node Only	Online On Home Node Only
Fallover policy	Fallover To Next Priority Node In The List	Fallover To Next Priority Node In The List
Fallback policy	Fallback To Higher Priority Node In The List	Fallback To Higher Priority Node In The List
Dynamic node priority policies		
Processing order	Parallel	Parallel
Service IP Labels	p650n01	p650n02
Application servers	db2	development
Volume groups	db2vg	devvg
Use forced varyon of volume groups, if necessary	false	false
Automatically import volume groups	false	false
Filesystems	ALL	ALL
Filesystem consistency check	fsck	fsck
Filesystem recovery method	sequential	sequential
Filesystems mounted before IP configured	false	false
Filesystems/Directories to Export		
Filesystems/Directories to NFS Mount		
Network For NFS Mount		
Primary Workload Manager Class		

Table 6-19shows the detailed resource group configuration for the new resource groups. Rg4 uses dynamic fallover policy: in case of failure of node p690n01 it falls over to any of remaining three node with the highest idle CPU.

Table 6-19 Resource group configuration 2

Resource name	rg3	rg4
Participating node names	p650n03, p690n01, p650n02	p690n01, p650n03, p650n02, p650n01
Startup policy	Online On Home Node Only	Online On Home Node Only
Fallover policy	Fallover To Next Priority Node In The List	Fallover Using Dynamic Node Priority
Fallback policy	Fallback To Higher Priority Node In The List	Fallback To Higher Priority Node In The List
Dynamic node priority policy		Fallover to the node with the highest idle CPU (cl_highest_idle_cpu)
Processing order	Parallel	Parallel
Service IP Labels	p650n03	p690n01
Application servers	itapp	app1
Volume groups	itappvg	app1vg
Use forced varyon of volume groups, if necessary	false	false
Automatically import volume groups	false	false
Filesystems	ALL	ALL
Filesystem consistency check	fsck	fsck
Filesystem recovery method	sequential	sequential
Filesystems mounted before IP configured	false	false
Filesystems/Directories to Export		

Resource name	rg3	rg4
Filesystems/Directories to NFS Mount		
Network For NFS Mount		
Primary Workload Manager Class		

6.3 Reconfiguring the cluster

The following major steps required for expanding the cluster from two to four node:

1. Prerequisites
2. Adding new nodes
3. Configuring IP network topology
4. Configuring disk heartbeat networks
5. Configuring resource groups and application servers
6. Start up the new nodes
7. Test the cluster

Tip: The whole cluster configuration process can be done while your cluster is up and running.

6.3.1 Prerequisites

Perform the following prerequisite tasks on the new nodes:

- Install AIX on the new nodes
- If possible install the latest AIX maintenance level and critical fixes on all nodes. This may requires downtime on the old nodes.
- Install same level of HACMP code and fixes on the new nodes
- If possible install the latest HACMP fixes on all nodes. This may requires downtime on the first two nodes.
- Set up and test the base network interfaces on the new nodes
- Edit the /etc/hosts file, so that includes all IP address of all cluster nodes. Be sure that this file is the same on all nodes. HACMP discovery and later the IP network configuration process use this file.

- For enhanced security reason we suggest that you put all IP interfaces of all nodes to the /usr/es/sbin/cluster/etc/rhosts file.
- If you do not use HACMP file collections, then copy all required pre/post-event scripts, error notification programs, application start/stop scripts and other user-defined files to the new nodes.
- Synchronize the user and group IDs to the new nodes.
- Create the required shared volume groups and filesystems.
- Reboot the nodes, so clcomdES is restarted.

Attention: Ensure that you have the same level of AIX and the same level of HACMP on all nodes.

6.3.2 Add new nodes to the cluster

1. Start **`smit hacmp`** on one of the original nodes. We will use this node for all operation except when stated otherwise.
2. Select **Extended Configuration**.
3. Select **Extended Topology Configuration**.
4. Select **Configure HACMP Nodes**.
5. Select **Add a Node to the HACMP Cluster**.
6. Enter the new node name.
7. Enter the communication path to the new node or press F4 to get a list of IP labels from /etc/hosts file. See SMIT screenshoot on Figure 6-4 on page 314. Also refer to Table 6-10 on page 306. Add node p630n3 and p690n01 using one of their base address as a communication path to the node.

```
                        Add a Node to the HACMP Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
* Node Name                                            [p650n03]
  Communication Path to Node                           [node4b1] +



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 6-4 Add a node to the HACMP cluster

Repeat this step for each node.

Discover HACMP-related information from the nodes

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Discover HACMP-related Information from Configured Nodes**.

Now HACMP discovers all pertinent information from all nodes, including the new ones. Later we will use the information to configure the cluster.

6.3.3 Cluster topology configuration

Follow this steps to configure the cluster IP network settings for the new nodes.

Add base interfaces to the cluster topology

See the cluster IP network settings in Table 6-9 on page 306 and Table 6-10 on page 306.

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Topology Configuration**.
4. Select **Configure HACMP Communication Interfaces/Devices**.
5. Select **Add Communication Interfaces/Devices**.
6. Select **Add Discovered Communication Interface and Devices**.

7. Select **Communication Interfaces**.
8. Select your network (e.g., net_ether_01).
9. Select the communication interface to add. The base addresses of the new nodes should show up on the pop-up list. See Figure 6-5.
10. Press Enter to add the selected interfaces.

```
                Configure HACMP Communication Interfaces/Devices

Move cursor to desired item and press Enter.

  +---------------------------------------------------------------------------+
  |      Select one or more Discovered Communication Interfaces to Add        |
  |                                                                           |
  | Move cursor to desired item and press F7. Use arrow keys to scroll.       |
  |     ONE OR MORE items can be selected.                                    |
  | Press Enter AFTER making all selections.                                  |
  |                                                                           |
  |   # Node / Network                                                        |
  |   #       Interface        IP Label                      IP Address       |
  |                                                                           |
  |   # net_ether_01 / p650n03                                                |
  |           en1              node3b2                       192.168.14       |
  |           en0              node3b1                       192.168.10       |
  |                                                                           |
  |   # net_ether_01 / p690n01                                                |
  |           en0              node4b1                       192.168.10       |
  |                                                                           |
  | F1=Help                 F2=Refresh            F3=Cancel                   |
  | F7=Select               F8=Image              F10=Exit                    |
F1| Enter=Do                /=Find                n=Find Next                 |
F9+---------------------------------------------------------------------------+
```

Figure 6-5 Configure base network interface

Repeat this step for each base interface on all new nodes (node3b1, node3b2, node4b1 and node4b2). If your network interfaces do not show up in the list, then double check the network settings of the new nodes and the /etc/hosts file, then re-run the HACMP discovery.

Check the new network topology

1. Start **`smit hacmp`**.
2. Select **Extended Configuration**.
3. Select **Extended Topology Configuration**.

4. Select **Show HACMP Topology**.
5. Select **Show Cluster Topology**. Press enter to see the new cluster topology information.

Tip: You can list the cluster topology with /usr/es/sbin/cluster/utilities/cllsif utility.

If you do it right, the cluster topology must be the sane as in Table 6-10 on page 306.

Configuring disk heartbeat networks

For convenience we suggest that you create separate volume groups for the disk heartbeat, so that the disks are not used for other purposes. Also import the volume groups to each node so you know what is the disks are used for.

See section 6.2.2, “Disk heartbeat configuration” on page 306 for the detailed disk heartbeat configuration.

1. Create the required number of enhanced concurrent capable volume group on one node. In our example we need three more volume groups: c23vg, c34vg and c41vg.
   e. Start `smit lvm`.
   f. Select **Volume Groups**.
   g. Select **Add a Volume Group**.
   h. Enter the following information (See SMIT screenshoot on Figure 6-6 on page 317):
      - Volume group name
      - Physical volume names
      - **Volume group MAJOR NUMBER**: provide a number which is available on all nodes. You can check the available major numbers by `lvlstmajor` command.
      - Set **Create VG Concurrent Capable** to **enhanced concurrent**.

```
                                Add a Volume Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  VOLUME GROUP name                                    [c23vg]
  Physical partition SIZE in megabytes +
* PHYSICAL VOLUME names                                [vpath5] +
  Force the creation of a volume group?                 no +
  Activate volume group AUTOMATICALLY                   no +
    at system restart?
  Volume Group MAJOR NUMBER                            [53] +#
  Create VG Concurrent Capable?                         enhanced concurrent +
  Create a big VG format Volume Group?                  no +
  LTG Size in kbytes                                    128 +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 6-6 Create an enhanced concurrent capable volume group

Create all three concurrent volume groups: c23vg, c34vg and c41vg.

Tip: Give some kind of meaningful name for your disk heartbeat volume group. For example we use **cXYvg** naming scheme, where **X** and **Y** stand for the two nodes' number that uses this disk for heartbeat, e.g., the volume group used for disk heartbeat between node3 and node4 is called c34vg.

2. Import the new volume groups on all nodes. Be careful to use the same volume group major number everywhere, e.g., **`importvg -V 53 -y c23vg vpath5`** or use **`smit importvg`**.
3. Re-run the HACMP cluster discovery (See "Discover HACMP-related information from the nodes" on page 314).
4. Now you can add the disk heartbeat pairs to the cluster configuration:
   a. Start **`smit hacmp`**.
   b. Select **Extended Configuration**.
   c. Select **Extended Topology Configuration**.
   d. Select **Configure HACMP Communication Interfaces/Devices**.
   e. Select **Add Communication Interfaces/Devices**.

f. Select **Add Discovered Communication Interface and Devices**.

g. Select **Communication Devices**.

h. Select the disk heartbeat pairs to add (see Figure 6-7 below). SMIT provides a list of the discovered disks with enhanced concurrent capable volume group on it. Select two devices from the list. The devices should have the same physical volume ID, they should be same physical disk. When select the disk pairs always look for the PVID (on the right side of the screen), because the hdisk names may differ on the nodes. One disk can be used only in one disk heartbeat network. Also check the output of **lspv** command to identify the volume group - disk relations.

```
                Configure HACMP Communication Interfaces/Devices

Mo+--------------------------------------------------------------------------+
  | Select Point-to-Point Pair of Discovered Communication Devices to Add    |
  |                                                                          |
  | Move cursor to desired item and press F7. Use arrow keys to scroll.      |
  |     ONE OR MORE items can be selected.                                   |
  | Press Enter AFTER making all selections.                                 |
  |                                                                          |
  | [MORE...12]                                                              |
  |     p690n01                          vpath3   /dev/vpath3   000154decb5  |
  |     p650n01                          vpath6   /dev/vpath6   000197caca4  |
  |     p650n02                          vpath6   /dev/vpath6   000197caca4  |
  | >   p650n03                          vpath6   /dev/vpath6   000197caca4  |
  | >   p690n01                          vpath6   /dev/vpath6   000197caca4  |
  |     p650n01                          vpath7   /dev/vpath7   000215cad90  |
  |     p650n02                          vpath7   /dev/vpath7   000215cad90  |
  |     p650n03                          vpath7   /dev/vpath7   000215cad90  |
  |     p690n01                          vpath7   /dev/vpath7   000215cad90  |
  | [BOTTOM]                                                                 |
  |                                                                          |
  | F1=Help                F2=Refresh              F3=Cancel                 |
  | F7=Select              F8=Image                F10=Exit                  |
F1| Enter=Do               /=Find                  n=Find Next               |
F9+--------------------------------------------------------------------------+
```

Figure 6-7 Adding disk heartbeat devices

For example, the disk heartbeat network between p650n03 and p690n01 goes through c34vg. The c34vg contains one disk, that disk is called vpath6 on both nodes. It's PVID is 000197caca4d816f. So when select the device pair in the SMIT menu, first go to the disk with the right PVID, then select the corresponding nodes.

Repeat this step for each disk heartbeat device pair for the new nodes. See section 6.2.2, “Disk heartbeat configuration” on page 306 for the disk heartbeat device pair configuration.

5. Check the new network topology:
   a. Start **smit hacmp**.
   b. Select **Extended Configuration**.
   c. Select **Extended Topology Configuration**.
   d. Select **Show HACMP Topology**.
   e. Select **Show Topology Information by Network Name**.
   f. Select **Show All Networks**. See Figure 6-8 for a sample output.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.
[TOP]
Network net_diskhb_01
        NODE p650n01:
                p650n01_vpath4_01       /dev/vpath4
        NODE p650n02:
                p650n02_vpath4_01       /dev/vpath4
        NODE p650n03:
        NODE p690n01:

Network net_diskhb_02
        NODE p650n01:
        NODE p650n02:
                p650n02_vpath5_01       /dev/vpath5
        NODE p650n03:
                p650n03_vpath5_01       /dev/vpath5
        NODE p690n01:
[MORE...42]
F1=Help             F2=Refresh          F3=Cancel           F6=Command
F8=Image            F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 6-8 Checking disk heartbeat network configuration

At this point we successfully configured the new cluster topology. Please double check the new topology and the prerequisites before you run the cluster verification and synchronization.

Run cluster verification and synchronization

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Verification and Synchronization**.
4. Use the following values to start cluster verification (see Figure 6-9):
   - **Emulate or Actual:** Actual.
   - **Verify changes only?:** No.
   - **Logging:** Standard.

```
     HACMP Verification and Synchronization (Active Cluster Nodes Exist)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Emulate or Actual                                  [Actual] +
* Verify changes only?                               [No] +
* Logging                                            [Standard] +



F1=Help          F2=Refresh       F3=Cancel          F4=List
F5=Reset         F6=Command       F7=Edit            F8=Image
F9=Shell         F10=Exit         Enter=Do
```

Figure 6-9 Cluster verification and synchronization

The cluster verification checks and populates HACMP ODM classes with the new topology configuration. If you have a file collections enabled then HACMP distributes the files to the new nodes. You have to correct the possible errors and warnings before you can continue the resource group configuration. After a successful verification the new nodes and communication interfaces should show up in the **/usr/es/sbin/cluster/clstat** utility.

Attention: When the cluster is online you cannot run HACMP verification and synchronization with “correct automatically the errors found during synchronization” option enabled. Also you cannot force synchronization if the verification fails.

6.3.4 Configure resources

At this point we assume that you already done the following prerequisite tasks:

- Created your shared storage, volume groups and filesystems.
- Installed the application.
- Created and tested the application start and stop scripts.

Configure service IP address

Use network configuration data from Table 6-10 on page 306 to proceed this step.

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Resource Configuration**.
4. Select **HACMP Extended Resources Configuration**.
5. Select **Configure HACMP Service IP Labels/Addresses**.
6. Select **Add a Service IP Label/Address**.
7. Select **Configurable on Multiple Nodes**.
8. Select your network, e.g., net_ether_01.
9. Press F4 for a list of discovered addresses. Press enter to commit. See Figure 6-10 bellow.

```
   Add a Service IP Label/Address configurable on Multiple Nodes (extended)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* IP Label/Address                                   p650n03 +
* Network Name                                       net_ether_01
  Alternate Hardware Address to accompany IP Label/A []
  ddress



F1=Help            F2=Refresh          F3=Cancel           F4=List
F5=Reset           F6=Command          F7=Edit             F8=Image
F9=Shell           F10=Exit            Enter=Do
```

Figure 6-10 Configure service IP label

Repeat this step for all new service IP labels: p650n03 and p690n01.

Define the new application servers

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Resource Configuration**.
4. Select **HACMP Extended Resources Configuration**.
5. Select **Configure HACMP Applications**.
6. Select **Configure HACMP Application Servers**.
7. Select **Add an Application Server**.
8. Enter the name of the application server, and the full path name of the start and stop scripts. You can configure an application monitor method here, if you have one. See SMIT screenshoot on Figure 6-11.

```
                          Add Application Server

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Server Name                                        [app1]
* Start Script                                       [/usr/ha/start.app1]
* Stop Script                                        [/usr/ha/stop.app1]
  Application Monitor Name(s) +



F1=Help          F2=Refresh       F3=Cancel        F4=List
F5=Reset         F6=Command       F7=Edit          F8=Image
F9=Shell         F10=Exit         Enter=Do
```

Figure 6-11 Add an application server

Create itapp and app1 application servers. Refer to Table 6-17 on page 309 for the application server configuration information.

Define resource groups

Refer to Table 6-18 on page 310 and Table 6-19 on page 311 for the detailed resource group configuration parameters.

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Resource Configuration**.

4. Select **HACMP Extended Resource Group Configuration**.
5. Select **Add a Resource Group**.
6. Enter the name of the resource group. Select the participating node names (press F4 to pop-up the node names). Set the startup, fallover and fallback policy (press F4 to get availably options). See Figure 6-12.

```
                    Add a Resource Group (extended)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Resource Group Name                                [rg4]
* Participating Nodes (Default Node Priority)        [p690n01 p650n03 p650n> +

  Startup Policy                                      Online On Home Node O> +
  Fallover Policy                                     Fallover To Next Prio> +
  Fallback Policy                                     Fallback To Higher Pr> +
  +--------------------------------------------------------------------------+
  |                           Fallover Policy                                |
  |                                                                          |
  | Move cursor to desired item and press Enter.                             |
  |                                                                          |
  |   Fallover To Next Priority Node In The List                             |
  |   Fallover Using Dynamic Node Priority                                   |
  |   Bring Offline (On Error Node Only)                                     |
  |                                                                          |
  | F1=Help                 F2=Refresh              F3=Cancel                |
F1| F8=Image                F10=Exit                Enter=Do                 |
F5| /=Find                  n=Find Next                                      |
F9+--------------------------------------------------------------------------+
```

Figure 6-12 Add a resource group

Repeat this step for all new resource group (rg3 and rg4).

Define resource group attributes

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Resource Configuration**.
4. Select **HACMP Extended Resource Group Configuration**.
5. Select **Change/Show Resources and Attributes for a Resource Group**.
6. Select the resource group you like to modify from the pop-up list.

7. Configure the resource group based on the detailed resource group configuration parameters shown in Table 6-19 on page 311. See SMIT screenshoot on Figure 6-13.

```
        Change/Show All Resources and Attributes for a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                   [Entry Fields]
  Resource Group Name                                   rg4
  Participating Nodes (Default Node Priority)           p690n01 p650n02 p650n>
* Dynamic Node Priority Policy                         [cl_highest_idle_cpu] +

  Startup Policy                                        Online On Home Node O>
  Fallover Policy                                       Fallover Using Dynami>
  Fallback Policy                                       Fallback To Higher Pr>
  Fallback Timer Policy (empty is immediate)            [] +
  Service IP Labels/Addresses                           [p690n01] +
  Application Servers                                   [app1] +
  Volume Groups                                         [app1vg] +
  Use forced varyon of volume groups, if necessary      false +
  Automatically Import Volume Groups                    false +
  Filesystems (empty is ALL for VGs specified)          [] +
  Filesystems Consistency Check                         fsck +
  Filesystems Recovery Method                           sequential +
  Filesystems mounted before IP configured              false +
[MORE...15]

F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 6-13 Configure resource group attributes

Repeat this step for all new resource group (rg3 and rg4). If any of your predefined resource does not show up (e.g., shared volume group or file system) run HACMP discovery again (see “Discover HACMP-related information from the nodes” on page 314).

Run cluster verification and synchronization

At this point we successfully configured the new cluster resources. You can perform any other configuration tasks you like, e.g., adding application monitors or pre/post-events.

Please double check the new resources before running cluster verification and synchronization, because some resource groups may get activated on the original nodes. In our example, both rg3 and rg4 resource group will be online on p650n01 or p650n02 after a successful cluster synchronization.

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Verification and Synchronization**.
4. Press Enter to start the verification and synchronization.

Because two nodes are already running and the new resource configuration may affect them, please check the /tmp/hacmp.out files on the original two nodes for possible errors.

6.3.5 Start HACMP on the new nodes

If the cluster verification was successful, you can start HACMP on the new nodes.

1. Start **smit clstart**.
2. Select the new nodes and press Enter to start HACMP.

Monitor the /tmp/hacmp.out file for possible errors on all nodes during start up. If you have done properly the design phase and the configuration, then everything should be fine now.

Test the cluster

As this exercise assumes that you keep the cluster running at all times on the original two nodes, we recommend that you test the new configuration in a maintenance window, when disrupting existing applications is not critical. We suggest that you perform takeover tests as soon as possible and test as many scenarios as possible. See 3.13, “Develop a cluster test plan” on page 212.

Table 6-20 shows the resource group placement if one or two node fail. If p690n03 is down, then we cannot predict the placement of rg4, it will fallover to the least utilized node, which has the highest idle CPU. We indicated this with parentheses in table below.

Table 6-20 Resource group placement if a node fails

	p650n01	p650n02	p650n03	p690n03
all nodes up	rg1	rg2	rg3	rg4
p650n01 down	down	rg2, rg1	rg3	rg4

	p650n01	p650n02	p650n03	p690n03
p650n02 down	rg1, rg2	down	rg3	rg4
p650n03 down	rg1	rg2	down	rg4, rg3
p690n03 down	rg1 (rg4)	rg2 (rg4)	rg3 (rg4)	down
p650n03 and p690n03 down	rg1 (rg4)	rg2, rg3, (rg4)	down	down
p650n01, p650n03 and p690n03 down	down	rg2, rg1, rg3, rg4	down	down

7

Cluster maintenance

This chapter provides basic rules of thumb while planning and performing maintenance operations on an HACMP cluster. The goal is to keep the cluster applications active as much as possible. We use the functionality within HACMP and AIX 5L to perform these operations. Of course, the scenarios are not exhaustive.

In this chapter AIX common procedures of troubleshooting, including monitoring the error log, are assumed. We do not discuss how to determine what problem exists whether dealing with problems either after they are discovered, or as preventative maintenance.

7.1 Change control and testing

Change control is imperative to provide high availability via clustered systems effectively. Change control is above and beyond documented procedures. It encompasses several things and is *not* optional.

Change control includes, but is not limited to:

- Limit root access
- Thoroughly documented and *tested* procedures
- Proper planning and approval of all changes

7.1.1 Test cluster

A test cluster is important to both maintaining proper change control and to the overall success of the production cluster. Test clusters allow thorough testing of administrative and/or maintenance procedures in an effort to find problems before the problem reaches the production cluster. Test clusters should not be considered a luxury but a *must have*.

Many current HACMP customers have a test cluster, or at least started out with a test cluster. However, over time these cluster nodes have become utilized within the company in some form. To use these systems requires a scheduled maintenance window much like the production cluster. If that is the case, don't be fooled as it truly is no longer a test cluster.

A test cluster, ideally, would be at least the same AIX, HACMP, and application level as the production cluster. It is preferred to have the hardware to also be as similar as possible. In most cases it is not practical to fully mirror the production environment, especially when there are multiple production clusters. There are several things that can be done to maximize a test cluster when there are multiple clusters that have varying levels of software.

Using logical partitioning (LPAR) and multiple varying rootvg images, via **`alt_disk_install`**, have become a common practice. Lpars allow a test cluster to be easily created with very little physical resources and can even be within the same physical machine. The multi-boot option allows customers to easily change cluster environments by simply booting the partition from another image. This also allows testing of many software procedures such as:

- Applying AIX maintenance
- Applying HACMP fixes
- Applying application maintenance

This type of test cluster would require at least one disk, per image, per LPAR. For example, if the test cluster had two nodes and three different rootvg images, it would require a minimum of six hard drives. This is still far easier than having six separate nodes in three different test clusters.

The overall cost of a test cluster can be minimized further by utilizing Power5 advanced virtualization. Advanced virtualization allows several rootvg images to reside on the same physical disk. It also allows LPARs to share physical network and DASD adapters via VLAN and VSCSI. More information about these features can be found in the redbook *Advanced POWER Virtualization on IBM @server p5 Servers: Introduction and Basic Configuration,* SG24-7940.

A test cluster also allows testing of hardware maintenance procedures. These procedures include, but are not limited to:

- Machine firmware updates
- Adapter firmware updates
- Adapter replacement
- Disk replacement

Other testing can be accomplished by utilizing the cluster test tool and/or event emulation.

7.2 Starting and stopping the cluster

Starting cluster services refers to the process of starting the RSCT subsystems required by HACMP, and then the HACMP daemons that enable the coordination required between nodes in a cluster. During startup, the Cluster Manager runs the **node_up event** and resource groups are acquired. Stopping cluster services refers to stopping these same daemons on a node and may or may not cause the execution of additional HACMP scripts, depending on the type of shutdown you perform.

Important: Starting with HACMP V5.3 the cluster manager process (clstrmgrES) is always running. It can be in one of two states, as displayed by executing the command:

```
lssrc -ls clstrmgrES
```

ST_INIT (start event has executed

ST_NOTCONFIGURED (start event has not executed)

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

During maintenance periods it is often necessary to stop and start cluster services. But before doing so, you need to understand the node(s) interactions it causes and the impact on your system's availability. The cluster must be in sync synchronized and verification should detect no errors. The following section briefly describes the processes themselves and then the processing involved in startup or shutdown of these services. Later in this section we describe the procedures necessary start or stop cluster services on a node.

7.2.1 Cluster Services

The main HACMP and RSCT daemons are as follows:

- **Cluster Manager daemon (clstrmgrES)**: This is the main HACMP daemon. Itmaintains a global view of the cluster topology and resources and runs event scripts in response to changes in the state of nodes, interfaces, or resources (or when the user makes a request).

  The Cluster Manager receives information about the state of interfaces from Topology Services. The Cluster Manager maintains updated information about the location, and status of all resource groups.The Cluster Manager is a client of Group Services, and uses the latter for reliable inter-daemon communication.

  In Versions 5.1 and 5.2 this daemon is started after starting cluster services. In Version 5.3 the clstrmgr daemon is started via the init process and should be running at all times. Also in 5.3, since the clstrmgr daemon is now a long running process, you cannot use **`lssrc -s clstrmgrES`** to determine the state of the cluster. Use**`/usr/es/sbin/cluster/utilities/clcheck_server grpsvcs`** instead.

- **Cluster Lock Manager daemon (cllockd)**: This daemon provides advisory locking services. The cllockd daemon is required on cluster nodes only if those nodes are part of a concurrent access configuration. Note that the Lock Manager is not supported on the 64-bit kernel and also was removed completely starting with HACMP V5.2.

- **Cluster Communication Daemon (clcomdES)**: This daemon, first introduced in Version 5.1, provides secure communication between cluster nodes for all cluster utilities such as verification and synchronization and system management (C-SPOC). The **`clcomd`** daemon is started automatically

at boot time by the **init** process. Starting with Version 5.2, clcomdES must be running before any cluster services can be started.

- **Cluster Information Program (clinfoES)**: This daemon provides status information about the cluster to cluster nodes and clients and calls the **/usr/es/sbin/cluster/etc/clinfo.rc scrip**t in response to a cluster event. The clinfo daemon is optional on cluster nodes and clients.)
- **Cluster SMUX Peer daemon (clsmuxpd)**: This daemon maintains status information about cluster objects. It publishes state information for cluster topology and resources to the MIB. It is a client of the Cluster Manager.

  This daemon works in conjunction with the Simple Network Management Protocol (snmpd) daemon. All cluster nodes must run the clsmuxpd daemon.

  Note: The **clsmuxpd** daemon cannot be started unless the snmpd daemon is running.

  In HACMP Version 5.3, **clsmuxpd** *no* longer exists.

- **Cluster Topology Services Subsystem**: The RSCT Topology Services subsystem monitors the status of network interfaces and publishes the state to clients, who access the information through Group Services membership.The main daemon is the **hatsd**. Topology Services also includes network interface modules **hats_nim*** which send and receive heartbeats.All cluster nodes must run the Topology Services subsystem.
- **Cluster Event Management Subsystem**: This RSCT subsystem matches information about the state of system resources with information about resource conditions of interest to client programs (applications, subsystems, and other programs).The haemd daemon runs on each node of a domain.

  Note: Event Management is only used by Oracle 9i; it is replaced by the Resource Monitoring and Control subsystem in HACMP 5.2 and up.

- **Event Management AIX Operating System Resource Monitor**: This RSCT daemon acts as a resource monitor for the event management subsystem and provides information about the operating system characteristics and utilization. The **emaixos** daemon is started automatically by Event Management.
- **Cluster Group Services Subsystem**: This RSCT subsystem provides reliable communication and protocols required for cluster operation. Clients are distributed daemons, such as the HACMP Cluster Manager and the Enhanced Concurrent Logical Volume Manager. All cluster nodes must run the hagsd daemon.

- **Cluster Globalized Server daemon (grpglsmd)**: This RSCT daemon operates as a Group Services client; its function is to make switch adapter membership global across all cluster nodes. All cluster nodes must run the **grpglsmd** daemon.
- **Resource Monitoring and Control Subsystem**: This RSCT subsystem acts as a resource monitor for the event management subsystem and provides information about the operating system characteristics and utilization. The RMC subsystem must be running on each node in the cluster. By default the **rmcd** daemon is setup to start from **inittab** when it is installed. The **rc.cluster** script ensures the RMC subsystem is running.

7.2.2 Starting cluster services

In this section we describe the startup options of cluster services on any single node, multiple nodes, or even all nodes. You should always start cluster services by using SMIT. The SMIT screen can be seen in Figure 7-1.

```
                          Start Cluster Services

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Start now, on system restart or both                 now +
  Start Cluster Services on these nodes               [Maddi,Melany] +
  BROADCAST message at startup?                        true +
  Startup Cluster Information Daemon?                  false +
  Reacquire resources after forced down ?              false +
  Ignore verification errors?                          false +
  Automatically correct errors found during            Interactively +
  cluster start?


F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 7-1 Start Cluster Services Menu

Executing as the root user, perform the following steps to start the cluster services on a node.

1. Enter the fastpath **smit cl_admin**

2. In SMIT, choose **Manage HACMP Services->Start Cluster Services** and press *Enter*.
3. Enter field values as follows:

- **Start now, on system restart or both**: Indicate whether you want to start cluster services and the **clinfoES** when you commit the values on this panel by pressing Enter (**now**), when the operating system reboots (**on system restart**), or on **both** occasions.

> **Note:** In a production environment it is generally *not* considered a best practice to have HACMP services startup automatically on system restart.
>
> The reason for this is directly related to the aftermath of system failure. If a resource group owning system crashes, and AIX is set to reboot after crash, it could restart cluster services in the middle of a current takeover. Depending on the cluster configuration this could cause resource group contention, resource group processing errors, or even a fallback to occur. All of which could extend an outage.
>
> However during test and maintenance periods, and even on dedicated standby nodes, it may be convenient to use this option.

- **Start Cluster Services on these nodes**: Enter the name(s) of one or more nodes on which you want to start cluster services. Alternatively, you can select nodes from a picklist. When entering multiple nodes manually separate the names with a comma as shown in Figure 7-1 on page 332.
- **BROADCAST message at startup?**: Indicate whether you want to send a broadcast message to all nodes when the cluster services start.
- **Startup Cluster Lock Services**: Choose **true** or **false** to start the cllockd/cllockdES daeamon. This is used only in a concurrent access environment.

> **Note:** HACMP 5.2 and up no longer supports cllockd or cllockdES (the Cluster Lock Manager).

- **Startup Cluster Information Daemon?**: Indicate whether you want to start the **clinfo** daemon. If your application uses Clinfo, if you use the **clstat** monitor, or you want to run event emulation, set this field to **true**. Otherwise, set it to **false**.
- **Reacquire Resources after Forced Down?**: The default is **false** because HACMP expects the resources to be online since it did not bring them offline. If you previously stopped the services using the **forced** option *and* the

resources have been brought offline manually since services have were stopped, if you want the resource to be brought back online automatically you will choose **true.**

The following two options were in added in HACMP V5.3

- **Ignore Verification Errors?**: Set this value to **true** for all selected nodes to start cluster services if verification finds no errors on the specified nodes or on the cluster in general. Set this value to **false** to stop all selected nodes from starting cluster services if verification finds errors on any node.
- **Automatically correct errors found during cluster start?**: The options are **Yes, No,** and **Interactively.**

  Yes will fix automatically, without prompting. No will not fix them and prevent cluster services from starting if errors are encountered. The Interactively option will prompt the user during startup of what errors are found and reply to fix, or not to fix, accordingly.

  Press *Enter*. The system starts the cluster services on the nodes specified, activating the cluster configuration that you have defined. The time that it takes the commands and scripts to run depends on your configuration (that is, the number of disks, the number of interfaces to configure, the number of filesystems to mount, and the number of applications being started).

  During the **node_up** event, resource groups are acquired. The time it takes to run each **node_up** event is dependent on the resource processing during the event. The **node_up** events for the joining nodes are processed sequentially.

  When the command completes executing and HACMP cluster services are started on all nodes specified. SMIT displays a command status window. Note that when the SMIT panel indicates the completion of the cluster startup, event processing in most cases has not yet completed. To verify the nodes are up you can use *clstat, WebSmit,* or even tail the /tmp/hacmp.out file on any node. More information about this can be found in “Cluster status checking utilities” on page 411.

7.2.3 Stopping cluster services

The steps below describe the procedure for stopping cluster services on a single, multiple or all nodes in a cluster by using the C-SPOC utility on one of the cluster nodes. C-SPOC stops the nodes sequentially, not in parallel. If any node specified to be stopped is inactive, the shutdown operation aborts on that node. Just like starting services, SMIT should always be used to stop cluster services. The SMIT screen to stop cluster services is shown in

```
                              Stop Cluster Services

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Stop now, on system restart or both                 now +
  Stop Cluster Services on these nodes                [Maddi,Melany] +
  BROADCAST cluster shutdown?                          true +
* Shutdown mode                                       graceful +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 7-2 Stop Cluster Services Menu

To stop cluster services:

1. Enter the fastpath **smit cl_admin**
2. In SMIT, select **Manage HACMP Services->Stop Cluster Services** and press *Enter*.
3. Enter field values in the SMIT panel as follows:

- **Stop now, on system restart or both**

  Indicate whether you want the cluster services to stop **now**, at **restart** (when the operating system reboots), or on both occasions. If you select **restart** or **both**, the entry in the **/etc/inittab file** that starts cluster services is removed. Cluster services will no longer come up automatically after a reboot.

- **BROADCAST cluster shutdown?**

  Indicate whether you want to send a broadcast message to users before the cluster services stop. If you specify true, a message is broadcast on all cluster nodes.

- **Shutdown mode**

  Indicate the type of shutdown:

  graceful: Shutdown after the **/usr/es/sbin/cluster/events/node_down_complete** script is run on the node to release its resources. Other cluster nodes do not take over the resources of the stopped node.

graceful with takeover: Shutdown after the **/usr/es/sbin/cluster/events/node_down_complete** script runs to release its resources. Other nodes take over the resources of the stopped node.

forced: Shut down immediately. The node retains control of all its resources.You can use this option to bring down a node while you perform maintenance or make a change to the cluster configuration, such as adding a network card.

However, since cluster services is stopped, the applications are no longer highly available. If a failure occurs, recovery for them will not be provided.

Observations

Note: When utilizing enhanced concurrent volume groups in either concurrent access mode, for disk heartbeat or fast disk takeover mode, the forced option is not available.

1. In previous HACMP versions when the Cluster Manager stopped, the applications were no longer highly available. If a failure occurred, recovery for the applications was not provided. In HACMP 5.3, the Cluster Manager persists even when cluster services are stopped. The rc.cluster and clstop scripts issue IPC™ commands to the Cluster Manager. The Cluster Manager runs an event script to start the RSCT stack (rc.cluster).
2. In addition, in HACMP 5.3, node_down_complete calls the routine to shutdown the RSCT stack. The Cluster Manager responds to stopsrc commands by terminating immediately—without running a node_down event—as required before installing maintenance or otherwise replacing the clstrmgr daemon or dependent libraries (for example libclstr.a).

Important: *Never* use the `kill -9` command to stop the Cluster Manager or any RSCT daemons. This causes an abnormal exit. SRC will run the clexit.rc script and halt the system immediately. This causes the other nodes to initiate a fallover.

7.3 Resource group and application management

In this section we discuss how to:

- Bring a resource group offline
- Bring a resource group online
- Move a resource group
- Suspend/Resume application monitoring

Understanding each of these is important, along with stopping and starting cluster services, as these are often used during maintenance periods.

In the following sections we start off with assuming that cluster services are running, the resource group(s) is online, the application(s) is running and the cluster is stable. If the cluster is not in the stable state, then the resource group related operations will not be possible.

All three resource group options we discuss can be done by using the **clRGmove** command. However, in our examples we use C-SPOC. They also all have similar SMIT screens and picklist. In an effort to streamline this documentation we show only one SMIT screen in each of the following sections.

When performing any of the resource group operations it is important to understand the *priority override location* setting. It is so important we have dedicated an entire section “Priority override location” on page 342.

7.3.1 Bring a resource group offline via SMIT

To bring a resource group offline:

1. Enter **smit cl_admin**
2. In SMIT, select **HACMP Resource Group and Application Management->Bring a Resource Group Offline.** The picklist appears. as shown in Figure 7-3 on page 338. It lists only the resource groups that are online or in the ERROR state on all nodes in the cluster.

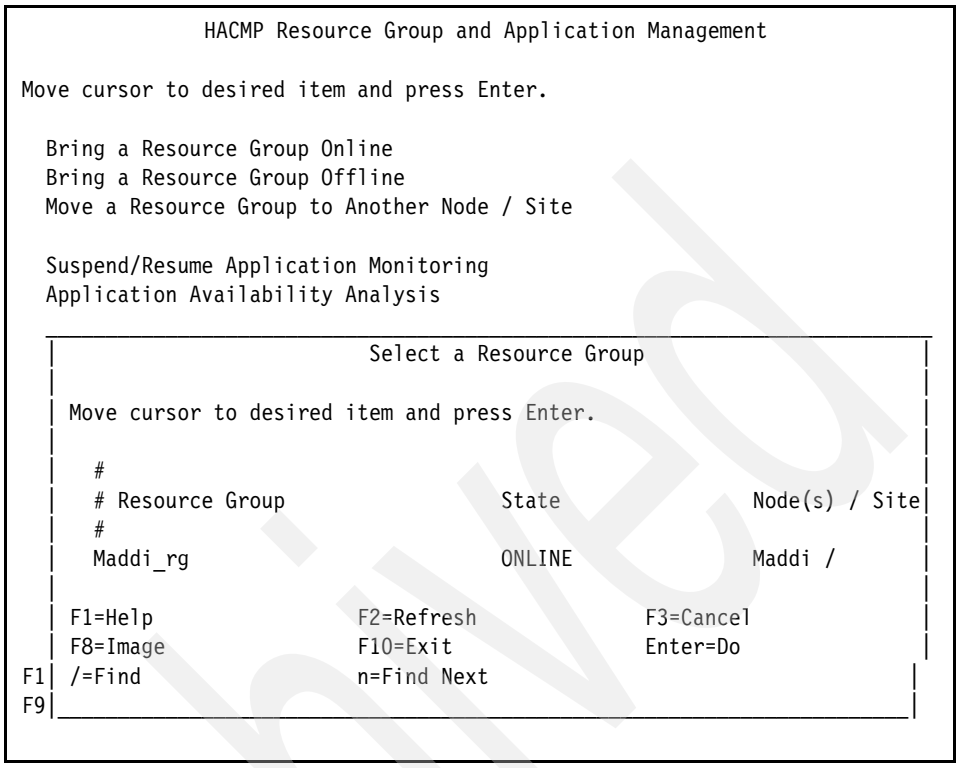

Figure 7-3 Resource Group picklist

3. Select the appropriate resource group from the list and press *Enter*.

   Once the resource group has been selected another picklist appears to **Select a Destination Node**. The picklist will only contain the node(s) that are currently active in the cluster that currently are hosting the previously selected resource group.

4. Select a destination node from the picklist and press *Enter*.

5. The final SMIT menu appears with the information selected in the previous picklists as shown in. In addition, there is one additional field needed to specify is the **Persist across Cluster Reboot?** Generally speaking you leave this set to the default of false. This field is directly related to the **POL setting** and more information can be found in “Priority override location” on page 342.

6. Verify the entries previously specified and then press *Enter* to execute the processing of the resource group to be brought offline.

Once processing is completed not only will the resource group be offline but also cluster services remain active on the node.

7.3.2 Bring a resource group online via SMIT

To bring a resource group online:

1. Enter **`smit cl_admin`**
2. In SMIT, select **`HACMP Resource Group and Application Management->Bring a Resource Group Online.`** The picklist appears. It lists only the resource groups that are online or in the ERROR state on all nodes in the cluster.
3. Select the appropriate resource group from the list and press *Enter*.

   Once the resource group has been selected another picklist appears to **Select a Destination Node**. The picklist will only contain those nodes that are currently active in the cluster and are participating nodes in the previously selected resource group.
4. Select a destination node from the picklist as shown in Figure 7-4 on page 340. When you select it, *it becomes a priority override location for this resource group.*
5. The final SMIT menu appears with the information selected in the previous picklists as shown in. In addition, there is one additional field needed to specify is the **Persist across Cluster Reboot?** Generally speaking you leave this set to the default of false. This field is directly related to the **POL** setting and more information can be found in “Priority override location” on page 342
6. Verify the entries previously specified and then press *Enter* to execute the moving of the resource group.

Upon successful completion, HACMP displays a message and the status, location, and a type of location (persistent or not) of the resource group that was successfully stopped on the specified node.

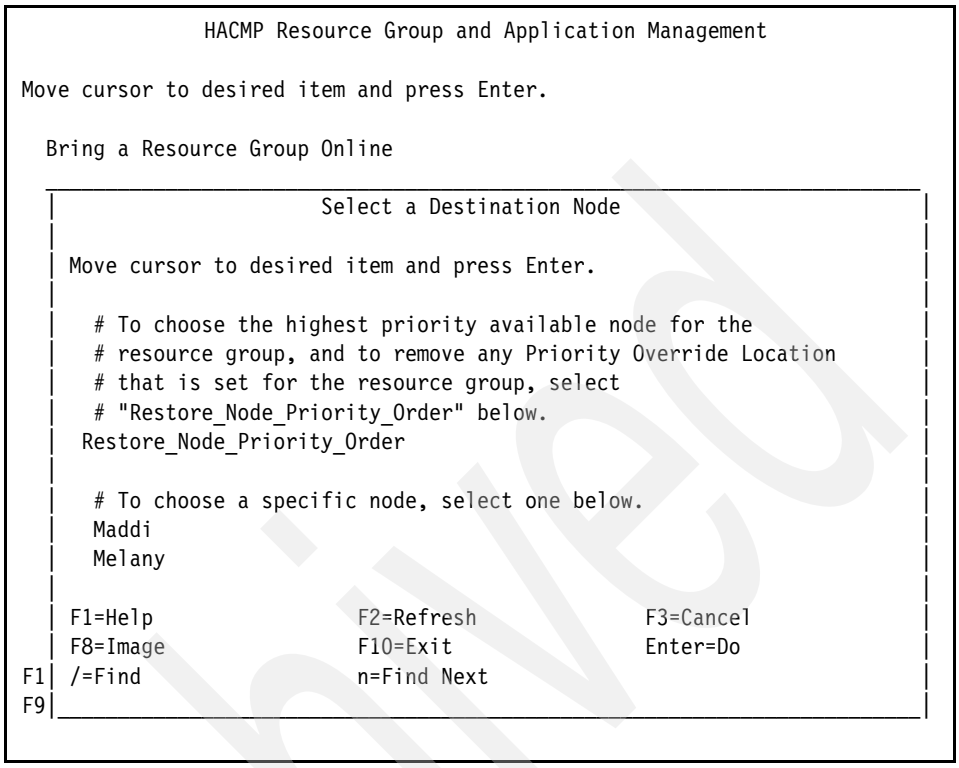

Figure 7-4 Destination node picklist

7.3.3 Move a resource group via SMIT

Moving a resource group consists of a graceful stopping of the resources on the current owning node and then processing the normal resource group startup procedures on the destination node. This results in a short period in which the application is not available.

HACMP V5.3 added the ability to move a resource group to another site. The concept is the same as moving it between local nodes. For our example we will be using the option to move to another node as opposed to another site.

To move a resource group:

1. Enter **`smit cl_admin`**
2. In SMIT, select **`HACMP Resource Group and Application Management-> Move a Resource Group to Another Node/Site->Move Resource Groups to Another Node.`** The picklist appears. It lists only the resource groups that are online or in the ERROR state on all nodes in the cluster.

3. Select the appropriate resource group from the list and press *Enter*.

   Once the resource group has been selected another picklist appears to **Select a Destination Node**. The picklist will only contain those nodes that are currently active in the cluster and are participating nodes in the previously selected resource group.

4. Select a destination node from the picklist. When you select it, *it becomes a priority override location for this resource group*.

5. The final SMIT menu appears with the information selected in the previous picklists. There is one additional field needed to specify, which is the **Persist across Cluster Reboot?** Generally speaking you leave this set to the default of false. **POL** is set regardless, the only difference is do you want to stay set even after stopping/starting of the cluster? More information can be found in “Priority override location” on page 342.

```
                              Move a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  Resource Group to be Moved                         Maddi_rg
  Destination Node                                   Melany
  Persist Across Cluster Reboot?                     false+


F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 7-5 Move a Resource Group Smit Screen

6. Verify the entries previously specified and then press *Enter* to execute the moving of the resource group.

If the event completes successfully, HACMP displays a message and the status, location, and a type of location (persistent or not) of the resource group that was successfully stopped on the specified node as shown in Figure 7-6 on page 342.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

[MORE...7]


Cluster Name: Testcluster

Resource Group Name: Maddi_rg
Priority Override Information:
    Primary Instance POL:   Melany
Node                         State
---------------------------- ---------------
Maddi                        OFFLINE
Melany                       ONLINE

[BOTTOM]

F1=Help            F2=Refresh          F3=Cancel           F6=Command
F8=Image           F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 7-6 Resource Group Status with POL setting displayed

Attention: Notice in the example above that the POL setting is set.

Anytime a resource group is moved to another node, application monitoring for the application(s) are suspended during the application stop. Once the application has restarted on the destination node application monitoring will resume. Additional information can be found in “Suspend/Resume application monitoring” on page 344.

7.3.4 Priority override location

In releases prior to HACMP 5.1, you could specify the “sticky” migration attribute for a migrated resource group. In HACMP 5.1, the “sticky” attribute is removed. Instead, you can specify the priority override location attribute. The functionality of the “sticky” attribute in HACMP 5.1 is equivalent to that of the persistent priority override location attribute. A priority override location is the destination node on which the resource group migration takes place.

A priority override location setting actually “overrides” the fallover/fallback configuration settings of the resource group. It can be specified to be as *persistent* or *not-persistent.* This setting is directly related to the SMIT screen field of **Persist Across Cluster Reboot**. The two options are *true* and *false*. The default is false for the POL setting to *not* persist across a cluster reboot.

The following information explains how POL works for non-concurrent resource groups when using the different flags of **clRGmove.**

Note: Only one resource group may be moved at a time with **clRGmove**.

- For every non-concurrent resource group movement that uses the **-n** flag, to explicitly specify a destination instead of the **-r** (restoring node priority) flag, the destination node becomes the Priority Override Location. The Priority Override Location lasts until you explicitly use the **-**r parameter for the destination node instead of **-n** when manually moving the resource group again.
- When moving a resource group offline, the resource group stays offline until you manually bring it back online. If you manually bring it back online with the **-n** flag to specify a node, that node becomes the Priority Override Location.
- When bringing a resource group back online with the **-r** flag, the active highest priority node is used and the Priority Override Location is removed from the resource group.

For concurrent resource groups:

- When taking a concurrent resource group offline on all nodes the Priority Override Location becomes OFFLINE for all nodes in the resource group. When bringing a concurrent resource group offline on just one node, the OFFLINE state for the resource group on node will be added to the Priority Override Location list.
- When bringing a concurrent resource group online on all nodes, the Priority Override Location is removed for all nodes in the resource group.
- When bringing a concurrent resource group online on just one node, the OFFLINE state for the resource group on that node is removed from the Priority Override Location list.

The POL setting of the resource groups can be viewed by executing **clRGinfo -p**. The output from this command is the same as seen from the results of executing a resource group operation via SMIT as shown in Figure 7-6 on page 342.

7.3.5 Suspend/Resume application monitoring

During application maintenance periods it is often desirable to bring the application offline only, as opposed to stopping cluster services. If application monitoring is being used it is required to suspend application monitoring before stopping the application. Otherwise HACMP will take the predefined recovering procedures when it detects the application is down, which is not desired during maintenance. Defining application monitors is explained in “Application monitoring” on page 421.

To suspend application monitoring:

1. Enter **smit cl_admin**
2. In SMIT, select **HACMP System Management->Suspend/Resume Application Monitoring->Suspend Application Monitoring** and press *Enter*.

   You are prompted to select the application server for which this monitor is configured. If you have multiple application monitors, they are all suspended until you choose to resume them or until a cluster event occurs to resume them automatically, as explained above.

The monitoring will stay suspended until either resumed manually or until the resource group is stop/restarted.

To resume application monitoring:

1. Enter **smit cl_admin**
2. In SMIT, select **HACMP System Management->Suspend/Resume Application Monitoring->Resume Application Monitoring** and press *Enter*.

   Choose the appropriate application server associated with the application monitor you want to resume.

Application monitoring will continue to stay active until either suspended manually or until the resource group is brought offline.

7.4 Scenarios

In this section we cover the common scenarios of:

- PCI hot-plug replacement of a NIC
- Loading AIX and HACMP fixes
- Replacing and LVM mirrored disk
- Application maintenance

7.4.1 PCI hot-plug replacement of a NIC

This section takes you through the process of replacing a PCI hot plug network interface card by utilizing the C-SPOC “PCI Hot Plug Replace a Network Interface Card” facility.

Special Considerations

Keep the following in mind before you replace a hot-pluggable PCI network interface card.

- Be aware of the following consideration: If a network interface you are hot-replacing is the only available keepalive path on the node where it resides, you *must* shut down HACMP on this node in order to prevent a partitioned cluster while the interface is being replaced.

  This is easily avoidable by having a working non-IP network between the cluster nodes.
- SMIT gives you the option of doing a graceful shutdown on this node. From this point, you can manually hot-replace the network interface card.
- Hot-replacement of Ethernet, Token-Ring, FDDI and ATM network interface cards is supported. This process is not supported for non-IP communication devices.
- You should manually record the IP address settings of the network interface being replaced to prepare for unplanned failures.
- You should not attempt to change any configuration settings while the hot replacement is in progress.

The SMIT interface simplifies the process of replacing a hot-pluggable PCI network interface card. HACMP supports only one PCI hot plug network interface card replacement via SMIT at one time per node.

Note: If the network interface was alive before the replacement process began, then between the initiation and completion of the hot-replacement, the interface being replaced is in a maintenance mode. During this time, network connectivity monitoring is suspended on the interface for the duration of the replacement process.

Scenario 1 (Live NICs Only)

Follow the procedure below when hot-replacing the following:

- A live PCI network service interface in a resource group and with an available non-service interface
- A live PCI network service interface not in a resource group and with an available non-service interface

– A live PCI network boot interface with an available non-service interface

1. Go to the node on which you want to replace a hot-pluggable PCI network interface card.
2. Type **smit hacmp**
3. In SMIT, select **System Management (C-SPOC)->HACMP Communication Interface Management->PCI Hot Plug Replace a Network Interface Card** and press *Enter*.

   You can also get to this panel with the fastpath **smitty cl_pcihp**

   SMIT displays a list of available PCI network interfaces that are hot-pluggable.
4. Select the network interface you want to hot-replace. Press *Enter*. The service address of the PCI interface is moved to the available non-service interface.
5. SMIT prompts you to physically replace the network interface card. After you have replaced the card, you are asked to confirm that replacement has occurred.

   If you select **yes**, the service address will be moved back to the network interface which has been hot-replaced. On aliased networks, the service address will not move back to the original network interface, but will remain as an alias on the same network interface. The hot-replacement is complete.

   If you select **no**, you must manually reconfigure the interface settings to their original values:

   a. Run the **drslot** command to take the PCI slot out of the removed state.

   b. Run **mkdev** on the physical interface.

   c. Use **ifconfig** manually as opposed to **smit chinet, cfgmgr,** or **mkdev** in order to avoid configuring duplicate IP addresses or an unwanted boot address.

Scenario 2 (Live NICs Only)

Follow the procedure below when hot-replacing a live PCI network service interface on a resource group but with no available non-service interface. Steps 1-3 are the same as in the previous scenario, so in this scenario we start from the smit fastpath of **smitty cl_pcihp.**

1. Select the network interface you want to hot-replace and press *Enter*.

   SMIT prompts you to choose whether to move the resource group to another node during the replacement process in order to ensure its availability.

2. If you choose to do this, SMIT gives you the option of moving the resource group back to the node on which the hot-replacement took place after completing the replacement process.

   If you do not move the resource group to another node, it will be offline for the duration of the replacement process.

3. SMIT prompts you to physically replace the network interface card. After you have replaced the card, you are asked to confirm that replacement has occurred.

   If you select **Yes**, the hot-replacement is complete.

   If you select **no**, you must manually reconfigure the interface settings to their original values:

   a. Run the **drslot** command to take the PCI slot out of the removed state.

   b. Run **mkdev** on the physical interface.

   c. Use **ifconfig** manually as opposed to **smit chinet, cfgmgr,** or **mkdev** in order to avoid configuring duplicate IP addresses or an unwanted boot address.

   d. (If applicable) Move the resource group back to the node from which you moved it in Step 2.

Scenario 3 (Non-alive NICs Only)

Follow the procedure below when hot-replacing the following:

- A non-alive PCI network service interface in a resource group and with an available non-service interface
- A non-alive PCI network service interface not in a resource group and with an available non-service interface
- A non-alive PCI network boot interface with an available non-service interface.

We begin again from the fastpath of **smitty cl_pcihp** as in the previous scenario.

1. Select the network interface you want to hot-replace and press *Enter*.

   SMIT prompts you to physically replace the network interface card.

2. After you have replaced it, SMIT prompts you to confirm that replacement has occurred.

   If you select **Yes**, the hot-replacement is complete.

   If you select **no**, you must manually reconfigure the interface settings to their original values:

   a. Run the **drslot** command to take the PCI slot out of the removed state.

b. Run **mkdev** on the physical interface.

c. Use **ifconfig** manually as opposed to **smit chinet, cfgmgr,** or **mkdev** in order to avoid configuring duplicate IP addresses or an unwanted boot address.

Hot-Replacing an ATM Network Interface Card

ATM network interface cards support multiple logical interfaces on one network interface card. An ATM network interface hot- replacement is managed the same as other network interface cards, with the following exceptions:

- All logical interfaces configured on the card being replaced that are not configured for and managed by HACMP are lost during the replacement process. They will not be reconfigured on the newly replaced ATM entered interface card. All other logical interfaces configured for and managed by HACMP on the ATM network interface card being replaced are restored when the replacement is complete.
- Since it is possible to have more than one service interface configured on an ATM network interface card—thus multiple resource groups on one ATM network interface—when you hot-replace an ATM network interface card, SMIT leads you through the process of moving each resource group on the ATM interface, one at a time.

For details, refer also to Chapter 12 in *High Availability Cluster Multi-Processing Administration Guide*, SC23-4862-06.

7.4.2 Fixes

This section relates to installing fixes (APARs/PTFS) to either and/or both AIX and HACMP. It is our recommendation that fixes/maintenance should be loaded once a quarter. However, based on customer feedback, it is more common to do so only twice a year around extended holiday weekends. Some cases dictate deviating from the standard practice as any serious problems are encountered.

Some AIX fixes can be loaded dynamically without a reboot. Kernel and device driver updates often require a reboot as installing updates to them runs a **bosboot**. One way to determine if a reboot is required is to check the .toc created via the **inutoc** command prior to installing the fixes. The contents of the file contains fileset information similar to Example 7-1:

Example 7-1 Checking the .toc prior to installing fixes

```
bos.64bit                   5.3.0.0                   I  b usr,root
#   Base Operating System 64 bit Runtime
bos.INed                    5.3.0.0                   I  N usr,root
```

In the above example the fileset bos.64bit requires a reboot as indicated by the "*b*" character in fourth column. The "*N*" character indicates that a reboot is not required.

Applying HACMP fixes is similar to AIX fixes. The filesets to be updated indicate if a cluster restart is necessary via the same method as listed above. Always check with support line if unsure about the effects of loading certain fixes.

When updating AIX or HACMP software it's our recommendation to:

- Take a cluster snapshot and save it somewhere off the cluster.
- Back up the operating system and data before performing any upgrade. Prepare a backout plan in case you encounter problems with the upgrade.
- *Always* do an initial run through on a test cluster.
- Use disk update if possible.
- Follow this same general rule for fixes to the application; follow specific instructions for the application.

The general procedure for applying either AIX or HACMP fixes is as follows:

- *Apply,* do not commit, APARs to standby node.
- Fallover (graceful shutdown with takeover) to standby machine.
- *Apply* APARs to primary node.

Before applying fixes to the standby node you will want to stop cluster services. After applying fixes, reboot the node if necessary. Restart cluster services to have the node rejoin the cluster as a standby.

In order to apply fixes to production nodes, stop cluster services gracefully with takeover. Once takeover has completed, cluster services should continue to stop. Upon cluster services stopping completely, apply fixes, reboot the node as needed, and restart cluster services. Depending on the resource group fallback policy, when the node rejoins the cluster it may acquire the resources. If not, you can use C-SPOC to move the resource group back to the original node.

7.4.3 Storage

Most shared storage environments today use some level of RAID for data protection and redundancy. When utilizing RAID (1,5, or 10) devices, individual disk failures normally do not require AIX LVM maintenance to be performed. Any procedures required are often external to cluster nodes and have no affects to

the cluster itself. However, if protection is provided by utilizing LVM mirroring then LVM maintenance procedures are required.

C-SPOC provides a facility to aid in the replacement of failed LVM mirrored disk. This facility, *Cluster Disk Replacement,* performs all the necessary LVM operations of replacing an LVM mirrored disk. To utilize this facility, ensure the following:

- You have root privilege.
- The affected disk, and preferably the entire volume group, is mirrored.
- The desired replacement disk is available to the each node and a PVID is already assigned to it and is shown on each node via **`lspv`**.

If physically replacing an existing disk, remove the old disk and replace the new one in its place. This of course assumes the drive is hot plug replaceable which is common.

To replace a mirrored disk via C-SPOC

1. Locate the failed disk. Make note of the PVID volume group.
2. Enter **`smitty cl_admin`**
3. In SMIT, select **HACMP Physical Volume Management->Cluster Disk Replacement** and press *Enter*.

   SMIT displays a list of disks that are members of volume groups contained in cluster resource groups. There must be two or more disks in the volume group where the failed disk is located. The list includes the volume group, the hdisk, the disk PVID, and the reference cluster node. (This node is usually the cluster node that has the volume group varied on.)
4. Select the disk for disk replacement (*source disk*) and press *Enter*.

   SMIT displays a list of those available shared disk candidates that have a PVID assigned to them, to use for replacement. (Only a disk that is of the same capacity or larger than the failed disk is suitable to replace the failed disk.)
5. Select the desired replacement disk (*destination disk*) and press *Enter*.

   SMIT displays your selections from the two previous panels.
6. Press *Enter* to continue or *Cancel* to terminate the disk replacement process.

   A warning message will appear telling you that continuing will delete any information you may have stored on the destination disk.
7. Press *Enter* to continue or *Cancel* to terminate.

   SMIT displays a command status panel, and informs you of the **`replacepv`** recovery directory. If disk configuration fails and you want to proceed with

disk replacement, you must manually configure the destination disk. If you terminate the procedure at this point, be aware that the destination disk may be configured on more than one node in the cluster.

The **replacepv** utility updates the volume group in use in the disk replacement process (on the reference node only).

Note: During the command execution, SMIT tells you the name of the recovery directory to use should replacepv fail. Make note of this information as it is required in the recovery process.

Configuration of the destination disk on all nodes in the resource group takes place.

If a node in the resource group fails to import the updated volume group, you can use the C-SPOC *Import a Shared Volume Group* facility as shown in Figure 8-20 on page 386.

C-SPOC will not remove the failed disk device information from the cluster nodes. This must done manually by running **rmdev -dl <devicename>.**

Other LVM related C-SPOC operations can be found in 8.4, “Shared storage management” on page 382.

7.4.4 Applications

Of course each application varies, however most application maintenance requires the application be brought offline. This can be done in a number of ways. The most appropriate method for any particular environment depends on the overall cluster configuration.

In a multi-tier environment where an application server is dependent on a database, and maintenance is to be performed on the database, then usually both the database and the application server will need to be stopped. It is most common that at least the database will be in the cluster. When using resource group dependencies, the application server may easily be part of the same cluster.

It is also common, to minimize the overall downtime of the application, that the application maintenance be performed first on the non-production nodes for that application. Traditionally this means on a standby node, however it is not very common that a backup/fallover node truly is a standby only. If not a true standby node then any work load or applications currently running on that node must be accounted for to minimize any adverse affects of installing the maintenance. Hopefully this has all been tested previously in a test cluster.

In most cases stopping cluster services is not needed. You can simply bring the resource group offline as described in “Bring a resource group offline via SMIT” on page 337. If the shared volume group is needed to be online during the maintenance then you can just suspend application monitoring and execute the application stop server script to bring the application offline. However, this will leave the service IP address online which may not be desirable.

In a multiple resource group and/or multiple application environment all running on the same node it may not be feasible to stop cluster services on the local node. Be aware of the possible affects of not stopping cluster services on the node in which application maintenance is being performed.

If during the maintenance period, the system encounters a catastrophic error resulting a crash, a fallover will occur. This may be undesirable if the maintenance has not been performed on the fallover candidates first and/or if the maintenance is incomplete on the local node. Though this may be a rare occurrence, the possibility exists and must be understood.

Another possibility is that if another production node fails during this maintenance period can a fallover occur successfully on the local node without adverse affects. If this is not desired, and there are multiple resource groups, then you may want to move the other resource groups to another node first and then stop cluster services on the local node.

If using persistent addresses, and you stop cluster services, local adapter swap protection is no longer provided. Though again rare, this makes it possible that when using the persistent address to perform maintenance and the hosting NIC fails that your connection will be dropped.

After performing the application maintenance you should *always* test the cluster again. Depending on what actions you chose to stop the application, you will either need to restart cluster services, bring the resource group back online via C-SPOC, or manually run the application start server script and resume application monitoring as needed.

8

Managing your cluster

In this chapter we describe HACMP cluster management and administration tips and tricks, including:

- C-SPOC in general
- File collections
- User administration
- Shared storage management
- Time synchronization
- Cluster verification and synchronization
- Cluster monitoring

8.1 CSPOC DP

C-SPOC (Cluster Single Point of Control) is a very useful tool, helping to maintain the whole cluster from one single point. It provides facilities for performing common cluster-wide administration tasks from any active node within the cluster. The downtime, that could be caused by cluster administration, is reduced by using C-SPOC.

High Availability Clusters require special attention regarding to a system administration. We strongly recommend that a change management discipline is strictly followed.

Before starting with the cluster management details description we want to emphasize the general best-practice about cluster administration:

- Wherever possible, use HACMP's C-SPOC facility to make changes to the cluster.
- Document routine operational procedures (for example, shutdown, startup, increasing size of a filesystem).
- Restrict access to the root password to trained HACMP administrators.
- Always take a snapshot of your existing configuration before making a change.
- Monitor cluster regularly.

8.1.1 C-SPOC in general

The C-SPOC functionality is provided through its own set of cluster administration commands, accessible trough SMIT menus. The location of the commands is /usr/es/sbin/cluster/cspoc. It uses cluster communication daemon clcomdES to execute commands on remote nodes. If this daemon is not running, the command could not be executed and C-SPOC operation fails.

Note: From the HACMP 5.3 Cluster Manager process *clstrmgrES* is initiated from init process, so is always running despite the cluster is started or not. The active node means the node where aside *clstrmgrES* also other necessary cluster services are running.

C-SPOC operations fails if any target node is down at the time of execution or selected resource is not available. It requires correctly configured cluster in the sense that all nodes within the cluster could communicate.

If node failure does occur during a C-SPOC operation, an error is displayed to the panel and the error messages as well as other error information are recorded

in the C-SPOC log (/tmp/cspoc.log - the default location). You should check this log when any C-SPOC problems occur. You could find more information about HACMP logs in the 8.7.5, “Log files” on page 418.

8.1.2 C-SPOC SMIT menu

C-SPOC SMIT menus are accessible by running **`smit hacmp`** **> System Management (C-SPOC)** or using a SMIT fast path **`smit cl_admin`**. The main C-SPOC functions or sub-menus are presented as appears in SMIT C-SPOC menu in order, as they appear within main C-SPOC menu:

- **Manage HACMP Services.** This part contains menus for start, stop cluster on one ore selected nodes as well as menu to show running cluster services on local node. You can find more details in the 7.2, “Starting and stopping the cluster” on page 329 and in the 8.7.2, “Cluster status and services checking utilities” on page 413.
- **HACMP Communication Interface Management.**
- **HACMP Resource Group and Application Manipulation.** This part contains menus and utilities for cluster resource group manipulation as well as application monitoring and application availability measurement tools. You can find more details about resource group manipulation in the Chapter , “” on page 336, about application monitoring in the 8.7.7, “Application monitoring” on page 421 and about application analyses tool in the 8.7.8, “Measuring an application availability” on page 424.
- **HACMP Log Viewing and Management.** This part contains the utilities for displaying the contents of log files and for settings some log file parameters like residing directory and level, debug level and format (standard html). You can find more details in the 8.7.5, “Log files” on page 418.
- **HACMP File Collection Management.** This part contains the utilities providing cluster-wide file synchronization capabilities through CSPOC file collection functions. The file synchronization utility is based on file collection. A file collection is a user defined set of files. You can find more details in the 8.2, “File collections SV” on page 356.
- **HACMP Security and Users Management.** This part contains menus and utilities for various security settings as well as users, groups and password management within cluster. You can find more details about security in Chapter 9, “Cluster security” on page 429 and about user management in 8.3, “User administration SV” on page 365.
- **HACMP Logical Volume Management.** This part contains the utilities providing shared volume group, shared logical volumes and shared file system management cluster-wide. You can find more details about this topics in the 8.4.2, “C-SPOC Logical Volume Manager” on page 386.

- **HACMP Concurrent Logical Volume Management.** This part contains the utilities providing concurrent volume group, concurrent logical volumes and concurrent file system management cluster-wide. You can find more details about this topics in the 8.4.3, “C-SPOC Concurrent Logical Volume Management” on page 388.
- **HACMP Physical Volume Management.** This part contains the utilities for physical volume management cluster-wide like physical volume adding, removing and replacement. It has support for datapath devices and cross-LVM mirroring as well. You can find more details about this topics in the 8.4.4, “C-SPOC Physical Volume Management” on page 388.
- **Open a SMIT Session on a Node.** This feature offers possibility to open a basic SMIT window on any active node in the cluster. You can initiate any SMIT action to any node in the cluster just from local SMIT menu.

8.2 File collections SV

HACMP provides cluster-wide file synchronization capabilities through C-SPOC file collection functions. The file synchronization utility is based on file collection. A file collection is a user defined set of files. You can add or remove files to a file collection and you can specify when HACMP synchronizes this files.

HACMP provides three ways to propagate your files:

- Manually: You can synchronize your files manually at any time. The files are copied from the local node to the remote one.
- Automatically during cluster verification and synchronization: The files are propagated from the node where you start the HACMP verification.
- Automatically when changes are detected: HACMP checks periodically the file collection on all nodes and if its see a file has changed, then it synchronize this file across the cluster. You can set up a timer for how frequently HACMP checks the file collections.

HACMP retains the file’s permissions, ownership, and time stamp and propagates them to the remote nodes. You can only specify ordinary files for a file collection, you cannot add symbolic links, directory, pipe, socket, device file (/dev/*), files from /proc directory and ODM files from /etc/objrepos/* and /etc/es/objrepos/*. Always use full path names. Each file can be added only to one file collection except those files that automatically added to HACMP_Files collection. The files shouldn’t be exist on the remote nodes, HACMP will create them during the first synchronization. The zero length or non-existent files are not propagated from the local node.

HACMP creates a backup copy of the modified files during synchronization on all nodes. These backups are stored in /var/hacmp/filebackup directory. Only one previous version is retained and you can only manually restore them.

The file collection logs are stored in /var/hacmp/log/clutils.log file.

Important: It is your responsibility to ensure that files on the local node (where you start the propagation) are the most recent and are not corrupted.

8.2.1 Predefined file collections

HACMP provides two file collections by default: **Configuration_Files** and **HACMP_Files**. None of them is set up for automatic synchronization by default. You can enable them by setting either the "`Propagate files during cluster synchronization`" or "`Propagate files automatically when changes are detected`" option to **Yes** in SMIT `Change/Show a file collection` menu, see chapter "Change a file collection" on page 360.

Configuration_Files

This collection contains the essential AIX configuration files:

- /etc/hosts
- /etc/services
- /etc/snmpd.conf
- /etc/snmpdv3.conf
- /etc/rc.net
- /etc/inetd.conf
- /usr/es/sbin/cluster/netmon.cf
- /usr/es/sbin/cluster/etc/clhosts
- /usr/es/sbin/cluster/etc/rhosts

You can easily add or remove files to this collections. See "Add files to a file collection" on page 362 for more information.

HACMP_Files

This file collection automatically collects all the user-defined scripts from the HACMP configuration. If you define any of the following files in your cluster configuration, then that files are automatically included in the HACMP_Files file collection:

- Application server start script

- Application server stop script
- Event notify script
- Pre-event script
- Post-event script
- Event error recovery script
- Application monitor notify script
- Application monitor cleanup script
- Application monitor restart script
- Pager text message file
- SNA Link start and stop scripts
- X.25 Link start and stop scripts
- HA Tape support start script
- HA Tape support stop script
- User-defined event recovery program
- Custom snapshot method script

Let's see an example of how it is working. Our cluster has an application server, called DB2. Its start script is /usr/ha/db2.start, stop script is /usr/ha/db2.stop. Also we have a custom post-event script for node_up event, called /usr/ha/post.node_up. This three files are automatically added to HACMP_Files file collection when we defined them during HACMP configuration. You can check this:

1. Start SMIT HACMP file collection management: **`smit cm_filecollection_mgt`**.
2. Select **Change/Show a File Collection**.
3. Select **HACMP_Files** from the pop-up list and press Enter.
4. Go down to **Collection files** field and press F4. As you can see on SMIT screenshoot on Figure 8-1 on page 359., the application start and stop scripts and the post event command is automatically added to this file collection.

```
                        Change/Show a File Collection

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  File Collection Name                                  HACMP_Files
  New File Collection Name                              []
  File Collection Description                           [User-defined scripts >
  +--------------------------------------------------------------------------+
  |                            Collection files                              |
  |                                                                          |
  | The value for this entry field must be in the                            |
  | range shown below.                                                       |
  | Press Enter or Cancel to return to the entry field,                      |
  | and enter the desired value.                                             |
  |                                                                          |
  |   /usr/ha/db2.start                                                      |
  |   /usr/ha/db2.stop                                                       |
  |   /usr/ha/post.node_up                                                   |
  |                                                                          |
  | F1=Help                 F2=Refresh              F3=Cancel                |
F1| F8=Image                F10=Exit                Enter=Do                 |
F5| /=Find                  n=Find Next                                      |
F9+--------------------------------------------------------------------------+
```

Figure 8-1 HACMP_Files file collection example

Attention: You cannot add or remove files directly to this file collection. If you start using HACMP_Files collection be sure that your scripts can work properly on all nodes.

If you don't want to synchronize all off your user-defined scripts or they are not the same on all nodes, then disable this file collection and create an other one, which includes only the required files.

8.2.2 Manage file collections

Here we describe how you can create, modify and remove a file collection.

Add a file collection

1. Start SMIT: **smitty->Communications Applications and Services->HACMP for AIX Select System Management (CSPOC)->HACMP File Collection Management**.

Or you can start HACMP File Collection Management by entering **smit cm_filecollection_menu**

2. Select **Manage File Collections.**
3. Select **Add a File Collections.**
4. Supply the requested information (see Figure 8-2):
   - **File Collection Name:** unique name for file collection.
   - **File Collection Description:** a short description of this file collection.
   - **Propagate files during cluster synchronization?:** if you set this **yes**, then HACMP propagates this file collection during cluster synchronization. This is a convenient solution for cluster related files, e.g., your application start up scripts automatically synchronized after you make any changes in the cluster configuration.
   - **Propagate files automatically when changes are detected?:** If you select **yes,** HACMP will check regularly the files in this collection and if any of them changed, then it re-propagate them.

   If both of the above options are left on **No,** then no automatically synchronization will take place.

```
                          Add a File Collection

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* File Collection Name                               [app_files]
  File Collection Description                        [Application config fi>
  Propagate files during cluster synchronization?     yes +
  Propagate files automatically when changes are det  no +
  ected?



F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 8-2 Add a file collection

Change a file collection

1. Start HACMP File Collection Management by entering **smit cm_filecollection_menu** fast path.

2. Select **Manage File Collections.**
3. Select **Change/Show a File Collections.**
4. Select a file collection from the pop-up list.
5. Now you can change the following information (see SMIT screen on Figure 8-3):
   - File collection name
   - Description
   - Propagate files during cluster synchronization (yes/no)
   - Propagate files automatically when changes are detected (yes/no)
   - **Collection files:** Press F4 here to see the list of files in this collection.

   See “Add a file collection” on page 359 for explanation of the above fields.

```
                        Change/Show a File Collection

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  File Collection Name                                Configuration_Files
  New File Collection Name                           []
  File Collection Description                        [AIX and HACMP configu>
  Propagate files during cluster synchronization?     no +
  Propagate files automatically when changes are det  no +
  ected?
  Collection files +




F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-3 Change a file collection

Remove a file collection

1. Start HACMP File Collection Management by entering **smit cm_filecollection_menu** fast path.
2. Select **Manage File Collections.**
3. Select **Remove a File Collections.**

4. Select a file collection from the pop-up list.
5. Press Enter again to confirm the deletion of the file collection.

Change the automatic update timer

Here you can set the timer for how frequently HACMP checks the files in the collections are changed. Only one timer can be set for all file collection.

1. Start HACMP File Collection Management by entering **`smit cm_filecollection_menu`** fast path.
2. Select **Manage File Collections.**
3. Select **Change/Show Automatic Update Time.**
4. Select a file collection from the pop-up list.
5. Supply the **Automatic File Update Time** in minutes. The value should be between 10 minutes and 1440 minutes (one day).

Add files to a file collection

1. Start HACMP File Collection Management by entering **`smit cm_filecollection_menu`** fast path.
2. Select **Manage File in File Collections.**
3. Select **Add Files to a File Collection.**
4. Select a file collection from the pop-up list end press Enter.
5. On the SMIT panel you can check the current file list or you can add new files (See Figure 8-4 on page 363):
   - To get the list of current files in this collection: go down to **Collection Files** field and press F4.
   - To add new files go to **New files** field and type the file name here, that you like to add to the file collection. You can add only one file at a time. The file name should start with "/". You can only specify ordinary files here, you cannot add symbolic links, directory, pipe, socket, device file (/dev/*), files from /proc directory and ODM files from /etc/objrepos/* and /etc/es/objrepos/*.

```
                        Add Files to a File Collection

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  File Collection Name                                  app_files
  File Collection Description                           Application Config Fi>
  Propagate files during cluster synchronization?       no
  Propagate files automatically when changes are det    no
  ected?
  Collection files +
* New files                                            [/db2/app.config] /



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-4 Adding files to a file collection

Attention: You cannot add files here to the HACMP_Files collection.

Remove files from a file collection

1. Start HACMP File Collection Management by entering **smit cm_filecollection_menu** fast path.
2. Select **Manage File in File Collections.**
3. Select **Remove Files from a File Collection.**
4. Select a file collection from the pop-up list end press Enter.
5. Select one or more files from the list and press Enter. See Figure 8-5 on page 364.
6. Press Enter again to confirm.

Figure 8-5 Remove files from a file collections

Attention: You cannot remove files from the HACMP_Files collection by this way.

Manualy propagate files in a file collection

You can manualy synchronize any file collection (see Figure 8-6 on page 365):

1. Start HACMP File Collection Management by entering **`smit cm_filecollection_menu`** fast path.
2. Select **Propagate Files in File Collections.**
3. Select a file collection from the pop-up list end press Enter.
4. Press Enter again to confirm.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

Manual file collection propagation called.
The following file collections will be processed:
app_files
Starting file propagation to remote node p650n01.
Successfully propagated file /db2/appconfig1.txt to node p650n01.
Successfully propagated file /db2/databases.conf to node p650n01.
Successfully propagated file /db2/appdata.conf to node p650n01.
Total number of files propagated to node p650n01: 3




F1=Help            F2=Refresh          F3=Cancel           F6=Command
F8=Image           F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 8-6 Manual propagation of a file collection

8.3 User administration SV

In an HACMP cluster the user IDs and password must be synchronized all the time. If the user and group ID are not same across your cluster, your application cannot work and users cannot access their files on the shared storage. Additionally we suggest that you synchronize the passwords too, in case of a takeover, users can log in conveniently without spending hours finding out what their password are on the backup node.

There are several options to consider for user and password synchronization:

- Using CSPOC: HACMP provides a set of utilities in CSPOC for easy user administration. In “CSPOC user and group administration” on page 366, we introduce the CSPOC user and group management function.
- Network Information Server (NIS): this is not as widely used today as it was a few years ago, but it’s still a powerful solution for centralized user management. NIS configuration in HACMP is very well documented in *HACMP/ES Customization Examples,* SG24-4498.
- LDAP: This is the best solution for managing a large number of users in a complex environment. LDAP can be easily set up to work together with

HACMP. For more information about LDAP, see *Understanding LDAP - Design and Implementation,* SG24-4986.

8.3.1 CSPOC user and group administration

HACMP provides CSPOC tools for easy user, group and password administration. The following functions implemented:

- Add user
- List users
- Modify user attributes
- Remove user
- Change password
- Add, list, change and remove groups

Add a user

To add a user on all nodes in the cluster:

1. Start SMIT: **`smit`**.
   - Select **Communications Applications and Services**.
   - Select **HACMP for AIX**.
   - Select **System Management (CSPOC)**.
   - Select **HACMP Security and Users Management**.

   Or you can start CSPOC HACMP Security and User Management by entering **`smit cl_usergroup`** fast path.
2. Select **Users in a HACMP Cluster**.
3. Select **Add a User to the Cluster**.
4. Select on which nodes you want to create users. If you leave the **Select Nodes by Resource Group** field empty, the user will be created on all nodes. If you select a resource group here, then the user will be created only on that subset of nodes that belongs to that resource group. In case of a two-node cluster, leave this field blank.

   If you have more than two nodes in your cluster, then you can create users on a subset of nodes or on all nodes. In our four-node cluster example in 6.2, “Four-node configuration” on page 304 we have two resource groups that are binded to node1 and node2 only (rg1 and rg2). If you like to create a user only on this two node (e.g., they can use the application on node1 and node2, but they not allowed to log in to node3 and node4), then select here the appropriate resource group.

Table 8-1 Cross-reference of users, resource groups and nodes

Resource group	Nodes	Users
rg1	node1, node2	db2adm, db2inst, db2user
rg2	node2, node1	galamb, karesz
rg3	node3, node4	matyi, adrien
rg4	node4, node3, node2, node1	app1adm, app1user

Table 8-1 is a cross-reference between users, resource groups and nodes. It shows that in our example, the app1adm user will be created on all nodes (leave the `Select Nodes by Resource Group` field empty), while users such as “karesz” and “galamb” will be created only on node 1 and node2 (select “rg2” at the `Select Nodes by Resource Group` field). See Figure 8-7 below.

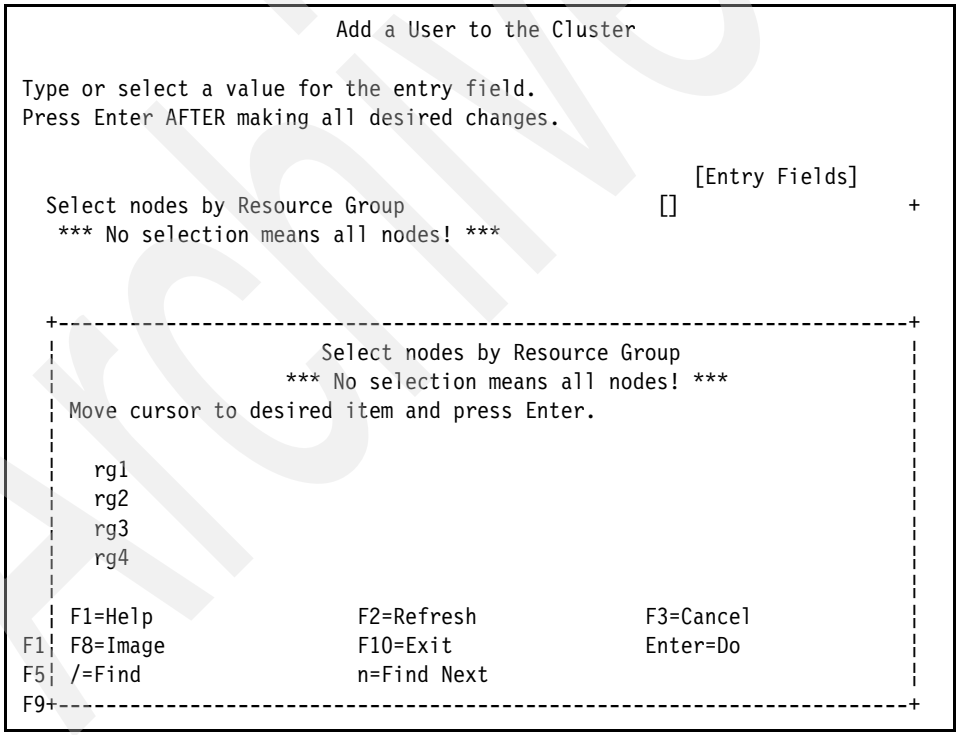

Figure 8-7 Select nodes by resource group

5. Create the user. You should supply the user name and other pertinent information just like when creating a normal user. You can specify the user ID

here, then CSPOC will check that the given ID is available on all nodes. If you leave the user ID field blank the user will be created with the first available ID on all nodes. See SMIT screen on Figure 8-8.

```
                          Add a User to the Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                   [Entry Fields]
  Select nodes by resource group
   *** No selection means all nodes! ***

* User NAME                                            [matyi]
  User ID                                              [] #
  ADMINISTRATIVE USER?                                  false +
  Primary GROUP                                        [] +
  Group SET                                            [] +
  ADMINISTRATIVE GROUPS                                [] +
  Another user can SU TO USER?                          true +
  SU GROUPS                                            [ALL] +
  HOME directory                                       [/home/matyi]
  Initial PROGRAM                                      []
  User INFORMATION                                     [Mr. Matyi]
[MORE...33]

F1=Help          F2=Refresh       F3=Cancel          F4=List
F5=Reset         F6=Command       F7=Edit            F8=Image
F9=Shell         F10=Exit         Enter=Do
```

Figure 8-8 Create a user on all cluster nodes

Attention: When creating a user with its home directory resides on a shared filesystem CSPOC doesn't check whether the filesystem is mounted or not. In this case CSPOC creates the user home directory under the empty mount point of the shared filesystem.You can fix this by moving the home directory under the shared filesystem.

If a user home directory is on a shared volume the user can log in only that node where the filesystem is currently mounted.

List cluster users

To list users in the cluster:

1. Start CSPOC HACMP Security and User Management by entering **`smit cl_usergroup`** command.
2. Select **Users in a HACMP Cluster**.
3. Select **List Users in the Cluster**.
4. Select on which nodes you want to list the users. If you leave the **Select Nodes by Resource Group** field empty the users will be listed on all nodes.

   If you select a resource group here, then CSPOC will list users from only that nodes that belong to the specified resource group.
5. Press Enter. See SMIT screen on Figure 8-9 on page 370.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

p650n01  root   0       /
p650n01  daemon 1       /etc
p650n01  bin    2       /bin
p650n01  sys    3       /usr/sys
p650n01  adm    4       /var/adm
p650n01  sshd   207     /var/empty
p650n01  matyi  302     /home/matyi
p650n01  adrien 305     /home/adrien
p650n01  karesz 307     /home/karesz
p650n01  db2adm 1000    /home/db2adm
p650n01  db2inst        1001    /home/db2inst
p650n01  db2user        1003    /home/db2user
p650n01  galam  312     /home/galamb
p650n02  root   0       /
p650n02  daemon 1       /etc
p650n02  bin    2       /bin
p650n02  sys    3       /usr/sys
p650n02  adm    4       /var/adm
p650n02  sshd   204     /var/empty
p650n02  matyi  302     /home/matyi
p650n02  adrien 305     /home/adrien
p650n02  karesz 307     /home/karesz
p650n02  db2adm 1000    /home/db2adm
p650n02  db2inst        1001    /home/db2inst
p650n02  db2user        1003    /home/db2user
p650n02  galam  312     /home/galamb

F1=Help             F2=Refresh          F3=Cancel           F6=Command
F8=Image            F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 8-9 Listing users in the cluster

Modify user attributes

To modify users attributes in the cluster:

1. Start CSPOC HACMP Security and User Management by entering **`smit cl_usergroup`** command.
2. Select **Users in a HACMP Cluster**.
3. Select **Change / Show Characteristics of a User in the Cluster**.

4. Select on which nodes you want to modify a users. If you leave the **Select Nodes by Resource Group** field empty any user can be modified on all nodes.

   If you select a resource group here, then you can modify a user that belongs to the specified resource group.

5. Enter the name of the user or press F4 to select from the pop-up user list.
6. Now you can modify the user attributes (See Figure 8-10):

```
                    Change / Show Characteristics of a User

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                  [Entry Fields]
* User NAME                                           adrien
  User ID                                            [305] #
  ADMINISTRATIVE USER?                                false +
  Primary GROUP                                      [staff] +
  Group SET                                          [staff] +
  ADMINISTRATIVE GROUPS                              [] +
  ROLES                                              [] +
  Another user can SU TO USER?                        true +
  SU GROUPS                                          [ALL] +
  HOME directory                                     [/home/adrien]
  Initial PROGRAM                                    [/usr/bin/ksh]
  User INFORMATION                                   [Mrs Adrien]
  EXPIRATION date (MMDDhhmmyy)                       [0]
  Is this user ACCOUNT LOCKED?                        false +
[MORE...36]

F1=Help            F2=Refresh         F3=Cancel          F4=List
F5=Reset           F6=Command         F7=Edit            F8=Image
F9=Shell           F10=Exit           Enter=Do
```

Figure 8-10 Modifying user attributes

Remove a user

1. Start CSPOC HACMP Security and User Management by entering **`smit cl_usergroup`** command.
2. Select **Users in a HACMP Cluster**.
3. Select **Remove a User from the Cluster**.

4. Select on which nodes you want to remove users. If you leave the **Select Nodes by Resource Group** field empty any user can be removed from all nodes.

   If you select a resource group here, then CSPOC will remove the user from only that nodes which belong to the specified resource group.

5. Enter the user name to remove or press F4 to select a user from the pop-up list.

6. **Remove AUTHENTICATION information:** select **Yes** to delete the user password and other authentication information. Select **No** to leave the user password in the /etc/security/passwd file. The default is **Yes**. See Figure 8-11.

```
                        Remove a User from the Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]

  Select nodes by resource group
   *** No selection means all nodes! ***

* User NAME                                           [suri] +
  Remove AUTHENTICATION information?                   Yes +



F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 8-11 Remove a user from the cluster

Add a group to the cluster

1. Start CSPOC HACMP Security and User Management by entering **`smit cl_usergroup`** command.

2. Select **Groups in a HACMP Cluster**.

3. Select **Add a Group to the Cluster**.

4. Select on which nodes you want to create a group. If you leave the **Select Nodes by Resource Group** field empty the group will be created on all nodes.

   If you select a resource group here, then CSPOC will create the group only on that subset of nodes which belong to the specified resource group. In our four-node cluster example in 6.2, “Four-node configuration” on page 304 we

have two resource groups that are bounded to node1 and node2 only (rg1 and rg2). If you like to create a group only on that two nodes, then select here the appropriate resource group.

Table 8-2 Cross-reference of groups, resource groups and nodes

Resource group	Nodes	Group
rg1	node1, node2	db2group
rg2	node2, node1	developers
rg3	node3, node4	itstaff
rg4	node4, node3, node2, node1	app1users

The Table 8-2 is a cross-reference between groups, resource groups and nodes. It shows that app1users present on all nodes (leave the `Select Nodes by Resource Group` field empty), while groups such as “db2group” will be created only on node1 and node2 (Select “rg1” at the `Select Nodes by Resource Group` field).

5. Create the group. See SMIT screen on Figure 8-12 on page 374. You should supply the group name, user list and other pertinent information just like when creating a normal group. Press F4 for the list of the available users to assign them with the group.

   You can specify the group ID here and CSPOC will check if the given ID is available on all nodes. If you leave the group ID field blank the group will be created with the first available ID on all nodes.

```
                        Add a Group to the Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Select nodes by resource group
   *** No selection means all nodes! ***

* Group NAME                                            [app1users]
  ADMINISTRATIVE group?                                  false +
  Group ID                                              [] #
  USER list                                             [db2adm,db2user] +
  ADMINISTRATOR list                                    [] +



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-12 Add a group to the cluster

List groups on the cluster

1. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.
2. Select **Groups in a HACMP Cluster**.
3. Select **List All Groups in the Cluster**.
4. Select on which nodes you want to list the groups. If you leave the **Select Nodes by Resource Group** field empty CSPOC lists all groups from all cluster nodes. If you select a resource group here, then CSPOC will list only that groups that are on the nodes which belong to the specified resource group.
5. CSPOC lists the groups and its attributes from the selected nodes as you can see on SMIT screenshoot on Figure 8-13 on page 375.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

p650n01  system 0       true    root    files
p650n01  staff  1       false adrien,karesz,db2adm,db2inst,db2user files
p650n01  bin    2       true    root,bin        files
p650n01  sys    3       true    root,bin,sys    files
p650n01  adm    4       true    bin,adm files
p650n01  security       7       true    root    files
p650n01  cron   8       true    root    files
p650n01  shutdown       21      true    files
p650n01  sshd   205     false   sshd    files
p650n01  hacmp  206     false   files
p650n01  haemrm 207     false   files
p650n01  db2group       208     false   db2adm,db2user  root    files
p650n02  system 0       true    root    files
p650n02  staff  1       false adrien,karesz,db2adm,db2user files
p650n02  bin    2       true    root,bin        files
p650n02  sys    3       true    root,bin,sys    files
p650n02  adm    4       true    bin,adm files
p650n02  security       7       true    root    files
p650n02  cron   8       true    root    files
p650n02  shutdown       21      true    files
p650n02  sshd   202     false   sshd    files
p650n02  hacmp  203     false   files
p650n02  haemrm 204     false   files
p650n02  db2group       208     false   db2adm,db2user  root    files

F1=Help           F2=Refresh          F3=Cancel           F6=Command
F8=Image          F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 8-13 List groups on the cluster

Change a group in the cluster

1. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.
2. Select **Groups in a HACMP Cluster**.
3. Select **Change / Show Characteristics of a Group in the Cluster**.
4. Select on which nodes you want to change the groups. If you leave the **Select Nodes by Resource Group** field empty you can modify any groups from all cluster nodes. If you select a resource group here, then CSPOC will change

only that groups that are on the nodes which belong to the specified resource group.

5. Change the group attributes. See Figure 8-14.

```
                Change / Show Group Attributes on the Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Select nodes by resource group

  Group NAME                                             db2group
  Group ID                                              [208] #
  ADMINISTRATIVE group?                                  false +
  USER list                                             [db2adm,db2user] +
  ADMINISTRATOR list                                    [root] +



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-14 Change / show group attributes on the cluster

Remove a group

1. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.
2. Select **Groups in a HACMP Cluster**.
3. Select **Remove a Group from the Cluster**.
4. Select from which nodes you want to remove the groups. If you leave the **Select Nodes by Resource Group** field empty CSPOC will remove the selected group from all cluster nodes. If you select a resource group here, then CSPOC will remove the group only from that nodes which belong to the specified resource group.
5. Select the group to remove. Press F4 to get a list of groups in the cluster.

Notes on using CSPOC user management

Some remarks regarding the CSPOS user and group administration:

- CSPOC User and password management requires to have the Cluster Communication daemon up and running on all nodes. You cannot use

CSPOC if any of your nodes are powered down. In this case you will get an error message similar to this: `clhaver[152]: cannot connect to node p650n01 rc=-1 errno=0`. However you can use CSPOC regardless of the state of the cluster.

- Be careful when selecting the nodes by the resource groups. You had to select exactly that nodes where the user or group what you like to modify or remove are present. You cannot modify or remove a user or group if that user or group is not present on any of the selected node.
- If you encounter any error using CSPOC check /tmp/cspoc.log for more information.
- The CSPOC user management cannot be used together with NIS or LDAP.

8.3.2 Password management

The HACMP CSPOC is a convenient way for the users to change their password on all cluster node at the same time. When somebody use the **passwd** utility from any node CSPOC propagates the new password to all nodes.

Set up CSPOP password management

The CSPOC password management utilities are disabled by default. Here are the steps how to enable it:

1. Change the system password utility to the cluster password program. On a standalone AIX machine the /usr/bin/passwd command is used to change a user's password. Now this command will be replaced by /usr/es/sbin/cluster/utilities/clpasswd which can change the password on the remote nodes too.
   a. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.
   b. Select **Passwords in an HACMP cluster**.
   c. Select **Modify System Password Utility**.
   d. Press F4 and select **Link to Cluster Password Utility** from the pop-up window. See Figure 8-15 on page 378.
   e. Select on which nodes you want to change the password utility. Just leave this field blank for all nodes. We suggest that you set up the cluster password utility on all nodes.

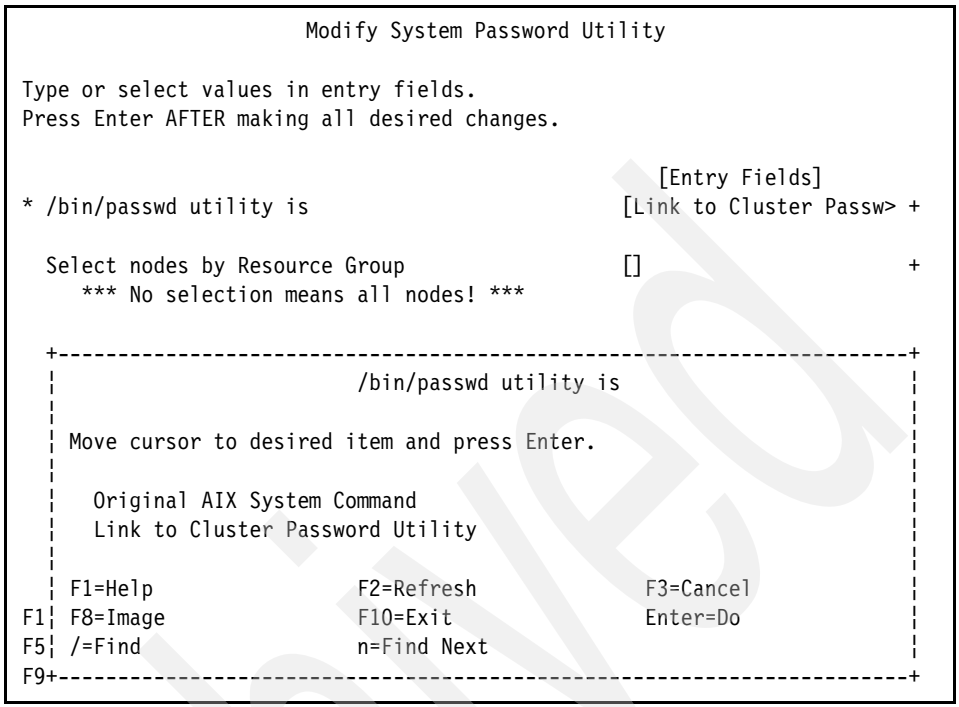

Figure 8-15 Modifying the system password utility

2. Create a list of users who can change their own password from any cluster node:

   a. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.

   b. Select **Passwords in an HACMP cluster**.

   c. Select **Manage List of Users Allowed to Change Password**.

   d. Now SMIT shows the users who are already allowed to change their password cluster-wide (see Figure 8-16 on page 379).

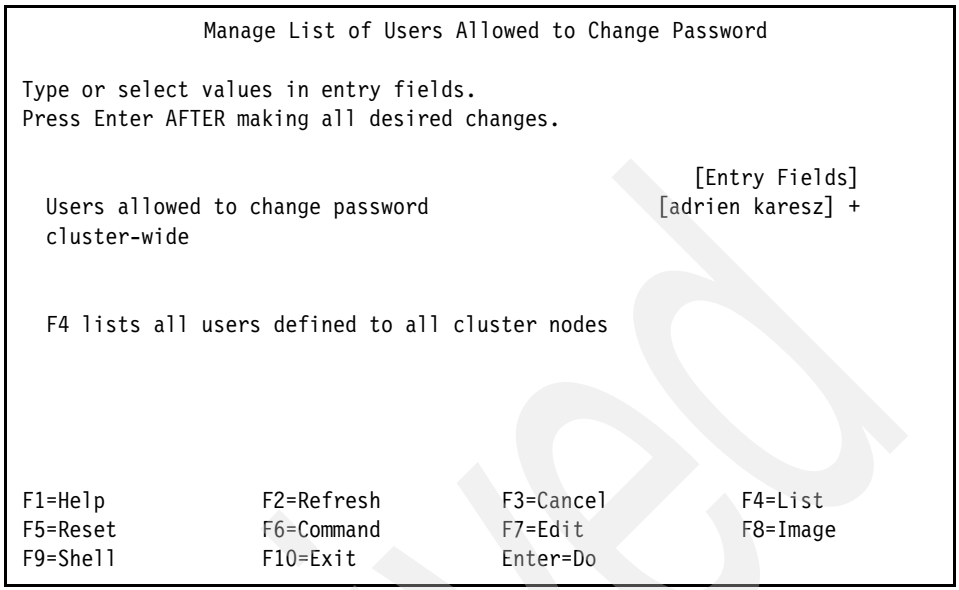

Figure 8-16 Managing the list of users allowed to change their password cluster-wide

e. To add users or change the list of the users who are allowed to change their password cluster-wide press F4 and select the user names from the pop-up list. Choose **ALL_USERS** to enable all current and future cluster users to use CSPOC password management. See Figure 8-17 on page 380.

   We suggest that you include only real life users here, and manually change the password for the technical users.

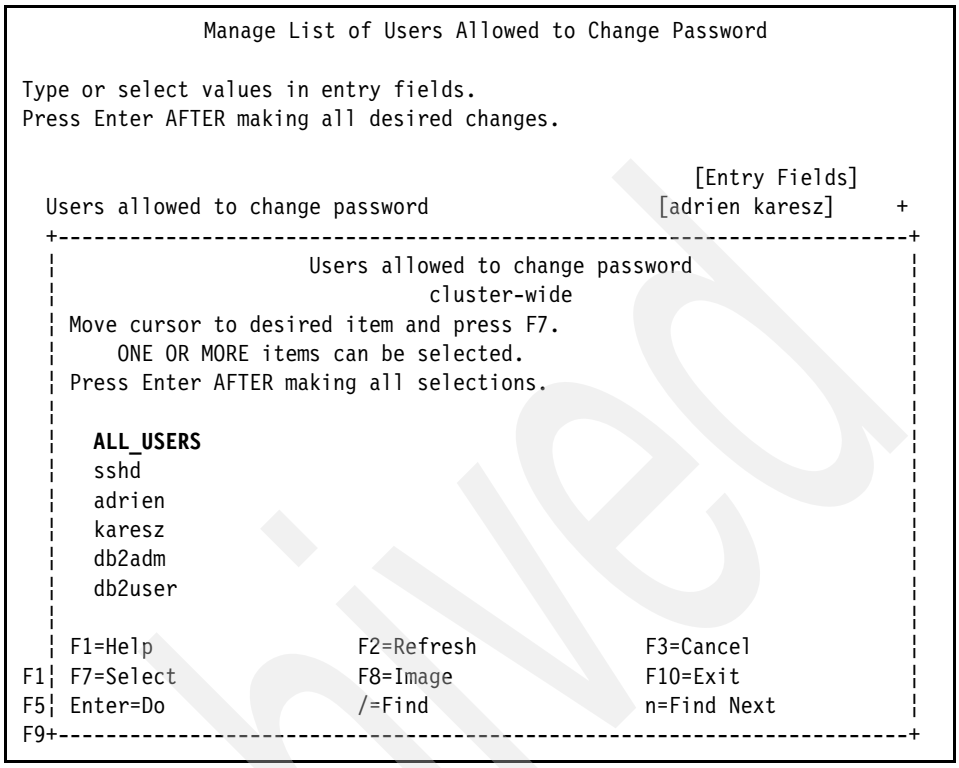

Figure 8-17 Selecting the users allowed to change their password cluster-wide

Attention: If you enable CSPOC password utilities for all users in the cluster, but you have users who are only on one node, then you get an error message similar like this:

```
# passwd joe
Changing password for "joe"
joe's New password:
Enter the new password again:
p650n02: clpasswdremote: User joe does not exist on node p650n02
p650n02: cl_rsh had exit code = 1, see cspoc.log and/or clcomd.log for more
information
```

The password is changed regardless of the error message.

Change a user password with CSPOC

1. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.

2. Select **Passwords in an HACMP cluster**.
3. Select **Change a User's Password in the Cluster**.
4. Select on which nodes you want to change the user's password. Just leave this field empty for all nodes. If you select a resource group here CSPOC change the password only on that nodes that belong to the resource group.
5. Type the user name or press F4 to select a user from the pop-up list.
6. Set **User must change password on first login** to **true** or **false** as you desire. See Figure 8-18 below.

```
                  Change a User's Password in the Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Selection nodes by resource group
   *** No selection means all nodes! ***

* User NAME                                            [karesz] +
  User must change password on first login?             true +



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-18 Change a user's password in the cluster

7. Press Enter and type the new password.

Tip: You can still use the AIX **passwd** command to change a specific user's password on all nodes.

Changing your own password

1. Start CSPOC HACMP Security and User Management by entering **smit cl_usergroup** command.
2. Select **Passwords in an HACMP cluster**.
3. Select **Change Current Users Password**.
4. Select on which nodes you want to change your password. Leave this field empty for all nodes. If you select a resource group here CSPOC change the password only on that nodes that belong to the resource group.

5. Your user name is shown on the SMIT screen. See Figure 8-19.

```
                        Change Current Users Password

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Selection nodes by resource group                    [] +
   *** No selection means all nodes! ***

  User NAME                                             adrien




F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-19 Change your own password

6. Press Enter and Change your password.

Now your password is changed on all the selected nodes.

Tip: You can use the **passwd** command to change your password on all nodes.

8.4 Shared storage management

C-SPOC utility simplifies maintenance of shared LVM components in the clusters. C-SPOC commands provide comparable functions in a cluster environment to the standard AIX 5L commands that work on a single node. By automating repetitive tasks on different nodes within the cluster, C-SPOC eliminates a potential source of errors and speeds up the cluster maintenance process. Although you can also use AIX 5L on each node to do these procedures, we suggest you to use C-SPOC wherever possible.

The C-SPOC commands only operate on both shared and concurrent LVM components that are defined as part of an HACMP resource group. When you use SMIT HACMP C-SPOC, it executes the command on the node that owns the LVM component (the node that has it varied on).

Note: The C-SPOC commands that modify LVM components require a resource group name as an argument. The LVM component that is the target of the command must be configured in the resource group specified. C-SPOC uses the resource group information to determine on which nodes it must execute the operation specified.

You can find some additional information about storage in Chapter 13, “Storage related considerations” on page 573.

8.4.1 Updating LVM components

When you change any definition of any shared LVM component in a cluster (including volume groups, logical volumes and file systems), the operation updates the AIX ODM data that describes the component on the local node and in the Volume Group Descriptor Area (VGDA) on all the disks in the volume group. This ODM update should be propagate to all nodes in the cluster in order to assure the proper cluster functionality.

If you make this LVM modifications through C-SPOC, the propagation of the ODM changes to all the nodes within the cluster will occur automatically.

If you make this LVM modifications using AIX commands on local node, you must propagate the ODM changes to all other nodes manualy. The LVM changes are related to the LVM structure of separate volume group.

Importing volume groups manualy

The regular AIX based procedure to propagate volume group ODM information to other nodes for non-concurrent volume groups is showed in the Example 8-1. You can use the same steps for enhanced concurrent volume groups as well. You can also use the equivalent AIX SMIT command instead of the command line.

Example 8-1 Importing AIX volume groups manually

```
Tasks performed on the local node (where the volume group is varied-on):
p630n03> lsvg -l app1vg
app1vg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
app1vglog           jfs2log    1     2     2    open/syncd    N/A
app1lv              jfs2       200   400   4    open/syncd    /app1
p630n03> umount /app1
p630n03> varyoffvg app1vg
p630n03> ls -l /dev/app1vg
crw-r-----   1 root     system       90,  0 Jun 21 13:09 /dev/app1vg

Tasks performed on all the other nodes:
```

```
p630n02> lspv |grep app1vg
vpath0          000685bf8595e225                    app1vg
vpath1          000685bf8595e335                    app1vg
vpath2          000685bf8595e445                    app1vg
vpath3          000685bf8595e559                    app1vg
p630n02> exportvg app1vg
p630n02> importvg -y app1vg -n -V90 vpath0
p630n02> chvg -a n app1vg
p630n02> varyoffvg app1vg
```

Attention: Ownership and permissions on logical volume devices are reset when a volume group is exported and then re-imported. After exporting and importing, a volume group is owned by root:system. Applications, such as some database servers, that use raw logical volumes may be affected by this. You must check the ownership and permissions before exporting VG and restore them manualy in case they are not root:system as default.

Instead of export / import command you can use the **importvg -L VGNAME HDISK** command on other nodes, but be aware that -L option could not be performed, if logical volume or file system has been removed on primary node. The **importvg -L** command preserves the logical volume devices ownership.

From HACMP 5.2 the usage of enhanced concurrent volume groups simplifies the LVM administration, since some LVM ODM modifications are directly propagated to the all cluster nodes. You could find more information about enhanced concurrent volume groups in Chapter 13, “Storage related considerations” on page 573.

and having the gsclvmd daemon running with cluster services.

Lazy update

HACMP provides special feature for automatic synchronization of LVM ODM information during fail-over in case, when recorded VGDA timestamps are different. This functionality is called lazy update. AIX 5L updates this time-stamp whenever the LVM component is modified. When another cluster node attempts to vary on the volume group, HACMP compares its copy of the time-stamp with the time-stamp in the VGDA on the disk. If the values are different, the HACMP software exports and re-imports the volume group before activating it. If the time-stamps are the same, HACMP activates the volume group without exporting and re-importing.

Note: We recommend to provide periodical cluster verification and to follow the recommended administration procedure rather than rely on lazy update feature.

However the lazy update could fail in case, that logical volume or file system has been removed on original node.

Note: From HACMP 5.2, HACMP does not require lazy update processing for enhanced concurrent volume groups, as it keeps all cluster nodes updated with the LVM information. Please refer to the 13.1.1, "Enhanced concurrent" on page 574.

Importing volume groups automatically

HACMP provides additional feature for importing VG automatically. It enables to automatically import shareable volume groups onto all the destination nodes in the resource group.This could be done through the **Extended Resource Configuration** SMIT menu. Automatic import allows you to create a volume group and then add it to the resource group immediately, without manually importing it onto each of the destination nodes in the resource group.

Run s`mit hacmp` > **Extended Configuration > Extended Resource Configuration > HACMP Extended Resource Group Configuration > Change/Show Resources and Attributes for a Resource Group** then select resource group and set the **Automatically Import Volume Groups** to *true*.

You have to ensure the following conditions in order for HACMP to import available volume groups:

- Volume group names must be the same across cluster nodes and unique to the cluster.
- Logical volumes and filesystems must have unique names.
- All physical disks must be known to AIX 5L and have PVIDs assigned.
- The physical disks on which the volume group resides are available to all of the nodes in the resource group.

Before this is necessary to have Auto Discover option set to true. You can set this by running `smit hacmp` > **Extended Configuration > Discover HACMP-related Information from Configured Nodes**.

Importing volume groups using C-SPOC

You could import the shared volume group to all nodes in cluster by C-CPOC utility. You can do this by running `smit hacmp` > **System Management (C-SPOC)**

> HACMP Logical Volume Management > Shared Volume Groups > Import a Shared Volume Group and selecting the volume group. A list of volume groups appears. Enhanced concurrent volume groups are also included as choices in pick-list for non-concurrent resource groups. After selecting the volume group, a list of physical disk devices appeared. After selecting physical disk for importing, fill the fields in importvg SMIT screen appears, as shown in Figure 8-20.

```
 Import a Shared Volume Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  Resource Group Name                                 C1ORG1
  VOLUME GROUP name                                   testvg
  Reference node                                      panther
  PHYSICAL VOLUME name                                vpath2
  Volume group MAJOR NUMBER                           [100]                +#
  Make this VG Concurrent Capable?                    no                    +
  Make default varyon of VG Concurrent?               no                    +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-20 C-SPOC importvg screen

At the end of this action, you can run the discovery process so that the new volume group is included in pick-list for future actions.

8.4.2 C-SPOC Logical Volume Manager

C-SPOC Logical Volume Manager menu gives to you the possibility to perform LVM commands from the menus and screens that are very similar to the AIX LVM SMIT menus (running **`smit lvm`**). The difference with C-SPOC is, that for the most task is necessary to select local node or the relevant resource (resource group, volume group), first. After that usually appears the SMIT screen, which looks very familiar to the AIX LVM replica. For AIX administrators, familiar with AIX LVM SMIT menus, would be quite easy to use the following C-SPOC options.

The next C-SPOC advantage is that from HACMP 5.3 all VPATH disk operations, that are currently supported on AIX 5L, are now supported by C-SPOC. You must have SDD 1.3.1.3 or greater installed.

You can select LVM C-SPOC menu for logical volume management cluster-wide by running **`smit cl_admin`** > **HACMP Logical Volume Management**. There you could find several menu selection possibilities:

- Shared Volume Groups.
  - List All Shared Volume Groups.
  - Create a Shared Volume Group.
  - Create a Shared Volume Group with Data Path Devices.
  - Set Characteristics of a Shared Volume Group.
  - Import a Shared Volume Group.
  - Mirror a Shared Volume Group.
  - Unmirror a Shared Volume Group.
- Shared Logical Volumes.
  - List All Shared Logical Volumes by Volume Group.
  - Add a Shared Logical Volume.
  - Set Characteristics of a Shared Logical Volume.
  - Show Characteristics of a Shared Logical Volume.
  - Change a Shared Logical Volume.
  - Remove a Shared Logical Volume.
- Shared File Systems.
  - Journaled File Systems.
  - Enhanced Journaled File Systems. For both file system types the submenus are as following:
    - Add an (Enhanced) Journaled File System.
    - Add an (Enhanced) Journaled File System on a Previously Defined Logical Volume.
    - List All Shared File Systems.
    - Change / Show Characteristics of a Shared (Enhanced) Journaled File System.
    - Remove a Shared File System.
- Synchronize Shared LVM Mirrors.
  - Synchronize by Volume Group.
  - Synchronize by Logical Volume.
- Synchronize a Shared Volume Group Definition.

You can find some more description about the specific tasks in the 8.4.5, “Examples” on page 389.

8.4.3 C-SPOC Concurrent Logical Volume Management

C-SPOC logical volume manager menu gives to you the possibility of performing LVM commands on the concurrent volume groups from the menus and screens that are very similar to the C-SPOC LVM menus, described in previous section.

You can select C-SPOC menu for concurrent logical volume management cluster-wide by running **`smit cl_admin`** > **HACMP Concurrent Logical Volume Management**. You could select among three main options:

- Concurrent Volume Groups.
  - List All Concurrent Volume Groups.
  - Create a Concurrent Volume Group.
  - Create a Concurrent Volume Group with Data Path Devices.
  - Set Characteristics of a Concurrent Volume Group.
  - Import a Concurrent Volume Group.
  - Mirror a Concurrent Volume Group.
  - Unmirror a Concurrent Volume Group.
- Concurrent Logical Volumes.
  - List All Concurrent Logical Volumes by Volume Group.
  - Add a Concurrent Logical Volume.
  - Set Characteristics of a Concurrent Logical Volume.
  - Show Characteristics of a Concurrent Logical Volume.
  - Remove a Concurrent Logical Volume
- Synchronize Concurrent LVM Mirrors.
  - Synchronize by Volume Group.
  - Synchronize by Logical Volume.

You can find some more description about the specific tasks in the 8.4.5, “Examples” on page 389.

8.4.4 C-SPOC Physical Volume Management

You can select C-SPOC menu for physical volume management and SDD virtual path management within whole by running **`smit cl_admin`** > **HACMP Physical Volume Management**. You could select among several options:

- Add a Disk to the Cluster.

- Remove a Disk From the Cluster.
- Cluster Disk Replacement.
- Cluster Data Path Device Management.
  - Display Data Path Device Configuration.
  - Display Data Path Device Status.
  - Display Data Path Device Adapter Status.
  - Define and Configure all Data Path Devices.
  - Add Paths to Available Data Path Devices.
  - Configure a Defined Data Path Device.
  - Remove a Data Path Device.
  - Convert ESS hdisk Device Volume Group to an SDD VPATH Device Volume Group.
  - Convert SDD VPATH Device Volume Group to an ESS hdisk Device Volume Group.
- Configure Disk/Site Locations for Cross-Site® LVM Mirroring. You could find more information about the Cross-Site mirroring implementation with an example in the Chapter 16, “HACMP with cross-site LVM” on page 623.

You can find some more description about the specific tasks in the 8.4.5, “Examples” on page 389.

8.4.5 Examples

In this session we present some examples and scenarios with C-SPOC LVM management. The examples we show are as follows:

1. Displaying the existing datapath configuration.
2. Displaying device adapter status.
3. Adding an enhanced concurrent VG on vpaths.
4. Adding a VG into a RG.
5. Creating a new LV.
6. Creating a new jfslog2 LV.
7. Creating a new FS.
8. Adding an additional vpath to a VG.
9. Increasing a FS size.
10. Removing a file system.

11. Synchronizing a volume group definition across the cluster nodes.

For our examples we use 3-node cluster with IP over aliasing configured, using also heartbeat over aliasing. The storage is ESS with two paths, so we did our actions on vpath devices. Figure 8-21 shows our testing cluster setup.

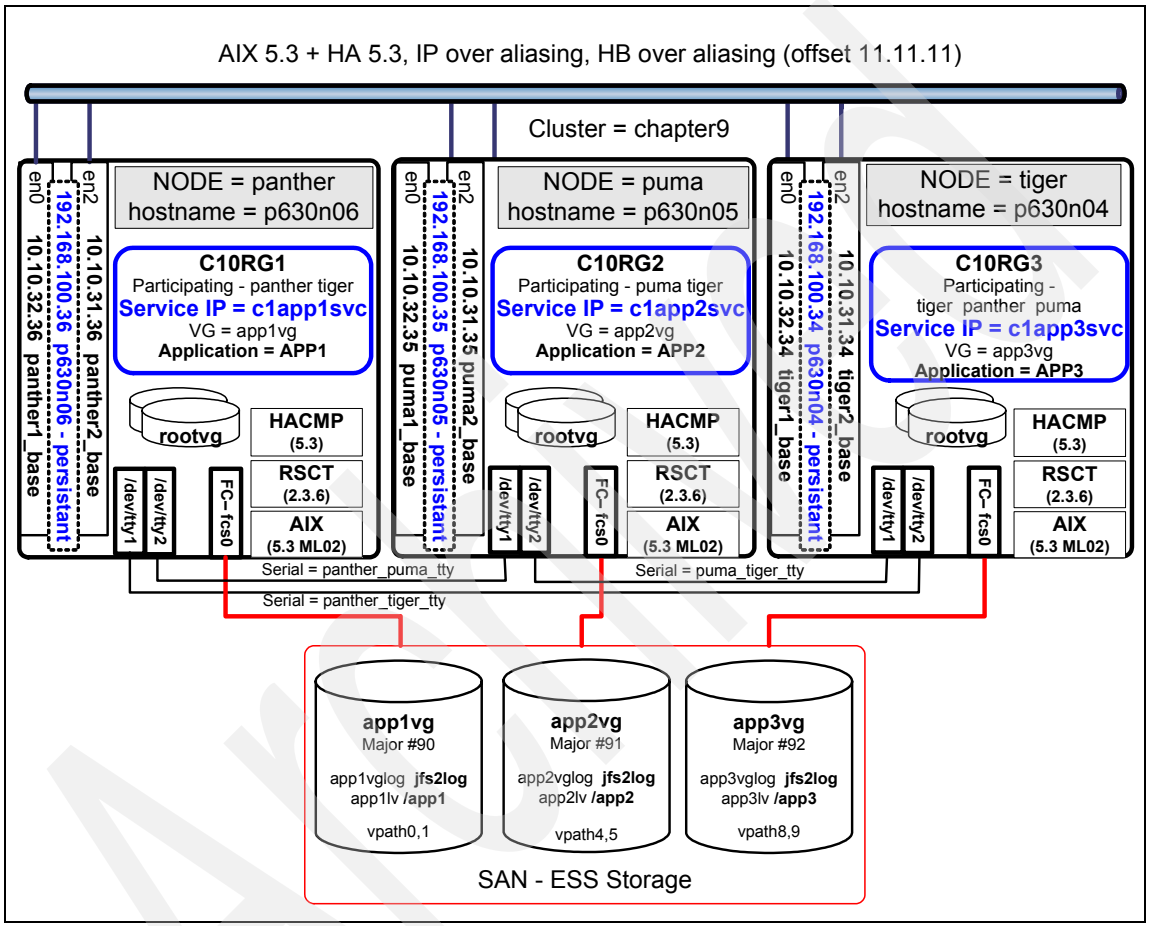

Figure 8-21 C-SPOC LVM testing cluster setup

Displaying the existing datapath configuration

This example shows how to display the existing vpath configuration with PVID information on the selected node.

We start the C-SPOC datapath configuration display by running **`smit cl_admin`** > **HACMP Physical Volume Management > Cluster Data Path Device**

Management > Display Data Path Device Configuration and then we select the node panther. The displayed information is shown in Example 8-2.

Example 8-2 C-SPOC data path display

```
PVID: 000685cf86a5dfe6
panther: vpath8 (Avail pv app3vg) 10822513 = hdisk11 (Avail ) hdisk23 (Avail )
PVID: 000685cf86a5e0f6
panther: vpath9 (Avail pv app3vg) 10922513 = hdisk12 (Avail ) hdisk24 (Avail )
PVID: 000685cf86a5e1fa
panther: vpath10 (Avail pv ) 10A22513 = hdisk13 (Avail ) hdisk25 (Avail )
PVID: 000685cf86a5e2fa
panther: vpath11 (Avail pv ) 10B22513 = hdisk14 (Avail ) hdisk26 (Avail )
PVID: 000685cf86aaa0a3
panther: vpath4 (Avail pv app2vg) 10422513 = hdisk7 (Avail ) hdisk19 (Avail )
PVID: 000685cf86aaa3a3
panther: vpath5 (Avail pv app2vg) 10522513 = hdisk8 (Avail ) hdisk20 (Avail )
PVID: 000685cf86aaa63e
```

Displaying a device adapter status

The example shows how to display the FC adapter status information on the selected node.

We start the C-SPOC device adapter status display by running **`smit cl_admin`** > **HACMP Physical Volume Management > Cluster Data Path Device Management > Display Data Path Device Adapter Status**, after this we select the node panther. The displayed information is shown in Example 8-3.

Example 8-3 C-SPOC display data path adapter.

```
panther:

Active Adapters :1

Adpt#     Name   State     Mode             Select     Errors  Paths  Active
    0   fscsi0  NORMAL   ACTIVE             109346          0     24       4
```

Adding an enhanced concurrent VG on vpaths

The following example shows how to add new enhanced concurrent volume group into the cluster.

Before creating a shared VG for the cluster using C-SPOC, we check if the following prerequisites are existing:

- All disk devices are properly attached to the cluster nodes.

- All disk devices are properly configured on all cluster nodes and the devices are listed as available on all nodes.
- Disks have a PVID.

We add the enhanced concurrent VG into the cluster by running:

smit cl_admin > HACMP Concurrent Logical Volume Management > Concurrent Volume Groups > Create a Concurrent Volume Group with Data Path Devices

Then we select the two nodes panther and puma. We choose this two nodes since we are planning to add testvg to C10RG1 resource groups, where this two nodes participate. After that we select appropriate vpaths in the SMIT menu as shown in Example 8-4. The PVID selection list shows only the unoccupied devices on the selected nodes.

Example 8-4 PVID selection list

```
                          Physical Volumes

Move cursor to desired item and press F7.
    ONE OR MORE items can be selected.
Press Enter AFTER making all selections.

[MORE...1]
  #   panther: vpath11: hdisk14 hdisk26
  #   tiger: vpath11: hdisk15 hdisk27
> 000685cf86af4e3c
  #   panther: vpath3: hdisk6 hdisk18
  #   tiger: vpath3: hdisk7 hdisk19
> 000685cf86af4bc7
  #   panther: vpath2: hdisk5 hdisk17
  #   tiger: vpath2: hdisk6 hdisk18
[MORE...3]
```

In the screen that appears after selecting vpaths, we enter all the necessary fields for creating VG as shown in Example 8-5:

Example 8-5 Creating VG

```
Create a Shared Volume Group with Data Path Devices

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                 [Entry Fields]
  Node Names                                         panther
  PVID                                               000685cf86af4bc7 0006>
```

```
  VOLUME GROUP name                                   [testvg]
  Physical partition SIZE in megabytes                 128                +
  Volume group MAJOR NUMBER                           [100]
  Enhanced Concurrent Mode                             true               +
  Enable Cross-Site LVM Mirroring Verification         false              +

  Warning:
  Changing the volume group major number may result
  in the command being unable to execute
  successfully on a node that does not have the
  major number currently available. Please check
  for a commonly available major number on all nodes
  before changing this setting.

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

After creating enhanced concurrent volume group testvg, we checked on the node tiger and panther, if information about new VG is registered. As expected, volume group ODM information has been successfully updated on the both nodes.

Adding a VG into a RG

Since the C-SPOC LVM commands only operate on the components that are defined as part of an HACMP resource group, we add VG to the resource group C10RG1.

This examples shows how to add VG to the existing resource group with the feature “Automatically Import Volume Groups”. This feature is useful in case, when you create new volume groups only on one local node and using “Automatically Import Volume Groups” feature imports VG to all other nodes participating selected resource group. If VG is added from C-SPOC, LVM ODM are already synchronized within all the nodes in cluster and “Automatically Import Volume Groups” in not necessary to be used.

We add VG to the existing resource group C10RG1 using **Change/Show Resources and Attributes for a Resource Group menu** in SMIT. We run **smit hacmp > Extended Resource Configuration > HACMP Extended Resource Group Configuration > Change/Show Resources and Attributes for a Resource Group,** then we select our RG C10RG1 and fill the fields as shown in Example 8-6

Example 8-6 Adding VG in RG with automatic import option

```
Change/Show All Resources and Attributes for a Custom Resource Group
```

```
Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                   [Entry Fields]
  Resource Group Name                                  C10RG1
  Participating Nodes (Default Node Priority)          cobra viper

  Startup Policy                                       Online On Home Node O>
  Fallover Policy                                      Fallover To Next Prio>
  Fallback Policy                                      Never Fallback

  Service IP Labels/Addresses                          [app1svc testvg]       +
  Application Servers                                  [APP1]                 +

  Volume Groups                                        [app1vg ]              +
  Use forced varyon of volume groups, if necessary     false                  +
  Automatically Import Volume Groups                   true                   +
[MORE...20]

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Note: The “Automatically Import Volume Groups” feature imports volume group only on the nodes, that participate in resource group, where the new volume group had been added. In our example puma node does not participate to the C10RG1 resource group, so testvg is imported only on tiger node.

After adding the volume group testvg into the resource group C10RG1, we synchronize the configuration in order to propagate resource group modifications to other nodes. We run **smit hacmp > Extended Configuration > Extended Resource Configuration > Extended Verification and Synchronization** and run synchronization with verification using default settings. You can find more information about synchronization and verification in the 8.6, “Cluster verification and synchronization” on page 401

Creating a new LV

The following example shows how to create a new LV in the selected VG, which is already active as part of RG.

We add the LV test01lv in the testvg VG, running **smit cl_admin > HACMP Logical Volume Management > Shared Logical Volumes.** Then we select the testvg VG from the displayed list, same as shown in Example 8-14 on page 400.

On the next screen that appears after, we select devices for the LV allocation as shown in Example 8-7.

Example 8-7 C-SPOC creating new LV -1

```
                          Physical Volume Names

 Move cursor to desired item and press F7.
     ONE OR MORE items can be selected.
 Press Enter AFTER making all selections.

   Auto-select
   panther vpath2
   panther vpath3
```

After that, we filled the necessary fields as shown in Example 8-8.

Example 8-8 C-SPOC creating new LV - 2

```
Add a Shared Logical Volume

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                              [Entry Fields]
  Resource Group Name                             C1ORG1
  VOLUME GROUP name                               testvg
  Reference node
* Number of LOGICAL PARTITIONS                    [10]                    #
  PHYSICAL VOLUME names
  Logical volume NAME                             [test01lv]
  Logical volume TYPE                             [jfs2]                  +
  POSITION on physical volume                      middle                 +
  RANGE of physical volumes                        minimum                +
  MAXIMUM NUMBER of PHYSICAL VOLUMES               []                      #
    to use for allocation
  Number of COPIES of each logical                 1                      +
    partition
[MORE...11]

F1=Help            F2=Refresh         F3=Cancel          F4=List
F5=Reset           F6=Command         F7=Edit            F8=Image
F9=Shell           F10=Exit           Enter=Do
```

The test01lv is created and information is propagated on the puma node. After that, we verified the cluster running:

`smit hacmp` > Problem Determination Tools > HACMP Verification

And, verification passed without errors.

Creating a new jfslog2 LV

For adding a new jfs2log logical volume testjlog2lv in testvg volume group we used the same procedure as described previously in “Creating a new LV” on page 394. In the C-SPOC creating new LV - 2 screen, shown in Example 8-8 on page 395, we select the jfs2log as type of the LV:

```
Logical volume TYPE [jfs2log]
```

After adding the jfs2log logical volume in the testvg volume group, we did the jfs2log formatting on panther node, where testvg volume group is active. We run the following command:

```
p630n06 >logform  /dev/testjloglv
logform: destroy /dev/rtestjloglv (y)?y
```

Creating a new FS

The following example shows how to create a jfs2 file system on the previous created logical volume. We make this task by running:

```
smit cl_admin->HACMP Logical Volume Management->Shared File
Systems->Enhanced Journaled File Systems->Add an Enhanced Journaled
File System on a Previously Defined Logical Volume
```

Then we select the previously created enhanced logical volume from SMIT list. After that we fill all necessary fields as shown in Example 8-9.

Example 8-9 C-SPOC creating jfs2 file system

```
Add an Enhanced Journaled File System on a Previously Defined Logical Volume

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  Node Names                                          panther,tiger,
  LOGICAL VOLUME name                                 test01lv
* MOUNT POINT                                        [/cltestfs]
  PERMISSIONS                                         read/write             +
  Mount OPTIONS                                       []                     +
  Block Size (bytes)                                  4096                   +
  Inline Log?                                         no                     +
  Inline Log size (MBytes)                            []                      #


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
```

```
F9=Shell            F10=Exit            Enter=Do
```

The /cltestfs file is created. We check the contents of /etc/filesystems on the node tiger and information about new file system has been added to the bottom of the /etc/filesystems file as expected.

Adding an additional vpath to a VG

This example shows how to add a new vpath to an existing shared volume group. We make this action using C-SPOC by running:

smit cl_admin > HACMP Logical Volume Management > Shared Volume Groups > Set Characteristics of a Shared Volume Group > Add a Volume to a Shared Volume Group

Then we select the volume group from the SMIT screen, the same one as shown in Example 8-14 on page 400. In the next screen that follows, we selected virtual path devices that we want to add to the testvg volume group:

```
¦ > panther vpath11
¦ > panther vpath6
```

After that we need just to confirm the action from the selection informative SMIT screen as shown in Example 8-10.

Example 8-10 C-SPOC Adding vpath to the share volume group

```
Add a Volume to a Shared Volume Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
  Resource Group Name                                 C10RG1
  VOLUME GROUP name                                   testvg
  Reference node                                      panther
  VOLUME names                                        vpath11 vpath6

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

We check a the vpath configuration on the node tiger with C-SPOC datapath configuration display by running:

smit cl_admin > HACMP Physical Volume Management > Cluster Data Path Device Management > Display Data Path Device Configuration

Then we select the node tiger. The command output shows that new vpaths are correctly added to testvg volume group.

Increasing a FS size

The following example shows the way, how to increase a share file system size with C-SPOC. We make this action by running:

SMIT **cl_admin > HACMP Logical Volume Management > Shared File Systems > Enhanced Journaled File Systems > Change / Show Characteristics of a Shared Enhanced Journaled File System**

Then SMIT file system selection list appears and we select /cltestfs filesystem as shown in Example 8-11.

Example 8-11 CSPOC file system changing - file system selection

```
          Enhanced Journaled File System Name and Resource Group

Move cursor to desired item and press Enter.

  # Resource Group         File System
    C1ORG1                 /app1
    C1ORG1                 /cltestfs
    C1ORG2                 /app2
    C1ORG3                 /app3

F1=Help                F2=Refresh             F3=Cancel
F8=Image               F10=Exit               Enter=Do
/=Find                 n=Find Next
```

After selecting the filesystem we enter the size of additional GB as shown in Example 8-12.

Example 8-12 CSPOC file system changing - SMIT screen fields

```
Change/Show Characteristics of a Shared Enhanced Journaled File System

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
  Resource Group Name                                 C1ORG1
  File system name                                    /cltestfs
  NEW mount point                                    [/cltestfs]
  SIZE of file system                                [+2G]
  Mount GROUP                                        []
  PERMISSIONS                                         read/write             +
  Mount OPTIONS                                      []                      +
  Start Disk Accounting?                              no                     +
```

```
  Block Size (bytes)
  Inline Log?
  Inline Log size (MBytes)

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Removing a file system

This example shows the way, how to remove a shared file system from the cluster, using C-SPOC. Before starting the action of file system removal, it is necessary to unmount this file system manualy from the local node. We make this with the following command:

umount /cltestfs

After that we run:

smit cl_admin > HACMP Logical Volume Management > Shared File Systems > Enhanced Journaled File Systems > Remove a Shared File System

Then we selected the file system from the similar SMIT pick-list as shown in Example 8-11 on page 398. After that we need just to confirm the action on the next SMIT screen. The screen is shown in Example 8-13.

Example 8-13 CSPOC remove a file system SMIT screen

```
                        Remove a Shared File System

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
  Resource Group Name                                 C10RG1
* FILE SYSTEM name                                    /cltestfs              +
  Remove Mount Point                                  yes                    +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Synchronizing a volume group definition across the nodes

If you make any changes on VG, LV or FS definition on local node, using AIX LVM commands instead of using the C-SPOC, you need to propagate LVM ODM changes to the other participating nodes. You can synchronize LVM ODM information among the cluster nodes by using C-SPOC feature **smit cl_admin >**

HACMP Logical Volume Management > Synchronize a Shared Volume Group Definition and then we select the volume group from a SMIT pick-list as shows the Example 8-14.

Example 8-14 C-SPOC Synchronizing VG definition - SMIT pick-list

```
¦   #Resource Group                    Volume Group                ¦
¦    C1ORG1                           app1vg                      ¦
¦   #Resource Group                    Volume Group                ¦
¦    C1ORG2                           app2vg                      ¦
¦   #Resource Group                    Volume Group                ¦
¦    C1ORG3                           app3vg                      ¦
¦   #Resource Group                    Volume Group                ¦
¦    C1ORG1                           testvg                      ¦
```

8.5 Time synchronization

We strongly suggest that you use time synchronization on the cluster nodes. Some application demands this, but anyway having the xntpd running will ease the cluster administration.

Configuring NTP clients

If your network already has an established time server you can set up the cluster nodes to get the accurate time information from it.

1. Modify the /etc/ntp.conf file to contains this three lines:

```
server 192.169.1.254 # your ntp time server's IP address goes here
driftfile /etc/ntp.drift
tracefile /etc/ntp.trace
```

2. Start xntpd daemon:

```
startsrc -s xntpd
```

3. Set up xntpd to start automatically at boot time. Remove the comment mark from the beginning of the following line in the /etc/rc.tcpip file:

```
start /usr/sbin/xntpd "$src_running"
```

Setting up a time server

If you do not have a timeserver you can easily set up one in the cluster. Without external time information the servers' time will not be accurate, but identical.

1. Choose one of the nodes to act as an NTP server. Preferably this should be the node with the highest priority in the cluster.
2. Modify the /etc/ntp.conf file on the NTP server:

```
disable auth
server 127.127.1.1 prefer # use the local clock as preferred
fudge 127.127.1.1 stratum 4
driftfile /etc/ntp.drift
```

3. Edit /etc/ntp.conf file on the other nodes:

```
server 10.10.1.1 # your cluster's ntp time server's base IP address goes
here
driftfile /etc/ntp.drift
tracefile /etc/ntp.trace
```

   We suggest that you use the NTP server node's base address or a persistent interface if any.

4. Edit the /etc/rc.tcpip file on all nodes so xntpd will start automatically:

```
start /usr/sbin/xntpd "$src_running"
```

5. Start xntpd on all nodes:

```
startsrc -s xntpd
```

8.6 Cluster verification and synchronization

Verification and synchronization of the HACMP cluster assures that all resources used or controlled by the HACMP, are configured appropriately and that all rules regarding resource ownership and other parameters are consistent across all cluster nodes.

The HACMP cluster is storing the information about all cluster resources and cluster topology, as well as some additional parameters in HACMP-specific object classes in the ODM. HACMP ODM files have to be consistent across all cluster nodes to assure a correct cluster behavior as designed. Cluster verification verify the consistency of HACMP ODM files across all nodes as well as verify if a HACMP ODM information is consistent with an AIX ODM information. If the verification is successful, that means the cluster configuration is synchronized across all the nodes. Synchronization takes effect immediately on an active cluster. Cluster synchronization synchronizes an ODM across the node by applying the ODM information from the node, where synchronization has been initialized, to the other nodes.

Attention: An inconsistent cluster topology, resources or other parameters across the nodes within the cluster may cause that the cluster will not work as designed

We recommend you to verify your the cluster configuration, whenever you configure, reconfigure, or update a cluster, or whenever you are changing

operating system parameters, that could affect cluster resources. We also recommend you to run cluster verification periodically.

8.6.1 Cluster verification and synchronization using SMIT

Using SMIT (running **smit hacmp** command) you have three different verification and synchronization menu paths for cluster verification:

- *Initialization and Standard Configuration* path
- *Extended Configuration* path
- *Problem Determination Tools* path

Initialization and Standard Configuration verification path

You can use the “Initialization and Standard Configuration” verification path by running:

smit hacmp > Initialization and Standard Configuration > Verify and Synchronize HACMP Configuration

When you are using the SMIT “Initialization and Standard Configuration” path, synchronization automatically follows a successful verification. There is no additional selectable options in SMIT menu. Automatically correct errors feature is always active while using “Initialization and Standard Configuration” path. You can find more information about automatically correct errors feature in 8.6.4, “Running automatically corrective actions during verification” on page 408.

Extended Configuration verification path

When you are using the Extended Configuration path, you have different options for types of verification and you can choose whether to synchronize or not.

You can take the following procedure to use Extended Configuration verification path on your cluster:

1. Run **smit hacmp -> Extended Configuration -> Extended Verification and Synchronization.**
2. Change the field parameters (for normal verification and synchronization leave parameters as default, as shown in Examples...) and press Enter.
3. The Figure 8-22 on page 404 shows the SMIT screen when cluster is active (DARE - Dynamic Reconfiguration).
4. The Figure 8-23 on page 404shows the SMIT screen when cluster is inactive.
5. After verification is done, you can see either it has been successful or it failed.

The Extended Verification and Synchronization path parameters depends on that if the cluster is active or inactive on the node, where verification is initiated. On active cluster the SMIT screen parameters are:

1. Emulate or Actual: option *Emulate* runs verification in emulation mode, no changes are applied while *Actual* applies changes.
2. Verify changes only: option *No* runs the full check of topology and resources while *Yes* verifies only the changes appeared from the time of last verification. This feature only relates to HACMP ODM files!
3. Logging: option *Verbose* sends full output to the console that is normally logged in clverify.log file.

Initiating verification on inactive cluster SMIT screen parameters are:

1. “Verify, Synchronize or both”: *option Verify* runs verification only, *Synchronize* runs synchronization only, *Both* runs both verification and synchronization according *Force synchronization if verification failes* option.
2. “Automatically correct errors found during verification”: for details description see the 8.6.4, “Running automatically corrective actions during verification” on page 408.
3. “Force synchronization if verification failes”: option *No* stops synchronization / verification procedure after verification errors are detected while *Yes* forces synchronization regardless verification result. In general we don’t recommend forcing the synchronization. In case of some specific situations, when the synchronization has to be forced, be sure to understand the consequences of the cluster configuration changes.
4. “Verify changes only”: *No* runs the full check of topology and resources while Yes verifies only the changes appeared from the time of last verification. This feature only relates to HACMP ODM files!
5. Logging: option *Verbose* sends full output to the console that is normaly logged in clverify.log file.

Note: Synchronization could be initiated either on active or inactive cluster. If some nodes are inactive, synchronization could be initiated only from an active node, using DARE (Dynamic Reconfiguration). You could find more information about DARE in the 8.6.2, “Dynamic cluster reconfiguration - DARE” on page 405.

```
        HACMP Verification and Synchronization (Active Cluster Nodes Exist)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Emulate or Actual                                   [Actual]             +
* Verify changes only?                                [No]                 +
* Logging                                             [Standard]           +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-22 Synchronization and verification screen - active cluster

```
                    HACMP Verification and Synchronization

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Verify, Synchronize or Both                         [Both]               +
* Automatically correct errors found during           [No]                 +
  verificatio

* Force synchronization if verification fails?        [No]                 +
* Verify changes only?                                [No]                 +
* Logging                                             [Standard]           +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-23 Synchronization and verification screen - inactive cluster

Problem Determination Tools verification path

You can use the “Problem Determination Tools” verification path by running **`smit hacmp` > Problem Determination Tools > HACMP Verification > Verify HACMP Configuration.**

If you are using the Problem Determination Tools path, you have more options for verification, such as custom defined verification method definition, but no

possibility to select synchronization as well. You can see the SMIT screen of the Problem Determination Tools verification path in Figure 8-24.

Note: Verification, using Problem Determination Tools path, could be initiated either from active or inactive node.

```
                        Verify HACMP Configuration

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  HACMP Verification Methods                         Pre-Installed        +
         (Pre-Installed, none)
  Custom Defined Verification Methods                []                   +
  Error Count                                        []                   #
  Log File to store output                           []
  Verify changes only?                               [No]                 +
  Logging                                            [Standard]           +


F1=Help           F2=Refresh        F3=Cancel         F4=List
F5=Reset          F6=Command        F7=Edit           F8=Image
F9=Shell          F10=Exit          Enter=Do
```

Figure 8-24 Verification screen using "Problem Determination Tools" path

In case that verification fails, is necessary to correct the errors and repeat the verification as soon as possible. The messages output from the verification indicate, where the error occurred (for example, on a node, a device, or a command). Section 8.6.3, "Verification log files" on page 407 describes the location and purpose of the verification logs.

8.6.2 Dynamic cluster reconfiguration - DARE

HACMP gives the possibility to make some changes to both the cluster topology and to the cluster resources while the cluster is running. This feature is called the dynamic reconfiguration, or DARE. You can make a combination of resource and topology changes via one dynamic reconfiguration operation making the whole operation faster, especially for complex configuration changes.

DARE supports, that resource and topology changes could be done in one operation. Starting with HACMP 5.3, DARE is supported in HACMP/XD configurations.

Attention: Do not make configuration changes or perform any action that affects a resource, if any node in the cluster is in a forced down state.

Attention: Dynamic reconfiguration is not supported during a cluster migration to a new version of HACMP.

You can make the following changes to cluster resources in an active cluster, dynamically:

- Add, remove, or change an application server.
- Add, remove, or change application monitoring.
- Add or remove the contents of one or more resource groups.
- Add, remove, or change a tape resource.
- Add or remove one or more resource groups.
- Add, remove, or change the order of participating nodes in a resource group.
- Change the node relationship of the resource group.
- Change resource group processing order.
- Add, remove or change the fallback timer policy associated with a resource. group. The new fallback timer will not have any effect until the resource group is brought online on another node.
- Add, remove or change the settling time for resource groups.
- Add or remove the node distribution policy for resource groups.
- Add, change, or remove parent/child or location dependencies for resource groups (some limitations apply here).
- Add, change, or remove inter-site management policy for resource groups
- Add, remove, or change pre- or post-events.

The dynamic reconfiguration can be initiated only from an active node, it means from a node having the cluster daemons started. You must make changes form a node that is up so that the cluster can be synchronized.

Before making changes to a cluster definition, ensure that:

- The same version of HACMP is installed on all nodes.
- Some nodes are up and running HACMP and they are able to communicate with each other. No node should be in a forced down state.

- The cluster is stable and the hacmp.out log file does not contain recent event errors or config_too_long events.

Depending on your specific cluster configuration and on the specific changes you plan to implement on you cluster environment, there are a lot a different possibilities and possible limitations while using dynamic reconfiguration utility. You have to understand all the consequences of some cluster configuration changes, so we recommend you to read the HACMP for AIX Administration Guide about the details before starting to make dynamic changes on your cluster environment.

8.6.3 Verification log files

During a cluster verification, HACMP collects configuration data from all the nodes as it runs through a series of checks. The verbose output is saved to the /var/hacmp/clverify/clverify.log file. The log file is rotated.

The following output shows the /var/hacmp/clverify/ directory contents with verification log files:

```
[p650n01][/var/hacmp/clverify]> ls -l
total 2024
-rw-------   1 root     system        99451 Jun 21 00:00 clverify.log
-rw-------   1 root     system        99956 Jun 20 12:13 clverify.log.1
-rw-------   1 root     system        98639 Jun 20 00:00 clverify.log.2
-rw-------   1 root     system        98639 Jun 19 00:00 clverify.log.3
-rw-------   1 root     system        98639 Jun 18 00:00 clverify.log.4
-rw-------   1 root     system        98549 Jun 17 18:46 clverify.log.5
-rw-------   1 root     system        98549 Jun 17 18:24 clverify.log.6
-rw-------   1 root     system        98866 Jun 17 18:02 clverify.log.7
-rw-------   1 root     system        98866 Jun 17 17:49 clverify.log.8
-rw-------   1 root     system        99091 Jun 17 16:58 clverify.log.9
-rw-------   1 root     system         9296 Jun 21 00:00 clverify_daemon.log
drwx------   4 root     system          256 Jun 17 11:42 fail
drwx------   4 root     system          256 Jun 21 00:00 pass
drwx------   4 root     system          256 Jun 20 12:12 pass.prev
```

On the node, where you initiate the verification utility, detailed information is collected into log files, which contain a record off all data collected and the tasks performed. These log files are written to the following directories and are used by a service technician to determine the location of errors:

- /var/hacmp/clverify/pass/nodename/ - if verification succeeds
- /var/hacmp/clverify/fail/nodename/ - if verification fails

Note: Verification requires 4 MB of disk space per each node in the /var filesystem in order to run. Typically, the /var/hacmp/clverify/clverify.log files require additional 1–2 MB of disk space. 18 MB of disk space is recommended for a four-node cluster.

8.6.4 Running automatically corrective actions during verification

HACMP 5.3 gives the possibility to run some automatic corrective actions during the cluster verification and synchronization. The activation of this option depends of the path, you are using for verification and synchronization.

The automatic corrective action feature could correct only some types of errors, that are detected during the cluster verification. The following list presents the errors that could be solved using this feature:

- HACMP shared volume group time stamps are not up to date on a node.
- The /etc/hosts file on a node does not contain all HACMP-managed IP addresses.
- SSA concurrent volume groups need unique SSA node numbers.
- A filesystem is not created on a node, although disks are available.
- Disks are available, but the volume group has not been imported to a node.
- Required /etc/services entries are missing on a node.
- Required HACMP snmpd entries are missing on a node.

Initialization and Standard Configuration verification path

When you use the “Initialization and Standard Configuration” verification path, automatically corrected errors feature is always active and it is not possible to disable.

Note: No automatic corrective actions take place during a DARE.

Extended Configuration verification path

When you use the “Extended Configuration” verification path, the activation ability of the automatically corrected errors feature depends on the cluster status. Basically you can disable this feature or you can run it in one of the two modes:

- Interactively (menu selection *Interactively*), when verification detects a correctable condition related to importing a volume group or to exporting and re-importing mount points and filesystems, you are prompted to authorize a corrective action before verification continues.

- Automatically (menu selection *Yes*), when verification detects that any of the error conditions exists, as listed in section Conditions That Can Trigger a Corrective Action, it takes the corrective action automatically without a prompt.

If the cluster is inactive, you can select the mode of automatically corrected errors feature directly in the “Extended Configuration” verification path menu by running **`smit hacmp`** **-> Extended Configuration -> Extended Verification and Synchronization** as showed in the Figure 8-23 on page 404. You change the mode with the “Automatically correct errors found during verification” field, by setting it as *Yes, No*, *Interactively*.

If the cluster is active, automatic corrective action feature is enabled by default. You can change a mode of the automatic corrective actions feature for the active cluster directly in the SMIT cluster start menu. You run **`smit hacmp`** > **System Management (C-SPOC) > Manage HACMP Services** > **Start Cluster Services** by selecting values to *Yes*, *No* or *Interactive.* This will set the automatic corrective action mode for:

1. Cluster “Extended Configuration” verification path.
2. The automatic cluster verification in phase off starting cluster services on a node or rejoining a node the cluster. You can find more information about in the 8.6.5, “Automatic cluster verification” on page 409.
3. The automatic cluster verification that runs periodically. You can find more information about in the 8.6.5, “Automatic cluster verification” on page 409.

Problem Determination Tools verification path

Using this verification path, an activation of the automatically corrected errors feature is not possible.

8.6.5 Automatic cluster verification

HACMP provides automatic verification in the following cases:

- Each time you start cluster services on a node
- Each time a node rejoins the cluster
- Every 24 hours

During automatic verification and synchronization, HACMP discovers and corrects several common configuration issues. This automatic behavior ensures that if you had not manually verified and synchronized your cluster prior to starting cluster services, HACMP will do so. Automatic verification and synchronization is often simply referred to as verification.

Using SMIT menus you can set the parameters for periodically **Automatic cluster verification checking** utility, by running **`smit hacmp`** **> Problem Determination Tools > HACMP Verification > Automatic Cluster Configuration Monitoring.** You can find the following fields in SMIT screen:

- Automatic cluster configuration verification. Here you can enable or disable the utility, by selecting either *Disable* or *Enable.*
- Node name. Here you can select nodes where the utility will run. By selecting the option *default* means all nodes or you can select one particular node.
- HOUR (00 - 23). Here you can define time, when the utility will start. Default value is 00:00 (midnight) and it could be set on any time.

Table 8-25 shows the SMIT screen for “Automatic Cluster Configuration Monitoring” parameters setting.

`smit clautover.dialog`

```
             Automatic Cluster Configuration Monitoring

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Automatic cluster configuration verification       Enabled             +
  Node name                                          Default             +
* HOUR (00 - 23)                                    [00]                +


F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 8-25 Automatic Cluster Configuration Monitoring

You can check the verification result of an automatic cluster verification in the verification log files. The default location for them is the /var/hacmp/clverify/ directory. You can find more about verification log files in 8.6.3, “Verification log files” on page 407.

8.7 Monitoring HACMP

By design, the HACMP provides high available application environment by masking or eliminating failures that appears either on hardware or software side of the high available environment. Masking the failures means that the active resources are moved from a failed component to the redundant one. So all high

available applications continue to work and the clients access and use them despite the failure.

As a result, it is possible that a component in the cluster has failed and that you are unaware of the fact. The danger here is that, while HACMP can survive one or possibly several failures, each failure that escapes your notice threatens a cluster's ability to provide a highly available environment, as the redundancy of cluster components is diminished.

To avoid this situation, we recommend you to regularly check and monitor the cluster. HACMP provides also various utilities that help you with cluster monitoring and with cation as follows:

- Automatic cluster verification. You can find more about this in the 8.6.5, "Automatic cluster verification" on page 409.
- Cluster status checking utilities.
- Resource Groups Information Commands.
- Topology Information Commands.
- Log files.
- Error notification.
- Application monitoring.
- Measuring application availability.
- Monitoring clusters from the enterprise system administration and monitoring tools (Tivoli, NetView®).

You can use either ASCII SMIT or WebSMIT to configure and manage the cluster.

8.7.1 Cluster status checking utilities

clstat

`/usr/es/sbin/cluster/clstat` is very useful tool that you can use for cluster status monitoring. It uses the **`clinfo`** library routines to display various information about the cluster, including name and state of the nodes, interfaces and resource groups.

This utility requires subsystem **clinfoES** to be active on the nodes, where **`clstat`** command is initiated.

The clstat utility is supported in two modes: ASCII mode and X Window mode. ASCII mode can run on any physical or virtual ASCII terminal, including xterm or aixterm windows. If the cluster node runs graphical mode, **`clstat`** displays the

output in graphical window. Before running the command ensure, that the DISPLAY variable is exported to the X server and X clients access is allowed.

The Figure 8-26 shows syntax of clstat command.

```
(single ASCII display)
  clstat -o [-c ID | -n name] [-s]

  (ASCII mode)
  clstat -a [-c ID | -n name] [-i] [-r interval][-s]

  (X mode)
   xclstat [-c ID | -n name] [-r interval] [-D debug_level][-s]
```

Figure 8-26 Clstat command syntax

clstat -a //runs the program in ASCII mode.

clstat -o //runs the program once in ASCII mode and exits (useful for capturing output from a shell script or cron job).

clstat -s //displays service labels that are both up and down, otherwise displays only service labels, which are active.

The Example 8-15 shows the **clstat -o** command output from our test cluster:

Example 8-15 clstat -o command output

```
clstat - HACMP Cluster Status Monitor
               -------------------------------------

Cluster: migr1  (1120388255)
Wed Jul  6 14:13:15 CDT 2005
               State: UP               Nodes: 3
               SubState: STABLE

        Node: panther           State: UP
           Interface: panther1_base (0)         Address: 10.10.31.36
                                                State:   UP
           Interface: panther2_base (0)         Address: 10.10.32.36
                                                State:   DOWN
           Interface: tty1_patnh (1)            Address: 0.0.0.0
                                                State:   UP
           Interface: tty2_panth (2)            Address: 0.0.0.0
                                                State:   UP
           Interface: c1app1svc (0)             Address: 192.168.100.86
                                                State:   UP
           Resource Group: C1ORG1                       State:  On line
```

```
Node: puma              State: UP
   Interface: puma1_base (0)          Address: 10.10.31.35
                                      State:   UP
   Interface: puma2_base (0)          Address: 10.10.32.35
                                      State:   DOWN
   Interface: tty1_puma (1)           Address: 0.0.0.0
                                      State:   UP
   Interface: tty_puma (3)            Address: 0.0.0.0
                                      State:   UP
   Interface: c1app2svc (0)           Address: 192.168.100.85
                                      State:   UP
   Resource Group: C1ORG2                     State:  On line

Node: tiger             State: UP
   Interface: tiger1_base (0)         Address: 10.10.31.34
                                      State:   UP
   Interface: tiger2_base (0)         Address: 10.10.32.34
                                      State:   DOWN
   Interface: tty2_tiger (2)          Address: 0.0.0.0
                                      State:   UP
   Interface: tty_tiger (3)           Address: 0.0.0.0
                                      State:   UP
   Interface: c1app3svc (0)           Address: 192.168.100.84
                                      State:   UP
   Resource Group: C1ORG3                     State:  On line
```

cldump

Another useful utility is **cldump** (**/usr/es/sbin/cluster/utilities/cldump**), It provides a snapshot of the key cluster status components: the cluster itself, the nodes in the cluster, the network interfaces connected to the nodes, and the resource groups status on each node.

Cldump utility does not have any parameter options, so you simply run cldump from command line.

8.7.2 Cluster status and services checking utilities

Checking cluster subsystem status

You can check the HACMP or RSCT subsystem status by running the **lssrc** command with **-s** or **-g** switches. It displays subsystem name, group, PID and status (active or inoperative).

lssrc -s subsystem_name //displays subsystem information for specific subsystem.

lssrc -g subsystem_group_name //displays subsystem information for all subsystems in specific group.

Attention: From the HACMP Version 5.3, cluster manager daemon *clstrmgrES* is initiated from the init process, so it starts automatically at boot time. The Cluster Manager must be running before any cluster services can start on a node. Since the clstrmgr daemon is now a long running process, you cannot use **lssrc -s clstrmgrES** to determine the state of the cluster. Use /usr/es/sbin/cluster/clstat or any other utility, described in this section, instead.

The Figure 8-27 shows subsystem names and group names for all subsystems, used by HACMP.

```
Subsystem_name   group_name

RSCT subsystems used by HACMP:
 topsvcs          topsvcs
 grpsvcs          grpsvcs
 grpglsm          grpsvcs
 emsvcs           emsvcs
 emaixos          emsvcs
 ctrmc            rsct
HACMP subsystems:
 clcomdES         clcomdES
 clstrmgrES       cluster
optional HACMP subsystems:
 clinfoES         cluster
```

Figure 8-27 Subsystem names and group names used by HACMP

clshowsrv

You have another possibility to display the status of HACMP subsystems by using **clshowsrv** command (**/usr/es/sbin/cluster/utilities/cldump**). It displays the status of all subsystems, used by HACMP or the status of the selected subsystem. The command output format is the same as **lssrc -s** command output.

The Figure 8-28 shows the syntax of the clshowsrv command.

```
clshowsrv [-a|-v] [clstrmgrES|clinfoES|clcomdES]
```

Figure 8-28 clshowsrv command syntax

clshowsrv -a //displays status of HACMP subsystem: clstrmgrES, clinfoES and clcomdES

clshowsrv -v //displays status of all HACMP and RSCT subsystems

The Example 8-16 shows output of **clshowres** command from our test cluster, when cluster services are running.

Example 8-16 clshowres -v command output

```
Status of the RSCT subsystems used by HACMP:
Subsystem         Group            PID          Status
 topsvcs          topsvcs          22756        active
 grpsvcs          grpsvcs          21858        active
 grpglsm          grpsvcs                       inoperative
 emsvcs           emsvcs           24932        active
 emaixos          emsvcs           28982        active
 ctrmc            rsct             13430        active

Status of the HACMP subsystems:
Subsystem         Group            PID          Status
 clcomdES         clcomdES         15738        active
 clstrmgrES       cluster          26498        active

Status of the optional HACMP subsystems:
Subsystem         Group            PID          Status
 clinfoES         cluster          26260        active
```

You can also run the command **clshowsrv -v** using SMIT menus: **smit hacmp > System Management (C-SPOC) > Manage HACMP Services > Show Cluster Services.**

8.7.3 Topology information commands

cltopinfo

cltopinfo command (**/usr/es/sbin/cluster/utilities/cltopinfo**) lists the cluster topology information using an alternative format that's easier to read and understand.

The Figure 8-29 shows the **cltopinfo** command syntax.

```
cltopinfo [-c] [-n] [-w] [-i]
```

Figure 8-29 Cltopinfo command syntax

You could use also SMIT menus to display different formats of the topology information, by running **smit hacmp > Extended Configuration > Extended Topology Configuration > Show HACMP Topology** and selecting desired format. The Figure 8-30 shows the SMIT menus for topology information view with different format options. This selections are consistent with command options, showed in the Figure 8-29 on page 415.

smit cm_show_menu

```
                          Show HACMP Topology

Move cursor to desired item and press Enter.

  Show Cluster Topology
  Show Cluster Definition
  Show Topology Information by Node
  Show Topology Information by Network
  Show Topology Information by Communication Interface


F1=Help             F2=Refresh          F3=Cancel           F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-30 SMIT cluster topology menu

topsvcs service

You could run the **lssrc -ls topsvcs** command to monitor the heartbeat activity, based on the topology service. The output of the *topsvcs* daemon activity shows you all heartbeats related information for all active network paths. Pointing to *Missed HBs*, *Packets sent*, *Packets* received and Errors fields in the output for the specific network path, gives you an information about the specific heartbeat activity. The Example 8-17 shows the part of the **lssrc -ls topsvcs** command output on our test cluster. The emphasized words point you to the interesting part of output information.

Example 8-17 lssrc -ls topsvcs command output

```
Subsystem         Group            PID     Status
 topsvcs          topsvcs          811144  active
Network Name   Indx Defd Mbrs St Adapter ID      Group ID
migr1_eth_01_0 [ 0]    3    3  S 11.11.1.11      11.11.1.13
migr1_eth_01_0 [ 0] en2          0x42cc1f00      0x42cc1f1b
HB Interval = 1.000 secs. Sensitivity = 10 missed beats
Missed HBs: Total: 64 Current group: 64
Packets sent    : 8893 ICMP 17 Errors: 0 No mbuf: 0
Packets received: 10616 ICMP 31 Dropped: 0
NIM's PID: 774256
rs232_1        [ 3]    2    2  S 255.255.0.2     255.255.0.2
rs232_1        [ 3] tty1         0x82cc1f02      0x82cc1f05
```

```
HB Interval = 2.000 secs. Sensitivity = 5 missed beats
Missed HBs: Total: 0 Current group: 0
Packets sent    : 5841 ICMP 0 Errors: 0 No mbuf: 0
Packets received: 6250 ICMP 0 Dropped: 0
NIM's PID: 893118
```

8.7.4 Resource groups information commands

clrginfo

You can displays a resource group's attributes within the cluster using **clRGinfo** command (**/usr/es/sbin/cluster/utilities/clRGinfo).** The command output shows a report on the location and state of one or more specified resource groups. The output of the command displays both the global state of the resource group as well as the special state of the resource group on a local node.

Figure 8-31 shows the clRGinfo command syntax.

```
clRGinfo [-h][-v][-a][-s|-c][-p][-t][-d][groupname1] [groupname2] ...
```

Figure 8-31 clRGinfo command syntax

clRGinfo -v // Displays the priority override location and a resource group's active timers.

clRGinfo -p // Displays the resource group's startup, fallover and fallback preferences.

clRGinfo -t // Queries the Cluster Manager on the local node only.

clRGinfo -c // Command, it lists the output in a colon separated format.

clRGinfo -a // Command provides information on what resource group. movements take place during the current cluster event - if you run it while a cluster event is being processed.

The resource group status is shown as:

- *Online*: the resource group is currently operating properly.
- *Offline*: the resource group is not operating in the cluster and is currently not in an error.
- *Acquiring*: A resource group is currently coming up on one of the nodes in the cluster. In normal conditions status changes to Online.

- *Releasing*: The resource group is in the process of being released from ownership by one node. In normal conditions stratus changes to Offline.
- *Error*: The resource group has reported an error condition. User interaction is required.
- *Unknown*: The resource group's current status cannot be attained, possibly due to loss of communication, because of an error with any resource in resource group or because a resource group dependency is not met.

If the cluster services are not running on the local node, the command determines a node where the cluster services are active and obtains the resource group information from the active cluster manager.

Instead of clrginfo you can use **clfindres** command, which is a link to clRGinfo. (**/usr/es/sbin/cluster/utilities/clfindres**)

Example 8-18 shows the clRGinfo command output on our test cluster

Example 8-18 clRGinfo command output

```
-----------------------------------------------------------------------------
Group Name     Group State                  Node
-----------------------------------------------------------------------------
rg1            ONLINE                       p650n01
               OFFLINE                      p650n02

rg2            ONLINE                       p650n02
               OFFLINE                      p650n01
```

8.7.5 Log files

HACMP is storing all information about cluster occurrence and writes the messages it generates to the system console and to several log files. Because each log file contains a different subset of the types of messages generated by HACMP, you can get different views of cluster status by viewing different log files.

Using C-SPOC utility you can do the following actions on log files

- View/Save/Delete HACMP Event Summaries. Using this selection you could display the contents or saves either deletes the cluster event summary.
- View Detailed HACMP Log Files. With this selection you could you could display HACMP scripts log (/tmp/hacmp.out), HACMP system log (/usr/es/adm/cluster.log), C-SPOC system log file (/tmp/cspoc.log).
- Change/Show HACMP Log File Parameters. Using this option you can set the debug level (high/low) and formatting option (default, standard, html-low, html-high) for the selected node.

- Change/Show Cluster Manager Log File Parameters. With this selection you could set the cluster manager debug level (standard/high)
- Change/Show a Cluster Log Directory. Using this menu you could set the new directory for the selected log file, as described below.
- Collect Cluster log files for Problem Reporting. Use this feature for collecting cluster snap data (clsnap command), that are necessary for additional problem determination and analyses. You could select here the debug option, include RSCT log files and select nodes included in this data collection. If not specify, default location for snap collection is in /tmp/ibmsupt/hacmp/ directory for clsnap and in /tmp/phoenix.snapOut directory for phoenix snap.

The list of all hacmp logs and their purpose description is as follows:

- strmgr.debug: generated by the clstrmgrES daemon, default directory is [/tmp].
- cluster.log: generated by cluster scripts and daemons, default directory is [/usr/es/adm].
- cluster.mmddyyyy: cluster history files generated daily, default directory is [/usr/es/sbin/cluster/history].
- cl_sm.log: generated by the cluster Shared Memory library, default directory is [/tmp].
- cspoc.log: generated by CSPOC commands, default directory is [/tmp]
- dms_loads.out - Generated by deadman's switch activity, default directory is [/tmp].
- emuhacmp.out: generated by the event emulator scripts, default directory is [/tmp].
- hacmp.out: generated by event scripts and utilities as they executes, default directory is [/tmp].
- clavan.log: generated by Application Availability, default directory is [/var/adm].
- clverify.log: generated by Cluster Verification utility, default directory is [/var/hacmp/clverify].
- clcomd.log: generated by clcomd daemon, default directory is [/var/hacmp/clcomd].
- clcomddiag.log: generated by clcomd daemon, debug information, default directory is [/var/hacmp/clcomd].
- clconfigassist.log: generated by Two-Node Cluster Configuration, default directory is [/var/hacmp/log].
- clutils.log: generated by cluster utilities, default directory is [/var/hacmp/log].

- cl_testtool.log: generated by the Cluster Test Tool, default directory is [/var/hacmp/log].

You have to ensure enough space for all log files in filesystems. The necessary amount of space in /var depends of a number of the nodes in cluster. You could calculate the value for every node, using the following estimations:

- 2MB should be free for writing the clverify.log[0-9] files.
- 4MB per node for writing the verification data from the nodes.
- 20MB for writing the clcomd log information.
- 1MB per node for writing the ODM cache data.

For example, for four node cluster you need 2 + 4x4 + 20 + 4x1 = 42 MB space in /var filesystem.

No estimation about necessary space of /tmp directory could be done, since there resides some debug log information files as well as some non-rotating log files. The size of this logs depends of activities, configuration and status of the cluster. From practical experience we recommend about 50 MB free space in /tmp directory.

You can change a default directory of the specific logfile in the SMIT menu, running **smit hacmp > System Management (C-SPOC) > HACMP Log Viewing and Management > Change/Show a Cluster Log Directory** and then selecting specific log file (SMIT fast-path: **smit clusterlog_redir.select)**. The default log directory is changes for all nodes in cluster. You have to make a cluster synchronization after changing the log parameters.

Attention: We recommend you to use local filesystems for new log location rather than shared or NFS filesystems. Having logs on shared or NFS filesystems may cause problems if the filesystem needs to unmount during a fall-over event. Redirecting logs to shared or NFS filesystems may also prevent cluster services from starting during node reintegration.

In addition, cluster generates also some debug files. They resides in /tmp directory. The contents of this files depends of the selected debug level.

- clinfo.debug - records an output generated by the event scripts as they run.
- clsmuxtrmgr.debug - is the smux peer function log file.
- clstlrmgr.debug - contains time-stamped, formatted messages generated by HACMP clstrmgrES activity.

8.7.6 Error notification

You can use the AIX 5L Error Notification facility to add an additional layer of high availability to an HACMP environment. You can add notification for failures of resources for which HACMP does not provide recovery by default.

You could find more information about the Automatic error notification, using and configuring the error notification with some examples in the 12.3, “Error notification” on page 560.

8.7.7 Application monitoring

After cluster services are started on the specific node and all resources are online, we recommend you to check, if all applications are started as well and if all services, supposed to be provided by applications, are available. You can verify, if all application process are running and if all other resources like file system, required by application, are available.

Starting with HACMP 5.2, 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.

However, HACMP cluster provides possibility to monitor the specified application automatically. In case if HACMP detects a process death or an application failure, it attempt to restart them. The application monitoring works in one of the two ways:

- The process application monitoring detects the termination of one or more processes of an application, using RSCT Resource Monitoring and Control (RMC).
- The custom application monitoring checks the health of an application with a custom monitor method at user-specified polling intervals.

You could configure the **Configure HACMP Application Monitoring** with SMIT menus by running **smit hacmp -> Extended Resource Configuration -> Extended Resource Configuration -> HACMP Extended Resources Configuration -> Configure HACMP Applications -> Configure HACMP Application Monitoring** and then select between **Configure Process Application Monitors** and **Configure Custom Application Monitors** menus. You can use also SMIT fast-path **smit cm_cfg_appmon**.

Process application monitoring

The process application monitoring facility uses a build-in monitoring capability, provided by RSCT and does not require any custom script. It detects only the

application process termination and is not able to detect any other malfunction of the application that could appear when application is running.

When HACMP finds that some application process are terminated, it tries to restart the application on the current node until a specified retry count is exhausted.

You can add a new process application monitor by running **smit hacmp > Extended Resource Configuration > Extended Resource Configuration > HACMP Extended Resources Configuration > Configure HACMP Applications > Configure HACMP Application Monitoring > Configure Process Application Monitors > Add a Process Application Monitor**, or using SMIT fast-path **smit cm_cfg_process_appmon.** The Figure 8-32 shows the SMIT screen with field entries for configuring our test process application monitor.

```
 Add a Process Application Monitor

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                  [Entry Fields]
* Monitor Name                                   [APP1_monitor]
* Application Server(s) to Monitor                APP1                +
* Monitor Mode                                   [Long-running monito>+
* Processes to Monitor                           [app1d app1testd]
* Process Owner                                  [root]
  Instance Count                                 [1]                 #
* Stabilization Interval                         [120]               #
* Restart Count                                  [5]                 #
  Restart Interval                               [600]               #
* Action on Application Failure                  [notify]            +
Notify Method                                  [/usr/local/App1Mon.sh>
  Cleanup Method                                 [/usr/local/App1Stop]
  Restart Method                                 [/usr/local/App1Start]

F1=Help          F2=Refresh         F3=Cancel          F4=List
F5=Reset         F6=Command         F7=Edit            F8=Image
F9=Shell         F10=Exit           Enter=Do
```

Figure 8-32 Adding process application monitor SMIT screen

We defined application monitored *APP1_monitor* for *APP1* application server. We used default monitor mode: *Long-running monitoring*, it means that In this mode, the application monitor periodically checks that the application server is running. The checking starts after the specified *Stabilization interval* has passed. The alternatives are *Startup Monitoring* and *both*. By selecting *Startup*

Monitoring option, the application monitor, checks that the application server has successfully started within the specified stabilization interval. We set *Stabilization interval* to the value of 120 s.

We defined the processes app1d and app1testd, owned by root, that will be monitored with this application monitor.

We can define the two different behavior of monitor in case of failure is still persistent after all retries, as defined in *Restart Interval* field. You can set this with *Action on Application Failure* field. Choices are *notify* and f*allover*. If notify is selected, no further action is taken after running the notify method. If fallover is selected, the resource group containing the monitored application moves to the another node in the cluster.

Notification method parameter defines the script, that executes each time when application restarts, fails completely, or falls over to the next node in the cluster. Configuring this method is strongly recommended.

Cleanup Method and *Restart Method* define the scripts for stopping and starting the application after failure is detected. The default start and stop scripts are used as those, defined in the application server configuration.

Custom application monitoring

Custom application monitor offers another possibility to monitor the application availability by using custom scripts, which could simulate client access to the services, provided by the application. It uses build-in monitoring facility monitoring capability, provided by RSCT and does require customized script. Based on the exit code of this script monitor establish if application is available or not. If scrip exit with return code 0, than application is available. Any other return code means that application is not available.

You can add a new custom application monitor using SMIT, by running **smit hacmp > Extended Resource Configuration > Extended Resource Configuration > HACMP Extended Resources Configuration > Configure HACMP Applications > Configure HACMP Application Monitoring > Configure Custom Application Monitors > Add a Custom Application Monitor**, or using SMIT fast-path **smit cm_cfg_custom_appmon**. The SMIT screen and its entries for adding this method into cluster configuration are similar to the process application monitor add SMIT screen, as showed in Table 8-32 on page 422.

The only different fields in configuring custom application monitors SMIT menu are as follows:

- Monitor Method. It defines the full path name for the script, that defines a method to check the application status. If the application is a database this script could connect to database and run a SQL select sentence over a

specific table in database. If the given result of the SQL select sentence is correct, it means that database works normaly.

- Monitor Interval. It defines the interval (in seconds), the monitor method will be run periodically.
- Hung Monitor Signal. It defines the signal that is sent to stop the Monitor Method if it doesn't return within *Monitor Interval* seconds. The default is SIGKILL(9).

Suspend/Resume Application Monitoring

After you configure the application monitor, you have to activate it. You can do this action with SMIT Resume application menu, running **smit cl_admin > HACMP Resource Group and Application Management > Suspend/Resume Application Monitoring > Resume Application Monitoring** and than you select the application server, connected with the monitor, you want to activate. The Example 8-19 shows the output, after we successfully resumed the application monitor on the application server APP1 in our test cluster.

Example 8-19

```
Jul  6 2005 18:00:17 cl_RMupdate: Completed request to resume monitor(s) for
applic ation APP1.
Jul  6 2005 18:00:17 cl_RMupdate: The following monitor(s) are in use for
applicati on APP1:
test
```

You could suspend or resume the application monitor any time. This action does not affect to the application server availability, but affects on the statistic result, shown by the application availability analysis tool. You can find more information about this tool in the 8.7.8, “Measuring an application availability” on page 424.

8.7.8 Measuring an application availability

You can use the application availability analysis tool for measuring the amount of time, when your high available applications are generally available. The HACMP software collects and logs the following information in time-stamped format:

- An application starts, stops, or fails.
- A node failes, shutdowns, or comes online, as well as cluster services are started or shut downed.
- A resource group is taken offline or moved.
- Application monitoring is suspended or resumed.

According the information, collected by the application availability analyze tool, you can select a time for measurement period and the tool displays uptime and downtime statistics for a specific application during that period. Using SMIT you could display:

- The percentage of uptime.
- The amount of uptime.
- The longest period of uptime.
- The percentage of downtime.
- The amount of downtime.
- The longest period of downtime.
- The percentage of time application monitoring was suspended.

The application availability analysis tool reports application availability from the HACMP cluster perspective. It can analyze only those applications which have been properly configured in cluster configuration.

This tool is showing only the statistics, that reflects the availability of the HACMP application server, resource group, and the application monitor (if configured). It could not measure any internal failure in application, that could be detected from end-user, as long it is not detected by application monitor.

Using the Application Availability Analysis tool

You could use application availability analysis tool immediately after you define the application servers, since the tool does not need any additional customization and it automatically collect statistics for all application servers.

You can display the specific application statistic, generated with Application Availability Analysis tool with SMIT menus using **`smit hacmp`** > **System Management (C-SPOC)** > **Resource Group and Application Management** > **Application Availability Analysis**. The Figure 8-33 on page 426 shows the SMIT screen display, for the application availability analyses tool in our test cluster environment.

smit cl_app_AAA.dialog

```
                    Application Availability Analysis

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
* Select an Application                               [APP1]               +
* Begin analysis on YEAR (1970-2038)                  [2005]               #
* MONTH (01-12)                                       [07]                 #
* DAY (1-31)                                          [06]                 #
* Begin analysis at HOUR (00-23)                      [16]                 #
* MINUTES (00-59)                                     [22]                 #
* SECONDS (00-59)                                     [00]                 #
* End analysis on YEAR (1970-2038)                    [2005]               #
* MONTH (01-12)                                       [07]                 #
* DAY (1-31)                                          [06]                 #
* End analysis at HOUR (00-23)                        [17]                 #
* MINUTES (00-59)                                     [42]                 #
* SECONDS (00-59)                                     [00]                 #

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 8-33 Adding Application Availability Analysis SMIT screen

In the SMIT menu of application availability analyses tool you just need to enter selected application server, enter start and stop time for statistics and run the tool. In the Example 8-20 you can see the application availability analyses tool output from our test cluster.

Example 8-20 Application availability analysis tool output

```
Analysis begins:                  Wednesday, 06-July-2005, 16:20
Analysis ends:                    Wednesday, 06-July-2005, 17:42
Application analyzed:             APP1

Total time:                       0 days, 1 hours, 22 minutes, 0 seconds

Uptime:
        Amount:                   0 days, 1 hours, 16 minutes, 51 seconds
        Percentage:               93.72%
        Longest period:           0 days, 1 hours, 10 minutes, 35 seconds

Downtime:
        Amount:                   0 days, 0 hours, 5 minutes, 9 seconds
        Percentage:               6.28%
        Longest period:           0 days, 0 hours, 5 minutes, 9 seconds
```

```
Log records terminated before the specified ending time was reached.

Application monitoring was suspended for 75.87% of the time period analyzed.

Application monitoring state was manually changed during the
time period analyzed.

Cluster services were manually restarted during the time period analyzed.
```

9

Cluster security

This chapter describes the HACMP security features and how you can setup your cluster to be more secure. The main topics are:

- Cluster and clcomd daemon security
- User and password management
- Encrypted cluster messaging
- Secure remote command execution
- WebSMIT security
- HACMP and firewalls
- RSCT Security

9.1 Cluster security and clcomd daemon

The Cluster Communication Daemon is a robust, secure transport layer for HACMP. Clcomd manages almost all inter-cluster communication, such as CSPOC, cluster verification and synchronization, file collections. Clcomd also enhance the cluster performance: it has a cache for HACMP ODM files for faster delivery, and can reuses it's existing socket connections.

The cluster manager daemon and the heart-beating is based on RSCT (see 9.6, "RSCT security" on page 448), while clinfo uses SNMP protocol.

There are two security modes for connection authentication in HACMP:

- *Standard:* the incoming connection is checked against an access list and only minimum required access is granted. In this chapter we always talk about the standard cluster security mode.
- *Kerberos:* all cluster communication is secured by using PSSP kerberos infrastructure. This security mode works only on SP system with PSSP software installed.

Since HACMP Version 5.1 there is no more need for .rhosts file. HACMP security based on connection authentication with the "least privilege" principle. The Cluster Communication daemon uses an internal access list to authorize other nodes to run command remotely. When a remote execution request arrives to a node, Clcomd checks the connection incoming IP address against the IP addresses found in the following locations:

1. HACMPnode ODM class
2. HACMPadapter ODM class
3. /usr/es/sbin/cluster/etc/rhosts file.

The Cluster Communication daemon use the principle of "least privileged". When a command is executed remotely only the minimum required privileges granted. This ensures that only trusted HACMP commands can be run remotely and as user *nobody* whenever possible. The cluster utilities are divided into two groups:

- trusted commands that are allowed to run as *root*,
- other commands that run as *nobody*.

Restriction: You cannot use Clcomd based authentication for your own scripts (application start and stop, custom events, etc.), you should still rely on rsh or SSH.

9.1.1 The /usr/es/sbin/cluster/etc/rhosts file

The /usr/es/sbin/cluster/etc/rhosts file should contain the list of the all IP addresses of all cluster nodes for enhanced security. This file is automatically gets the node connection information (node's base addresses) from HACMP during discovery and cluster verification and synchronization. Also you can manually edit /usr/es/sbin/cluster/etc/rhosts file to include nodes' IP addresses.

Important: The /usr/es/sbin/cluster/etc/rhosts file should have the following permissions:

- owner: root
- group system
- permissions: 0600

Initial cluster setup

During initial cluster setup the /usr/es/sbin/cluster/etc/rhosts file is empty. Normally HACMP will put the peer nodes' base addresses there during the first discovery and cluster synchronization. For a secure initial configuration we suggest that you add your cluster's IP addresses to /usr/es/sbin/cluster/etc/rhosts file on all nodes before you start the HACMP configuration.

Attention: During the initial setup, when the /usr/es/sbin/cluster/etc/rhosts file is still empty on a HACMP node it's possible that an unwelcome host connects first and puts its IP address to the rhosts file. In this case the other peer nodes cannot connect until the /usr/es/sbin/cluster/etc/rhosts file manually corrected.

9.1.2 Disabling Cluster Communication daemon

You can disable clcomd for higher security. Please consider that without clcomd the following functions cannot work:

- HACMP verification and synchronization
- CSPOC, including LVM, user and password management
- File collections
- Message authentication and encryption

You can disable Cluster Communication daemon by stopping clcomd: **`stopsrc -s clcomdES`**. To restart clcomd: **`startsrc -s clcomdES`**.

9.1.3 Additional cluster security features

The HACMP ODM files are stored in the /etc/es/objrepos directory. In order to improve security their owner is *root* and group ID is *hacmp*. Their permission is

0640, except HACMPdisksubsystem which is 0600. All cluster utilities intended for public use have hacmp setgid turned on so they can read the HACMP ODM files.The hacmp group is created during HACMP installation, if it's not already there.

9.2 Using encrypted inter-node communication

HACMP supports encrypted messaging between the cluster hosts. All Cluster Manager, Cluster Communication daemon and RSCT traffic can be encrypted.

The encryption is based on a symmetric key scheme, provided by Cluster Security Services (CtSec). CtSec is part of RSCT. See also 9.6, "RSCT security" on page 448.

The following RSCT encryption modules can be used:

- Data Encryption Standard (md5_des)
- Triple DES (md5_3des)
- Advanced Encryption Standard (md5_aes)

The encryption put an additional load on the CPUs, but the fail-over time is not affected.

9.2.1 Encryption key management

The RSCT encryption modules use symmetric keys. It means that all cluster node has the same key for encryption/decryption. On each node the security keys are located in /usr/es/sbin/cluster/es directory. The file name reflects the selected encryption method:

- key_md5_des
- key_md5_3des
- key_md5_aes

You should generate the key, when you first time set up the message encryption, when you modify the cluster security configuration, or when your company's security rules dictate. One key can be used as long as you like.

HACMP provides the mechanism to distribute the keys among the nodes:

- Automated key distribution: the new key is distributed among the cluster hosts automatically after you have generated it. This method is convenient, but less secure because the keys are copied over the network.

- Manual key distribution: you should manually copy the key file to all node. You can use SFTP or SCP, but the most secure distribution method is to have a floppy disk to distribute the keys. We suggest that you use this method.

Attention: Please check that the key-file's owner is root, the group ID is system and the permission is 0400. If somebody can copy your key or intercepts over the network, the cluster security will be in serious danger.

9.2.2 Set up message encryption

The main configuration steps:

1. Installing the preferred RSCT encryption module.
2. Enable automatic distribution of the keys (automatic key distribution only).
3. Enable the message authentication and encryption.
4. Generate and distribute the keys.
5. Activate the key.
6. Synchronize the cluster.
7. Disable automatic distribution of the keys (automatic key distribution only).

For security reason we suggest that you use manual key distribution.

You should run the commands always from the same node except step 2. and step 7. which required to execute on all nodes.

Important:

- Please, ensure that the authentication and encryption settings are consistent across the cluster. All nodes should have to use identical key files. Otherwise the HACMP cannot communicate with the nodes.
- Please do not perform any other HACMP configuration task while setting up the encrypted cluster messaging.
- Be sure, that your cluster is in good working order, and it recently synchronized and verified, and the nodes can communicate to each other.

1. Prerequisites.

   You should install your preferred encryption library from the AIX 5L Expansion Pack CD. Install the following filesets using **`smit install_latest`**:

   – **rsct.crypt.des** for using md5_des,

   – **rsct.crypt.3des** for using md5_3des,

– **rsct.crypt.aes256** for using md5_aes.

If your cluster is already running you have to restart clcomd now:

```
stopsrc -s clcomdes
startsrc -s clcomdes
```

2. Enable automatic distribution of the keys on all node.

   Perform this step only if you like to use automatic key distribution.

   a. Go to SMIT **HACMP Cluster Security**: start **smit**.

   – Select **Communications Applications and Services**.
   – Select **HACMP for AIX**.
   – Select **System Management (C-SPOC)**.
   – Select **HACMP Security and Users Management**.
   – Select **HACMP Cluster Security**.

   Or, use **smit cm_config_security** fast-path.

   b. Select **Configure Message Authentication Mode and Key Management**.

   c. Select **Enable/Disable Automatic Key Distribution**.

   d. Change **Enable/Disable Key Distribution** to **Enabled**. See Figure 9-1 on page 435.

   e. Press Enter to confirm.

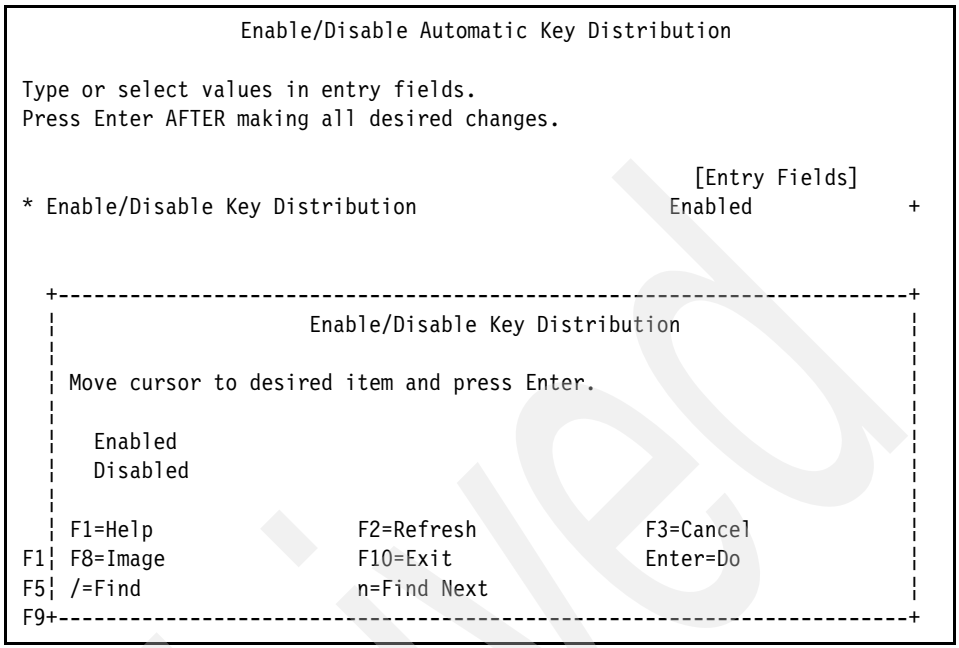

Figure 9-1 Enable automatic key distribution

Repeat this steps on all nodes.

3. Enable or change the message authentication and encryption method:
   a. Go to SMIT **HACMP Cluster Security**: start **`smit cm_config_security`**.
   b. Select **Configure Message Authentication Mode and Key Management**.
   c. Select **Configure Message Authentication Mode**.
   d. Press F4 and select the **Message Authentication Mode**, what you would like to use (see Figure 9-2 on page 436):
      - **md5_des**
      - **md5_3des**
      - **md5_aes**
      - **None**: Neither message authentication nor encryption is used.
   e. Set **Enable Encryption** to **Yes**.
   f. Press Enter.

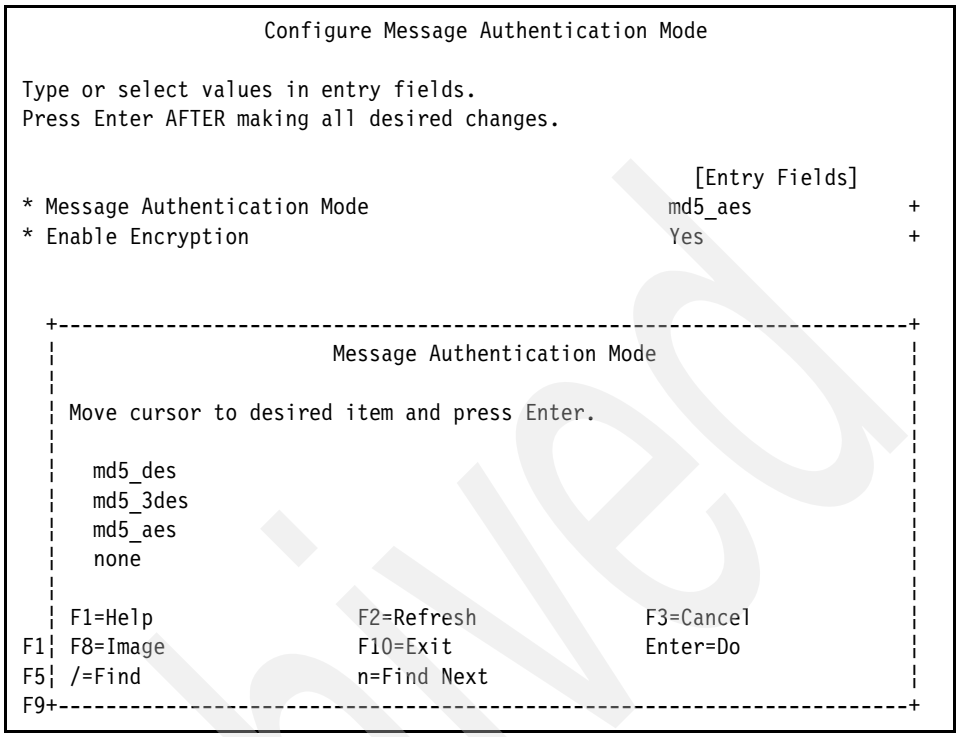

Figure 9-2 Configure message authentication mode

4. Generate and distribute the keys
   a. Go to SMIT **HACMP Cluster Security**: start **`smit cm_config_security`**.
   b. Select **Configure Message Authentication Mode and Key Management**.
   c. Select **Generate/Distribute a Key**.
   d. Press F4 and select the **Type of Key to Generate**. This should be the same as what you selected in Step 3., “Enable the message authentication and encryption.” on page 433. See SMIT screenshoot on Figure 9-3 on page 437.
   e. Select **Distribute a Key**:
      - **Yes**: if you use automatic key distribution.
      - **No**: if you prefer manual key distribution.
   f. Press Enter.

```
                        Generate/Distribute a Key

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Type of Key to Generate                            md5_aes +
* Distribute a Key                                   No +




F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 9-3 Generate a key

If you selected automatic key distribution, then proceed to Step 5., “Activate the key.” on page 433.

Now copy the key file from this node to the other hosts. You can use any transfer method what you like, but we suggest that you use one of the following way:

– SCP: secure, encrypted remote copy utility. Example:

```
scp /usr/es/sbin/cluster/etc/key_md5_aes \
node2:/usr/es/sbin/cluster/etc/key_md5_aes
```

– Floppy: copy the corresponding key file from /usr/es/sbin/cluster/etc directory to a floppy disk. Then copy the file from the floppy to all node.

If you already have a key file just replace that with the new one.

Please take care of the key file, do not send it unencrypted over a network (e.g., don’t use ftp or rcp) and be sure to have its permission set to 0400.

5. Activate the key
   a. Go to SMIT **HACMP Cluster Security**: start **`smit cm_config_security`**.
   b. Select **Configure Message Authentication Mode and Key Management**.
   c. Select **Activate the new key on all HACMP cluster nodes**.
   d. Press Enter again to confirm.

Important: Do not activate the new key until all the cluster node has the same key file installed.

6. Synchronize the cluster. If you encounter any error related to cluster communication, then disable both message authentication and encryption and start over this procedure.
7. Disable automatic distribution of the keys on all node

   Perform this step only if you like to use automatic key distribution.

   a. Go to SMIT **HACMP Cluster Security**: start **`smit cm_config_security`**.
   b. Select **Configure Message Authentication Mode and Key Management**.
   c. Select **Enable/Disable Automatic Key Distribution**.
   d. Change **Enable/Disable Key Distribution** to **Disabled**.
   e. Press Enter to confirm.

   Repeat this step on all node.

Important: Always disable automatic distribution of keys after you successfully generated and distributed the new keys. Otherwise the cluster security will be compromised.

Changing the authentication key

You can use a key as long as you like, but longer you use the same key, the bigger the chance that somebody can intercept or it. For superb security we suggest that you change it regularly, like once a month.

1. Enable automatic distribution of the keys on all node. Perform this step only if you use automatic key distribution.

   a. Go to SMIT **HACMP Cluster Security**: start **`smit cm_config_security`**.
   b. Select **Configure Message Authentication Mode and Key Management**.
   c. Select **Enable/Disable Automatic Key Distribution**.
   d. Change **Enable/Disable Key Distribution** to **Enabled**.
   e. Press Enter to confirm.

   Repeat this step on all node.
2. Generate and distribute the keys

   a. Go to SMIT **HACMP Cluster Security**: start **`smit cm_config_security`**.
   b. Select **Configure Message Authentication Mode and Key Management**.
   c. Select **Generate/Distribute a Key**.

d. Press F4 and select the **Type of Key to Generate**. This should be the same as your message authentication mode.

e. Select **Distribute a Key**:

- **Yes**: if you use automatic key distribution.
- **No**: if you prefer manual key distribution.

f. Press Enter

g. If you use manual key distribution, then copy the key file from this node to the other hosts. Replace the old files.

3. Activate the key

a. Go to SMIT **HACMP Cluster Security**: start `smit cm_config_security`.

b. Select **Configure Message Authentication Mode and Key Management**.

c. Select **Activate the new key on all HACMP cluster nodes**.

d. Press Enter again to confirm.

4. Disable automatic distribution of the keys on all nodes. Perform this step only if you use automatic key distribution.

a. Go to SMIT **HACMP Cluster Security**: start `smit cm_config_security`.

b. Select **Configure Message Authentication Mode and Key Management**.

c. Select **Enable/Disable Automatic Key Distribution**.

d. Change **Enable/Disable Key Distribution** to **Disabled**.

e. Press Enter to confirm.

Repeat this step on all node.

9.2.3 Troubleshooting message authentication and encryption

If you encounter any cluster communication error (e.g., cluster verification fails or CSPOC cannot communicate with other nodes), then turn of both message authentication and message encryption:

1. Go to SMIT **HACMP Cluster Security**: start `smit cm_config_security`.
2. Select **Configure Message Authentication Mode and Key Management**.
3. Select **Configure Message Authentication Mode**.
4. Set **Message Authentication Mode** to **None**.
5. Set **Enable Encryption** to **No**.
6. Press Enter.

7. Synchronize the cluster.

9.2.4 Checking the current message authentication settings

You can check the current message authentication and encryption settings:

1. Start **smit hacmp**.
2. Select **Extended Configuration**.
3. Select **Extended Topology Configuration**.
4. Select **Show HACMP Topology**.
5. Select **Show Cluster Definition**. Press Enter to see the cluster definition settings. See SMIT screenshoot on Figure 9-4.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

Cluster Name: two2four
Cluster Connection Authentication Mode: Standard
Cluster Message Authentication Mode: md5_aes
Cluster Message Encryption: Enabled
Use Persistent Labels for Communication: No



F1=Help          F2=Refresh         F3=Cancel          F6=Command
F8=Image         F9=Shell           F10=Exit           /=Find
n=Find Next
```

Figure 9-4 Checking the message authentication settings

9.3 Secure remote command execution in a HACMP

Note: Although not mandatory in an HACMP cluster, secure shell is a very popular method for securing remote command execution in today's networking environment.

The application start and stop scripts, customized cluster events and other scripts may require to run commands on remote nodes. The rsh can be still used for this purpose, but it's not secure because of its .rhosts file based

authentication. We strongly suggest that you use Secure Shell (SSH). Also DLPAR operation requires SSH.

SSH and Secure Socket Layer (SSL) together provide authentication, privacy and data integrity. SSH authentication based on public - private key infrastructure, while SSL encrypts network traffic.

SSH has the following utilities to replace the classic r-commands:

ssh Secure remote shell, similar to **rsh** or **rlogin**.

scp Secure remote copy, similar **rcp**.

sftp Encrypted file transfer utility, similar to **ftp**.

9.3.1 Installing SSH

IBM conveniently supply the SSH and all of its components with AIX. Here you can see the required components:

rpm.rte	Support for rpm packages, automatically installed with the base AIX
perl	Automatically installed with the base AIX
openssl.rpm	Open Source Secure Socket Layer support (AIX Toolbox for Linux Applications CD)
openssh.base.client	Open Source Secure Shell client commands (AIX Expansion Pack)
openssh.base.server	Open Source Secure Shell server (AIX Expansion Pack)
openssh.msg.*	Message catalog for SSH (AIX Expansion Pack)
openssh.man.*	Man pages for Secure Shell (AIX Expansion Pack)
openssh.license	Open Source License for SSH (AIX Expansion Pack)
prngd.rpm	Random number generator, required only on AIX 5.1 and 5.2. (AIX Toolbox for Linux Applications CD)

Alternatively you can download the latest packages from the internet:

- OpenSSL from IBM AIX Toolbox for Linux Applications site (requires a few minutes registration procedure):

  http://www6.software.ibm.com/dl/aixtbx/aixtbx-p

- OpenSSH for AIX:

  http://www.sourceforge.net/projects/openssh-aix

In our test we used openssl-0.9.6m-1 and openssh.3.8.0.53.

The installation steps:

1. Check that rpm.rte is installed: **`lslpp -l rpm.rte`**.
2. Check that Perl is installed: **`lslpp -l perl.rte`**.
3. If you have AIX 5.1 or 5.2 install Prngd from AIX Toolbox for Linux Applications CD with SMIT.
4. Install OpenSSL: you can use **`smit install_latest`** to install it from AIX Toolbox for Linux Applications CD. Otherwise use rpm command to install it:

   ```
   rpm -i openssl-0.9.6m-1.aix5.1.ppc.rpm
   ```

5. Use **`smit install_latest`** to install openssh.base, the license file and the corresponding language filesets.
6. Start SSH daemon: **`startsrc -s sshd`**.

9.3.2 Setting up SSH for passwordless remote command execution

Some scripts may requires to use SSH and SCP in passwordless configuration. In this case SSH use the private - public key pairs for authentication. The private key is stored in the local host, while the public key is sent to the remote nodes and added to their authorized keys database.

When a SSH command is executed (e.g., **`ssh node2 date`**) the communication is signed with the originating node private or secret key. This signature can be decrypted with the originating node's public key only. If the receiving node can decrypt the signature successfully, then sender host is the one it claims to be.

Perform the following steps on each cluster node to set up passwordless SSH remote command execution:

1. Login with the required user identity.
2. Generate your authentication key pair:

   ```
   ssh-keygen -t rsa -f ~/.ssh/node1
   ```

   Press Enter for the passphrase (no password).

   This command generates two files:

   – ~/.ssh/node1: this is your secret key

   – ~/.ssh/node1.pub: this is your public key

3. Rename your secret key to *identity*:

   ```
   mv ~/.ssh/node1 ~/.ssh/identity
   ```

4. Add the public key to your *authorized_keys* file on the local node, so the SSH will work for the localhost:

   ```
   cat ~/.ssh/node1.pub >> ~/.ssh/authorized_keys
   ```

5. Copy your public key to all other hosts:

   ```
   scp ~/.ssh/node1.pub nodeX:~/.ssh/node1.pub
   ```

   Repeat this command for each node in the cluster.

6. Add node1's public key to the *authorized_keys* file on the remote hosts:

   ```
   ssh nodeX "cat ~/.ssh/node1.pub >> ~/.ssh/authorized_keys"
   ```

   Repeat this command for each node in the cluster.

7. Repeat steps 1 to 6 on all hosts.

Now SSH should work between any of your node without asking for password.

Important: Please take care of your private and public key files: they are the keys to your system. If somebody unauthorized gets that files he may able to login to your system.

You can find more information about OpenSSL and OpenSSH on the internet:

- *OpenSSL project* Web site:

  http://www.openssl.org

- *OpenSSH project* page:

  http://www.openssh.org

- *OpenSSH on AIX*:

  http://www.sourceforge.net/projects/openssh-aix

9.4 WebSmit security

WebSMIT is a very nice tool to configure and monitor your cluster. However WebSMIT access HACMP files and runs cluster commands as a root user. this can be a potential security exploit but with careful security planning and implementation the risk can be totally eliminated. This chapter describes the security requirements and mechanisms used in WebSMIT environments.

The WebSMIT can be configured to have maximum security:

- Encrypted communication between client and server using SSL.
- SSL certificate for authentication.
- Web server and AIX authentication.
- Only selected AIX users can access WebSMIT.
- Only a selected SMIT panels accessible through WebSMIT.

WebSMIT requires SSL and an SSL certificate for authentication. A self-signed certification is generated during Apache installation, this can be used by WebSMIT.

9.4.1 WebSMIT security settings

The WebSMIT security settings stored in **/usr/es/sbin/cluster/wsm/wsm_smit.conf** file. By default the file contains the highest security settings. Here we describe the security options.

Attention: If you change any of the WebSMIT settings, then you have to restart the HTTP server.

Authorized port

The `AUTHORIZED_PORT` is the TCP port is used for WebSMIT communication. The default is 42267. We suggest you to change this setting to a different number to prevent that somebody can easily find WebSMIT knowing this port number. If you define this port number here, then WebSMIT will use that protocol (http or https) which is associated with this port in the HTTP server configuration file. For example, if you define port 42267 as an SSL port in httpd.conf then your browser will use encryption, and you should use “https://” in the URL.

Allow only secure http

The `REDIRECT_TO_HTTPS=1` setting allows only secure https communication, so all data is transported encrypted through the network. If you defined `AUTHORIZED_PORT` then this variable is not used. Otherwise your browser will be redirected to an SSL port regardless of the original URL. See Table 9-1 to see the relation between `AUTHORIZED_PORT` and `REDIRECT_TO_HTTPS` settings.

Table 9-1 Relation between AUTHORIZED_PORT and REDIRECT_HTTPS settings

	AUTHORIZED_PORT set	**AUTHORIZED_PORT not set**
REDIRECT_TO_HTTPS=0	Port security settings from httpd.conf will be used	Insecure HTTP protocol used, no encryption, no SSL.
REDIRECT_TO_HTTPS=1	Port security settings from httpd.conf will be used	HTTPS protocol used with SSL and encryption

Tip: We suggest that you set up an SSL TCP port in your httpd.conf file and use that port as an authorized port for WebSMIT.

AIX authentication

WebSMIT can use AIX authentication in addition to the Web server authentication by setting `REQUIRE_AUTHENTICATION=1`. In this case WebSMIT asks for an AIX user ID and password. Because AIX authentication mechanisms are in use, login failures can cause an account to be locked.

Tip: Set up both Web server authentication and AIX authentication for enhanced WebSMIT security.

Accepted users

The `ACCEPTED_USERS` variable contains a list of AIX user ID, who can get access to WebSMIT. All users who stated here will have **root** level of authority in WebSMIT. The default is `ACCEPTED_USERS="root"`.

Tip: We strongly recommend that you create a separate AIX user for WebSMIT access.

Session Timeout

The `SESSION_TIMEOUT` variable defines the session time out value, when the WebSMIT user have to re-login again regardless that he is active or not. The default is 20 minutes.

Web server user ID

The `REQUIRED_WEBSERVER_UID` variable is the user ID used by WebSMIT cgi-bin scripts. This should be the same ID what you use in your Web server. Also ensure that this user ID does not have AIX login privilege, thus nobody can “su” to it. The default is “nobody”.

9.4.2 Allow or deny specific SMIT panels in WebSMIT

You can configure which SMIT panels can be accessed through WebSMIT. SMIT panel name is the SMIT fast path name for a given panel, you can get this name by pressing F8 in SMIT. For example the panel name for SMIT Extended Topology Configuration is “`cm_extended_topology_config_menu_dmn`”. See Figure 9-5 on page 446.

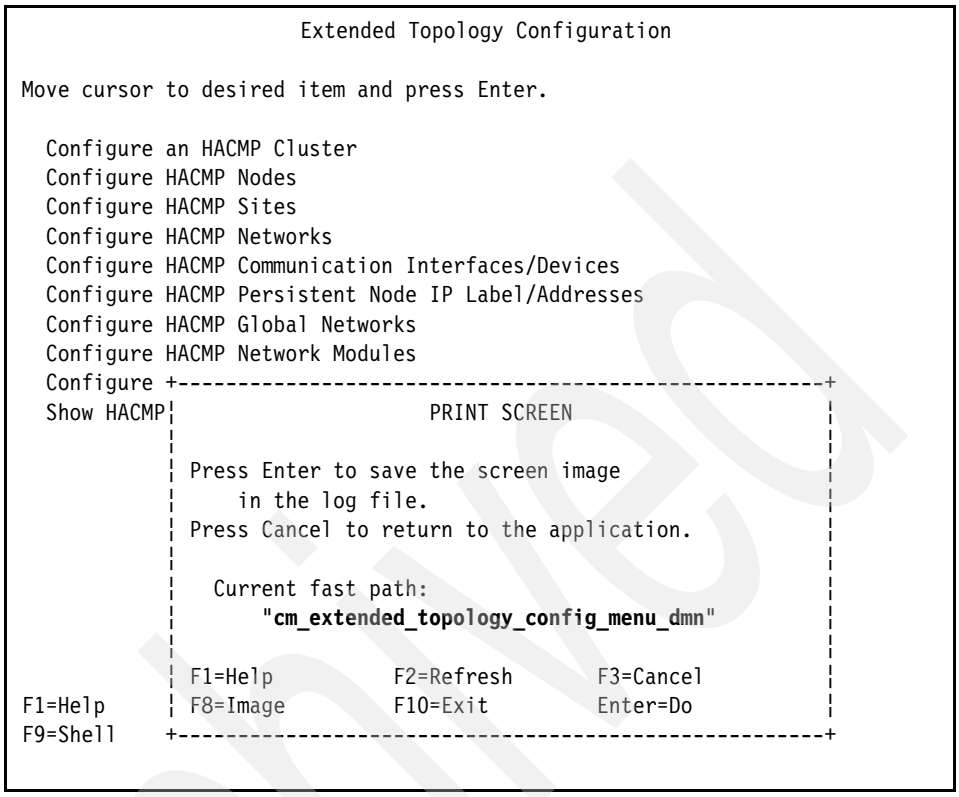

Figure 9-5 SMIT Extended Topology Configuration fast path

The **/usr/es/sbin/cluster/wsm/wsm_smit.allow** file contains a list of SMIT panels which are only allowed to access from WebSMIT. All other panels will be rejected.

In the **/usr/es/sbin/cluster/wsm/wsm_smit.deny** file you can list that SMIT panels that are not accessible from WebSMIT. If a panel is listed in both allow and deny file then the deny setting has the precedence

When a SMIT panel listed in **/usr/es/sbin/cluster/wsm/wsm_smit.redirect**, WebSMIT will redirect the page to a given URL. This file already contains some panels, please do not modify them.

Tip: After finished your cluster configuration deny all SMIT panels except the HACMP cluster status page. See Example 9-1.

Example 9-1 Sample /usr/es/sbin/cluster/wsm/wsm_smit.allow file

```
################################################
# webSMIT .allow file                          #
################################################
cm_hacmp_main_menu_dmn
```

9.4.3 WebSMIT logs

WebSMIT log all of its operation in a similar fashion like SMIT. The **wsm_smit.log** file is located in ../log directory, relative to the cgi-bin scripts. For example, if you use the cgi-bin scripts from the default /usr/es/sbin/cluster/wsm/cgi-bin directory, then log file is in /usr/es/sbin/cluster/wsm/log. Also the latest WebSMIT command can be found in **wsm_smit.script** file here.

HACMP does not manage this log files, they grow indefinitely just like the normal smit.log file. The **snap -e** command can collects WebSMIT log files only if they are in the default /usr/es/sbin/cluster/wsm/log directory.

9.5 HACMP and firewalls

There are some consideration for putting a HACMP cluster behind a firewall:

- HACMP doesn't require any open port on the firewall, there is no outside traffic from Clcomd, RSCT or Cluster Manager. You only need open ports for your application and system management (e.g., SSH).
- Ensure that all service IP can communicate with the outside network regardless where they are bounded. Take in consideration that during a network failure a service interface will move from one base adapter to the other one. In case of a takeover the failing node's service address moves to other node.
- Don't put a firewall between the nodes. In a HACMP/XD cluster your nodes may connect through a public network. In this case use Virtual Private Networks or other solution that is transparent for the Cluster Communication Daemon.
- If you use netmon.cf file for enhanced network failure detection be sure that the IP address listed in this file can be reached (ping) through your firewall.
- If you have Clinfo clients coming through a firewall, then you should open the clinfo_client port: 6174/tcp.
- Be sure that your firewall solution is redundant, otherwise the firewall is a single point of failure.

9.6 RSCT security

In this chapter, we describe the terminology and concepts used in RSCT security. We also describe the RSCT security architecture and mechanisms used by HACMP. Knowledge of RSCT and its base components (such as RMC, Resource managers, and so forth) is assumed. It is important to understand that most of the components and mechanisms explained in this chapter do not need to be configured, the RSCT security layer work out of the box. Also some function and files are configured only when HACMP message authentication and encryption is set up (see 9.2, “Using encrypted inter-node communication” on page 432). Some of the described functions is not used by HACMP but they are integral part of RSCT security.

9.6.1 RSCT and HACMP

HACMP uses RSCT as an underlying infrastructure layer. HACMP nodes contact its peer nodes using RSCT components to ensure they are still alive or to request access to their resources. The Topology Services and Group Services implements the HACMP heartbeat using an RSCT peer domain. The RMC subsystem is used for:

- Custom events
- Application monitoring
- Dynamic node priority
- Exporting network status for use by Oracle RAC.

Figure 9-6 on page 449 shows how HACMP and RSCT components is related to each other.

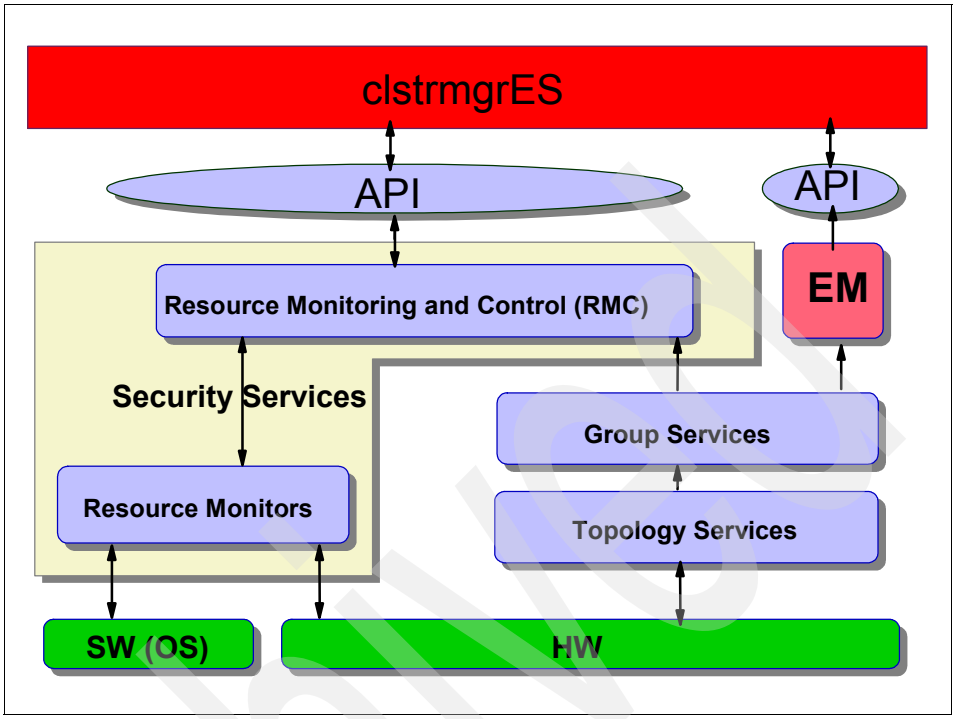

Figure 9-6 RSCT and HACMP

Every time an HACMP component, for example, the cluster manager, of the one server sends a functional request to a RMC resource, either local or remote, the RSCT subsystem is called and the requests are sent to the node where the resource is located. See Figure 9-7 on page 450.

Because remote connections are simply TCP/IP socket connections, they must be secured in order to ensure that both the requestor and the server node is the one the cluster expects it to be. This is where the Cluster Security Services (CtSec) comes into play.

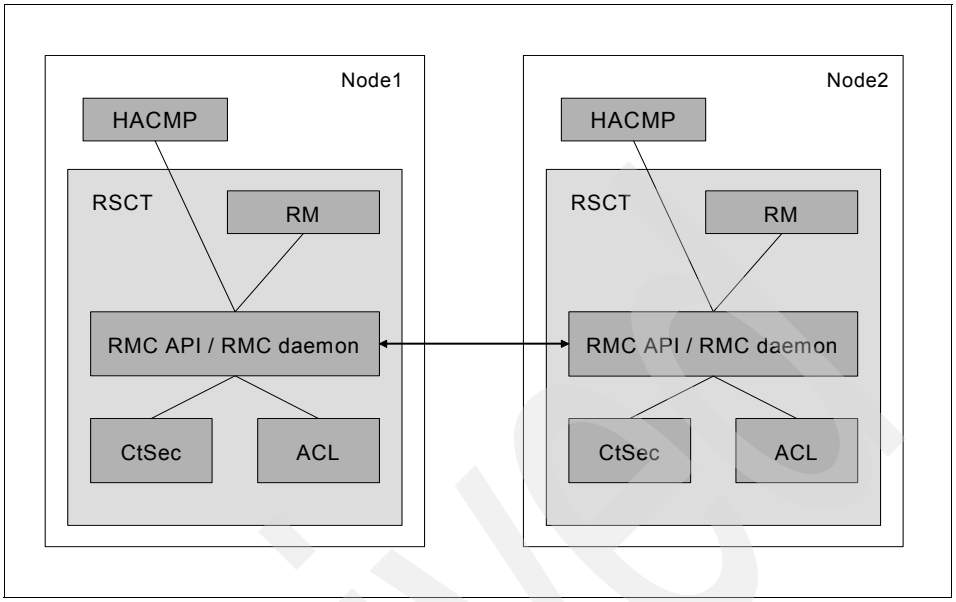

Figure 9-7 Basic cluster communication overview

The communication between cluster members is a client/server communication. In the following sections, *client* is used for an application, for example, a local resource manager or a HACMP function that is using the RMC client API.

These client applications are linked against the RMC and CtSec shared libraries. If those clients request access to resources on a remote node, the RMC daemon on that remote node will be the *server* for those requests.

9.6.2 Cluster Security Services (CtSec) overview

Cluster Security Services (CtSec) are integrated into the RSCT subsystem and are used by RMC to determine the identity of a client from a node. This authentication process results in a security context that is used by RMC for the communication between the participating nodes to fulfill the client's request.

Note: The security context is at a client/server level, not at a node level.

To create this security context, CtSec uses credentials for the authentication. Those credentials are used to determine the authenticity of a node and the client application. To access resources on a remote node, both nodes send and compare the credentials during the authentication process.

This process allows the following:

- A client to present information about itself that cannot be imitated by others
- A server to clearly identify the client by its given credentials
- A client to be sure it is retrieving data from the correct server

The credential-based authentication involves other components to create and identify credentials. This component-based architecture, shown in Figure 9-8, also allows future extensions to the security layer in RSCT.

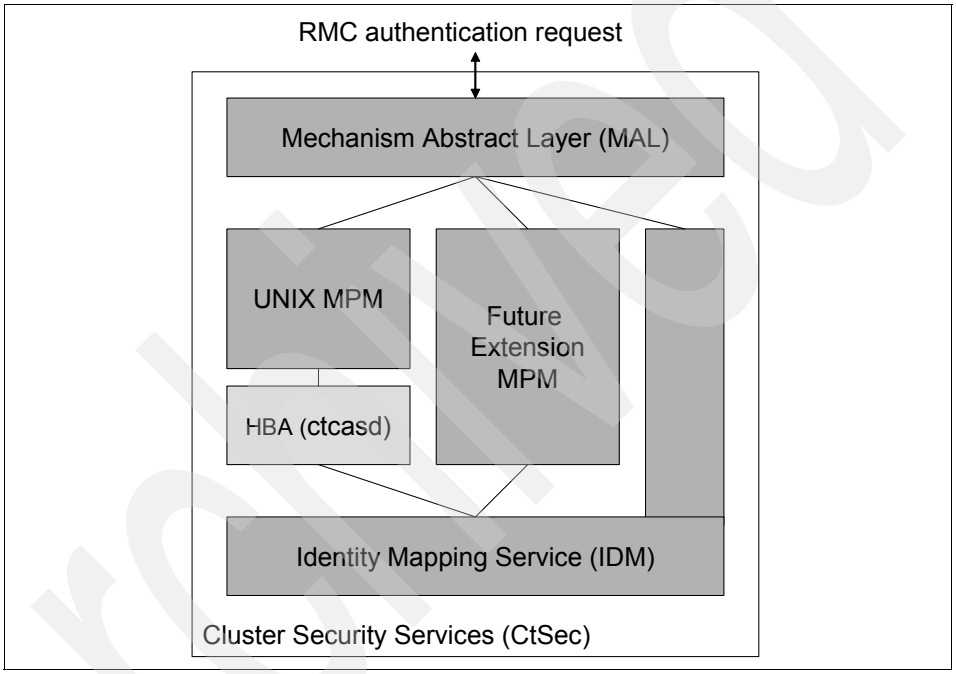

Figure 9-8 Cluster Security Services (CtSec) architecture

Within RMC, CtSec is responsible for authentication only. RMC itself is responsible for authorization by using an access control list (ACL) to grant or deny access to resources within the cluster.

9.6.3 Components of Cluster Security Services (CtSec)

The CtSec library consists of several components required to provide the current and future security functions.

The security context, created by CtSec, contains credentials for both the client and the server, the state of authentication (authenticated or unauthenticated),

and the session information (for example, session key or expiration time). The security context is created by the components of CtSec and is sent to RMC as a result of the CtSec authentication.

9.6.4 Mechanism abstract layer (MAL)

CtSec exports an interface to applications that need to implement cluster security. This interface is called mechanism abstract layer (MAL). MAL provides a generic, mechanism-independent interface for the underlying security mechanisms. MAL sends these general instructions to the configured security modules.

Those pluggable security mechanism modules are called mechanism pluggable module (MPM). The result of MAL and the configured MPM is the security context that contains credentials and possibly a session key used by both parties involved in the communication process.

If a client turns off authentication by setting the environment variable CTSEC_CC_MECH=none, MAL does not involve MPM for authentication, but returns an unauthenticated security context as the result of the authentication process.

The configuration of MPMs is done in the /var/ct/cfg/ctcec.cfg file. This file contains all MPMs CtSec should use during the authentication process. In future releases, IBM will consider the development of other security modules. Module usage is designed to be based on predefined priorities.

If a client requests access to resources, MAL is used to initiate the security context. If this process is successful, this context is cached by the MAL layer for later requests, and it is retained in memory until it is terminated by the application.

9.6.5 Mechanism pluggable module (MPM)

Each mechanism pluggable module (MPM) converts the general tasks, received from the MAL layer, into necessary tasks the security mechanism uses to satisfy the MAL request.

The MPM gathers all credentials that are necessary to fulfill the authentication process for this specific security mechanism. The MPM also maps network identities to local identities (see 9.6.8, “Identity mapping service” on page 454).

Currently only UNIX MPM is supported. UNIX MPM has been developed for both 32- and 64-bit operation, depending on the operating system kernel. Due to the modular architecture of MAL and MPM, other security mechanisms may be

added in future releases. MPMs are object modules that are loaded by the MAL during run time.

MPMs are located in the /usr/sbin/rsct/lib directory. Each MPM must have a link in /usr/lib/ that points to the respective file in /usr/sbin/rsct/lib/ (see Example 9-2).

Example 9-2 Location of MPMs

```
ls -al /usr/sbin/rsct/lib/*.mpm*
-r--r--r--   1 bin   bin   192120 Sep 06 13:26 /usr/sbin/rsct/lib/unix.mpm
-r--r--r--   1 bin   bin   199952 Sep 24 11:27 /usr/sbin/rsct/lib/unix.mpm64
ls -al /usr/lib/*.mpm*
lrwxrwxrwx 1  root  system  27 Oct 11 10:22 /usr/lib/unix.mpm ->
                                            /usr/sbin/rsct/lib/unix.mpm
lrwxrwxrwx 1  root  system  29 Oct 11 10:22 /usr/lib/unix.mpm64 ->
                                            /usr/sbin/rsct/lib/unix.mpm64
```

9.6.6 UNIX mechanism pluggable module

The core of the UNIX MPM is the ctcasd client API (see 9.6.7, “Host-based authentication with ctcasd” on page 453), which creates the host-based authentication (HBA) credentials.

The ctcasd component is called only if TCP/IP sockets for remote connections are used. If the request is for the local host, UNIX MPM uses UNIX domain sockets within the kernel. The kernel security is trusted, and no further security features are required.

9.6.7 Host-based authentication with ctcasd

The Cluster Technology Cluster Authentication Service daemon (ctcasd) creates credentials based on the node’s host name and client identity. Remember that a *client* is an application that uses the RMC client API.

Additionally, ctcasd creates a session key that can be used as a symmetric key for the node communication after finishing the authentication. This session key is created to allow RSCT to encrypt and decrypt data using a symmetric key algorithm, which is faster than public and private key (PPK) algorithms.

To ensure data privacy for the secret session key and data integrity for the host credentials, ctcasd uses the host’s public/private key pair to encrypt and decrypt this information. In actual implementation, this key pair is generated with the RSA512 algorithm. It is possible to change this to a 1024-bit key by editing the ctcasd configuration file (/usr/sbin/rsct/cfg/ctcasd.cfg).

To ensure that ctcasd uses the correct public key during the encryption process, the public keys in the trusted host list (THL) file are associated with the host name of the node. For this reason, it is necessary that all nodes within the HACMP cluster resolve names identically.

Important: To ensure identical host name resolution, all participating cluster members should use a method for name resolution that gives *identical* results on all nodes in a HACMP cluster. The name resolution method and order can be changed in /etc/netsvc.conf (for AIX systems). All hosts should also use either short or fully qualified host names. If the cluster consists of nodes in different domains, fully qualified host names *must* be used.

First, ctcasd encrypts the session key with the target host's public key derived from the THL file. This ensures that only the target node can decrypt this session key with its private key, and data privacy is ensured.

To ensure data integrity for the host credentials, ctcasd now encrypts the whole credential data structure using the initiator's private key. Everyone can decrypt the data block with the sender's public key, but the target node can be sure it was sent by the expected node.

The distribution of public keys to all nodes is performed by RSCT. By adding a node to the peer cluster, RSCT runs commands to achieve the public key exchange between the management server and its nodes.

The public key exchange is done over the network. During this exchange, the network must be secure against tracing and spoofing, because the keys are binded to a node within the cluster.

Attention: These ctcasd keys are not equal to the keys used for HACMP message encryption.

9.6.8 Identity mapping service

The identity mapping (IDM) service result is used for authorization. The IDM service maps the network identities to local identities if there is a mapping rule specified in the configuration files (called maps).

This local identity can be used by RMC to retrieve the permission from the RMC access control list (ACL). To see how RMC gathers permissions for resources from the ACL, see 9.6.9, "Resource Monitoring and Control access control list" on page 456.

Because, by default, there is no common user space inside a RSCT cluster, same user names on different hosts might not have the same permissions to access resources.

As described in 9.6.9, “Resource Monitoring and Control access control list” on page 456, RMC gathers permissions for resources by using the client’s network identity first.

For example, if user *david* should exist on all nodes within a 4-nodes cluster and should have access to a resource on a specific node, the RMC ACL file on that node must contain an entry for each network identity of the client. Example 9-3 shows the output for a specific resource in the RMC ACL file for 4 nodes.

Example 9-3 Output of /var/ct/cfg/ctrmc.acl for a specific resource entry

```
IBM.FileSystem
  david@node001   *   rw   # access for david from node 001
  david@node002   *   rw   # access for david from node 002
  david@node003   *   rw   # access for david from node 003
  david@node004   *   rw   # access for david from node 004
```

This is the main reason for which IDM was designed. IDM simply maps network identities in a management cluster to a local identity in order to avoid hundreds of lines in the RMC ACL file.

The mapping configuration file, /var/ct/cfg/ctsec_map.global, contains the mapping relationship between local and network identities.

To follow our example, the mapping file will need only one line to map the user david on all the cluster nodes to a single local identity, called mapped_david:

```
unix:david@<cluster>=mapped_david
```

This maps the user david, coming from every node in the current active cluster, to the local identity mapped_david for mappings initiated by the UNIX MPM. This local identity does not need to exist as a real user in the operating system. Access to resources works even if this identity does not exist locally (in /etc/passwd).

With this mapped identity, the resource in the ACL file needs only one entry, as shown in Example 9-4.

Example 9-4 RMC ACL file using local identities

```
IBM.FileSystem
  mapped_david   *   rw   # access for david from node 001
```

Within the ACL file, you can easily distinguish between network identities and local mapped identities. Network identities always contain an @ and the host name or node ID where the client comes from. Local mapped identities only consist of a specifier, because the host name has already been specified in the mapping file.

Initially, the global mapping file contains these entries, as shown in Example 9-4 on page 455, (for example, unix: specifies the MPM used for authentication, and =root specifies the local identity).

Example 9-5 Global mapping file

```
cat /var/ct/cfg/ctsec_map.global
unix:root@<cluster>=root
unix:root@<any_cluster>=any_root
unix:*@LOCALHOST=*
```

Basically, the first line says that every request to RMC, coming from root at any node within the current active cluster is mapped to the local identity root. Every root request from nodes outside the active cluster is mapped to any_root.

Every client request from local host to local host resources, from any user, is simply mapped to the local user. The mapping file lookup process is based on a first-matching basis, that means the first identifier that fits the given network identity will be mapped to the appropriate local identity, and no further mapping for the same network identity will occur.

9.6.9 Resource Monitoring and Control access control list

RMC uses an access control list (ACL) to grant or deny access to resources within the RSCT peer domain. The ACL is updated by HACMP during the cluster administration procedures (for example, adding and removing nodes).

After RMC successfully authenticates the client, it makes a lookup into the ACL to get the permissions on the given criteria (see Figure 9-7 on page 450). The RMC ACL file is located in /usr/sbin/rsct/cfg/ctrmc.acl.

The ACL file consists of stanzas containing the resource classes and the defined permissions of users and hosts within the cluster. RMC uses two different identities for the retrieving the permissions stored in the ACL. These identities are as follows:

- The client's network identity presented by the client's user name and its host name, for example, root@node1.
- The local mapped identity, as described in the 9.6.8, "Identity mapping service" on page 454.

The order of the lookup process in the ACL file is first the network identity and then local identity.

Part 4

Advanced topics (with examples)

The advanced topics in Part 4 cover:

- Dynamic LPAR (DLPAR) and Virtualization (VIO)
- Extending resource group capabilities
- Customizing events
- Storage related considerations
- Networking

10

Dynamic LPAR (DLPAR) and Virtualization (VIO)

In this chapter, we discuss the following topics:

- Implementing HACMP and DLPAR (on Power4 systems)
- Implementing HACMP and virtualization (vio/vscsi/vlan)

Since this is an advanced topic, a basic understanding of LPAR, DLPAR, and virtualization and their associated terminology is assumed. Detailed information about these topics can be found in the Redbooks:

- *Partitioning Implementations for IBM @server p5 Servers,* SG24-7039
- *Advanced POWER Virtualization on IBM @server p5 Servers: Introduction and Basic Configuration,* SG24-7940

These Redbooks and many others can be downloaded from:

`http://www.redbooks.ibm.com/`

The information and steps provided are in addition to an existing HACMP cluster. The steps of configuring an entire HACMP cluster are omitted to reduce replication of the steps in this publication. Information about configuring a basic HACMP cluster can be found in Chapter 4, “Cluster installation scenarios” on page 231.

10.1 Implementing DLPAR with HACMP

In this section we cover the following in regards to DLPAR:

- Requirements
- Application provisioning
- Defining DLPAR to HACMP
- Our test configuration
- Test results

It is expected that proper LPAR and DLPAR planning is part of the overall process before implementing any similar configuration. It is important to understand, not only the requirements and how to, but to understand the overall affects each decision has on the overall implementation.

10.1.1 Requirements

To use the integrated DLPAR functions, and/or CUoD, of HACMP on Power4, all LPAR nodes in the cluster should have at *least* the following levels installed:

- AIX 5.2
- HACMP 5.2.0.1 (via IY58577 for DLPAR support)
- APAR IY58497 (for CUoD support)
- RSCT 2.3.3.1
- OpenSSH 3.4p1

The OpenSSH software can be obtained from any of the following sources:

- AIX 5.2 Bonus pack
- AIX 5.3 Expansion pack
- Linux Toolbox CD
- Downloaded from:
  http://sourceforge.net/projects/openssh-aix

OpenSSh for AIX has its own software prerequisites of:

- rpm.rte
- zlib compression/decompression library (zlib)
- Psuedo random number generator daemon (prngd)
- OpenSSL Cryptographic Libraries (OpenSSL)

These prerequisites can be downloaded from:

http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html

The OpenSSL package can be found by clicking on the “AIX Toolbox Cryptographic Content” link, registering, and accepting the license agreement.

HMC attachment to the LPARs is required for proper management and DLPAR capabilities. The HMC must be network attached on a common network with the LPARs to allow remote DLPAR operations. The HMC must also have at *least* the following levels installed:

- HMC 3 Version 2.6
- HMC build level/firmware 20040113.1 or later

Important: APAR IY69525 for HACMP V5.2 or APAR IY73051 for HACMP V5.3, is required to support DLPAR, CUoD and CBU functionality on Power5 systems.

At the time of writing, the APARs required for Power5 support were unavailable. Any additional software levels needed for Power5 support have yet to be determined.

Attention: A key configuration requirement is that the LPAR partition name, the AIX hostname and the HACMP node name must all match. We show this in Figure 10-1 on page 464.

Other considerations

There are several things to consider when planing a cluster to include DLPAR operations. This include, but are not limited to:

- Encountering possible config_too_long during DLPAR events
- Mix of LPARs and non-LPAR systems
- CUoD provisioning

As Power5 DLPAR/CUoD support becomes available, there are additional possible configurations:

- Mixing Power4 and Power5 DLPAR
- Using shared and/or dedicated CPUs
- Using capped and/or uncapped CPUs

As with any cluster, the configuration must be tested thoroughly. This includes anything that can be done to simulate or produce a real work load for the most realistic test scenarios as possible.

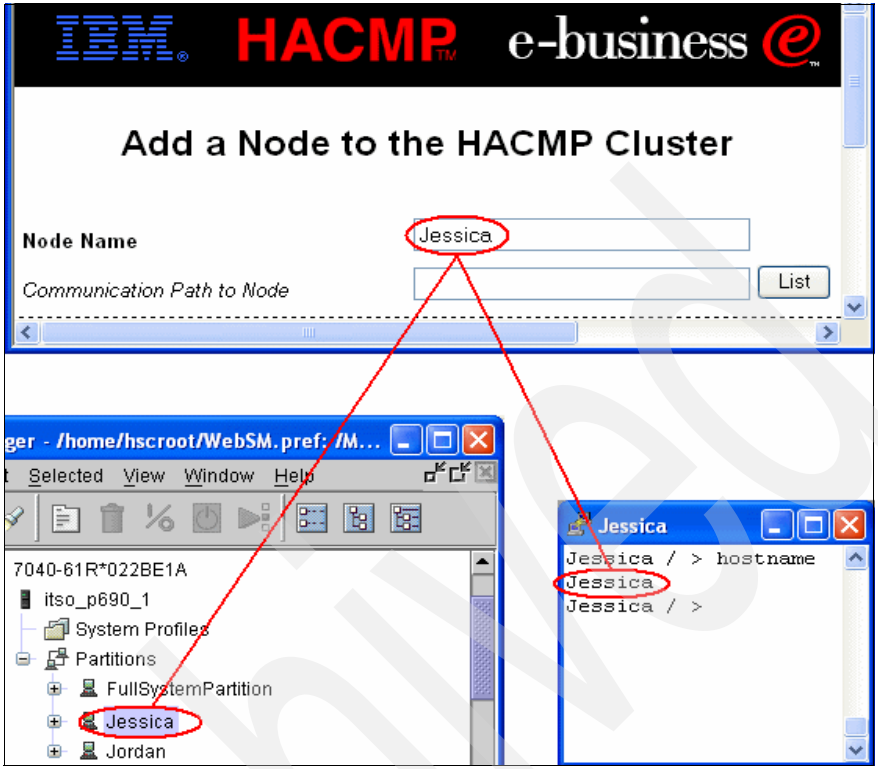

Figure 10-1 Partition name, AIX hostname, HACMP nodename matching

10.1.2 Application provisioning

Details on this topic can be found *High Availability Cluster Multi-Processing Administration Guide*, SC23-4862-06.

This section describes the flow of actions in the HACMP cluster, if the application provisioning function through DLPAR and CUoD is configured. It also includes several examples that illustrate how resources are allocated, depending on different resource requirements.

Overview

When you configure an LPAR on the HMC (outside of HACMP), you provide LPAR minimum, desired and maximum values for the number of CPUs and amount of memory. These values can be obtained by running the **`lshwres`** command on the HMC. The stated minimum values of the resources must be available when an LPAR node starts. If more resources are available in the free pool on the frame, an LPAR can allocate up to the stated desired values.

During dynamic allocation operations, the system does not allow that the values for CPU and memory go below the minimum or above the maximum amounts specified for the LPAR.

HACMP obtains the LPAR minimums and LPAR maximums amounts and uses them to allocate and release CPU and memory when application servers are started and stopped on the LPAR node.

HACMP requests the DLPAR resource allocation on the HMC before the application servers are started, and releases the resources after the application servers are stopped. The Cluster Manager waits for the completion of these events before continuing the event processing in the cluster.

HACMP handles the resource allocation and release for application servers serially, regardless if the resource groups are processed in parallel. This minimizes conflicts between application servers trying to allocate or release the same CPU or memory resources. Therefore, you must carefully configure the cluster to properly handle all CPU and memory requests on an LPAR.

These considerations are important:

- Once HACMP has acquired additional resources for the application server, when the application server moves again to another node, HACMP releases only those resources that are no longer necessary to support this application on the node.
- HACMP does *not* start and stop LPAR nodes.

It is possible to create a custom event or customize application start/stop scripts to stop LPAR nodes if desired.

Acquiring DLPAR and CUoD Resources

If you configure an application server that requires a minimum and a desired amount of resources (CPU or memory), HACMP determines if additional resources need to be allocated for the node and allocates them if possible.

In general, HACMP tries to allocate as many resources as possible to meet the desired amount for the application, and uses CUoD, if allowed, to do this.

The LPAR Node has the LPAR Minimum

If the node owns only the minimum amount of resources, HACMP requests additional resources through DLPAR and CUoD (if applicable).

In general, HACMP starts counting the extra resources required for the application from the minimum amount. That is, the minimum resources are

retained for the node's overhead operations, and are *not* utilized to host an application.

The LPAR Node has Enough Resources to Host an Application

The LPAR node that is about to host an application may already contain enough resources (in addition to the LPAR minimum) to meet the desired amount of resources for this application.

In this case, HACMP does not allocate any additional resources and the application can be successfully started on the LPAR node. HACMP also calculates that the node has enough resources for this application in addition to hosting all other application servers that may be currently running on the node.

Resources Requested from the Free Pool and from the CUoD Pool

If the amount of resources in the free pool is insufficient to satisfy the total amount requested for allocation (minimum requirements for one or more applications), HACMP requests resources from CUoD (if enabled).

If HACMP meets the requirement for a minimum amount of resources for the application server, application server processing continues. Application server processing continues even if the total desired resources (for one or more applications) have not been met or are only partially met. In general, HACMP attempts to acquire up to the desired amount of resources requested for an application.

If the amount of resources is insufficient to host an application, HACMP starts resource group recovery actions to move the resource group to another node.

Minimum Amount Requested for an Application Cannot be Satisfied

In some cases, even after HACMP requests to use resources from the CUoD pool, the amount of resources it can allocate is less than the minimum amount specified for an application.

If the amount of resources is still insufficient to host an application, HACMP starts resource group recovery actions to move the resource group to another node.

The LPAR node is Hosting Application Servers

In all cases, HACMP checks whether the node is already hosting application servers that required application provisioning, and that the LPAR maximum for the node is not exceeded:

- Upon subsequent fallovers, HACMP checks if the minimum amount of requested resources for yet another application server plus the amount of

resources already allocated to applications residing on the node exceeds the LPAR maximum.

- In this case, HACMP attempts resource group recovery actions to move the resource group to another LPAR. Note that when you configure the DLPAR and CUoD requirements for this application server, then during cluster verification, HACMP warns you if the total number of resources requested for all applications exceeds the LPAR maximum.

Allocation of Resources in a Cluster With Multiple Applications

If you have multiple applications in different resource groups in the cluster with LPAR nodes, and more than one application is configured to potentially request additional resources through the DLPAR and CUoD function, the resource allocation in the cluster becomes more complex.

Based on the resource group processing order, some resource groups (hence the applications) might not be started. We explain this further in “Examples of using DLPAR and CUoD Resources” on page 468

Releasing DLPAR and CUoD Resources

When the application server is stopped on the LPAR node (the resource group moves to another node), HACMP releases only those resources that are no longer necessary to support this application server on the node. The resources are released to the free pool on the frame.

HACMP first releases the DLPAR or CUoD resources it acquired last. This implies that the CUoD resources may not always be released before the dynamic LPAR resources are released.

The free pool is limited to the single frame only. That is, for clusters configured on two frames, HACMP does not request resources from the second frame for an LPAR node residing on the first frame.

Also, if LPAR 1 releases an application that puts some DLPAR resources into free pool, LPAR 2 which is using the CUoD resources does not make any attempt to release its CUoD resources and acquire the free DLPAR resources.

Stopping LPAR Nodes

When the Cluster Manager is forced down on an LPAR node, and that LPAR is subsequently shutdown (outside of HACMP), the CPU and memory resources are released (not by HACMP) and become available for other resource groups running on other LPARs. HACMP does not track CPU and memory resources that were allocated to the LPAR and does not retain them for use when the LPAR node rejoins the cluster.

Note: If you are using the On/Off license for CUoD resources, and the LPAR node is shutdown (outside of HACMP), the CUoD resources are released (not by HACMP) to the free pool, but the On/Off license continues to be turned on. You may need to manually turn off the licence for the CUoD resources that are now in the free pool. (This ensures that you do not pay for resources that are not being currently used).

If the LPAR is not stopped after the Cluster Manager is forced down on the node, the CPU and memory resources remain allocated to the LPAR for use when the LPAR rejoins the cluster.

Changing the DLPAR and CUoD Resources Dynamically

You can change the DLPAR and CUoD resource requirements for application servers without stopping the cluster services. Synchronize the cluster after making the changes.

The new configuration is not reflected until the next event that causes the application (hence the resource group) to be released and reacquired on another node. In other words, a change in the resource requirements for CPUs, memory or both does not cause the recalculation of the DLPAR resources. HACMP does not stop and restart application servers solely for the purpose of making the application provisioning changes.

If *another* dynamic reconfiguration change (i.e., an rg_move) causes the resource groups to be released and reacquired, the new resource requirements for DLPAR and CUoD are used at the end of this dynamic reconfiguration event.

Examples of using DLPAR and CUoD Resources

The following examples explain CPU allocation and release. The process for memory is very similar. While these are descriptions of how it works, we also provide real results from our test configuration in “Test results” on page 488

Note: Be aware that once HACMP acquires additional resources for an application server, when the server moves again to another node, it takes the resources with it, that is, the LPAR node releases all the additional resources it acquired.

The configuration is an 8 CPU frame, with a two-node (each an LPAR) cluster. There are 2 CPUs available in the CUoD pool, that is through the CUoD activations. The nodes have the partition profile characteristics shown in Table 10-1 on page 469 and Table 10-2 on page 469:

Table 10-1 Profile characteristics

Node Name	LPAR Minimum	LPAR Maximum
Longhorn	1	9
Hurricane	1	5

There are three application servers defined, each belonging to separate resource groups.

Table 10-2 Application requirements

Application Server Name	CPU Desired	CPU Minimum	Allow CUoD
App1	1	1	Yes
App2	2	2	No
App3	4	4	No

Example 1: No CPUs added at start, some are released upon stop

The starting configuration settings are as follows:

- Longhorn has 3 CPUs allocated
- Hurricane has 1 CPU allocated
- Free pool has 4 CPUs allocated

The applications servers are started in the following order:

- Longhorn starts App2, no CPUs are allocated to meet the requirement of 3 CPUs. (3 CPUs is equal to the sum on Node1's LPAR minimum of 1 plus App2 desired amount of 2).
- Longhorn stops App2. 2 CPUs are released, leaving 1 CPU, the minimum requirement. (Since no other application servers are running, the only requirement is Longhorn LPAR minimum of 1).

Example 2: No CPUs added due to RG Processing Order

In this example we start off with the same configuration settings as shown in the example above.

The application servers are started as follows:

- Longhorn starts App1, no CPUs are allocated since the requirement of 2 is met. Longhorn starts App3, 3 CPUs are allocated to meet the requirement of 6. There is now 1 CPU in the free pool.

– Longhorn attempts to start App2. After Longhorn has acquired App1 and App3, the total amount of CPUs Longhorn must now own to satisfy these requirements is 6, which is the sum of Longhorn LPAR minimum of 1 plus App1 desired amount of 1 plus App3 desired amount of 4.

  Since App2 minimum amount is 2, in order to acquire App2, Longhorn needs to allocate 2 more CPUs, but there is only 1 CPU left in the free pool and it does not meet the minimum requirement of 2 CPUs for App2. The resource group with App2 is not acquired locally as there is only 1 CPU in the free pool and CUoD use is not allowed. If no other member nodes are present then the resource group goes into the error state.

  If node hurricane would have been a member node of the App 2 resource group, and active in the cluster, then an rg_move would have been invoked in an attempt to bring up the resource group on node hurricane.

Example 3: Successful CUoD Resources Allocation and Release

The starting configuration settings are as follows:

– Longhorn has 3 CPUs allocated.
– Hurricane has 1 CPU allocated.
– The free pool has 4 CPUs.

The application servers are started in the following order:

– Longhorn starts App3, 2 CPUs are allocated to meet the requirement of 5.
– Longhorn starts App2, 2 CPUs are allocated to meet the requirement of 7. There are now no CPUs in the free pool.
– Longhorn starts App1, 1 CPU is taken from CUoD and allocated to meet the requirement of 8.
– Longhorn stops App3, 4 CPUs are released and 1 of those CPUs is put back into the CUoD pool.

Example 4: Resource Group Failure, minimum resources not met

In this example the resource group acquisition fails due to the fact that the minimum resources needed are not currently available as the LPAR has reached it's maximum.

The configuration is as follows:

– Longhorn has 1 CPU.
– Hurricane has 1 CPU.
– Free pool has 6 CPUs.

The application servers are started in the following order

– Hurricane starts App3, 4 CPUs are allocated to meet the requirement of 5. There are now 2 CPUs in the free pool.
– Hurricane attempts to start App2, but App2 goes into error state since the LPAR maximum for hurricane is 5 and hurricane cannot acquire more CPUs.

Note: If the minimum resources for App2 would have been set zero instead of one, the acquisition would have succeeded as no additional resources would have been required.

Example 5: Resource group failure, LPAR min. and max. are same

In this example we demonstrate a real example we encountered during our early testing. This is a direct result of improper planning in regards to how application provisioning works.

We are still using an 8 CPU frame, however the additional application servers and nodes are not relevant to this example.The LPAR configuration for node longhorn is shown in Table 10-3:

Table 10-3 LPAR characteristics for node longhorn

LPAR Minimum	LPAR Desired	LPAR Maximum
4	4	4

The App1 application server has the settings shown in Table 10-4:

Table 10-4 Application requirements for App1

Minimum number of CPUs	Desired number of CPUs
1	4

The starting configuration is as follows:

– Longhorn has 4 CPUs allocated.
– Free pool has 4 CPUs

App1 application server is started locally on node longhorn. During acquisition the, the LPAR minimum is checked and added to the application server minimum which returns a total of 5. This total exceeds the LPAR maximum setting and results in the resource group going into the error state.

Though technically the LPAR may already have enough resources to host the application, because of the combination of settings, it results in a failure.

Generally speaking you would not have the minimum and maximum settings equal.

This scenario could have been avoided in any one of these three ways:

- Change LPAR min. to 3 or less.
- Change LPAR max. to more than 4.
- Change App1 minimum CPUs to 0.

10.1.3 Configuring DLPAR to HACMP

Some of the following information came from an existing whitepaper that was created prior to the integration of DLPAR with HACMP. This whitepaper focused on customizing HACMP event script to utilize DLPAR. However, it also included other key preparations ares as well. This white paper has been used internally to IBM and its business partners.

We will cover the following steps needed to configure DLPAR to an HACMP cluster.

- Name resolution
- Install ssh on HACMP nodes
- Configure HMC ssh access
- Define HMC and managed systems to HACMP
- Define DLPAR resources to HACMP (i.e application provisioning)

Name Resolution

One common issue seen is name resolution not being consistent across all systems. If name resolution is not configured correctly, the DLPAR feature cannot be used. The underlying Reliable Scalable Cluster Technology (RSCT) infrastructure expects an identical hostname resolution on all participating nodes. If this is not the case RSCT will be unable to communicate properly.

Ensure that all nodes and the HMC are configured identical by checking the following list. The phrase ‘All systems’ includes all HACMP nodes and the HMC.

- All systems must resolve the participating hostnames and IP addresses identical. This includes reverse name resolution.
- All systems must use the same type of name resolution, either short or long name resolution
- All systems should use the same name resolution order, either local or remote. To ensure this, check the following files:
  - /etc/hosts on all systems

- /etc/netsvc.conf on all AIX nodes
- /etc/host.conf on the HMC

We expect it is common knowledge how to check these files on the AIX systems. It is not as well known how to do so on the HMC.

Make sure that the HMC is aware of the host LPARs. The host names and IPs should be listed in the HMC hosts list. We recommend configuring the hosts information through the HMC console since each version of the HMC code continues to restrict command line options. You can verify this on the Power4 HMC by clicking on **HMC Maintenance->System Configuration->Customize Network Settings->Hosts**. If the addresses and names are not listed, you can add them by clicking on **New**. Our HMC host file configuration is shown in Figure 10-2 on page 474.

Note: If using Power5 HMCs, click **HMC Management->HMC Configuration->Customize Network Settings.**

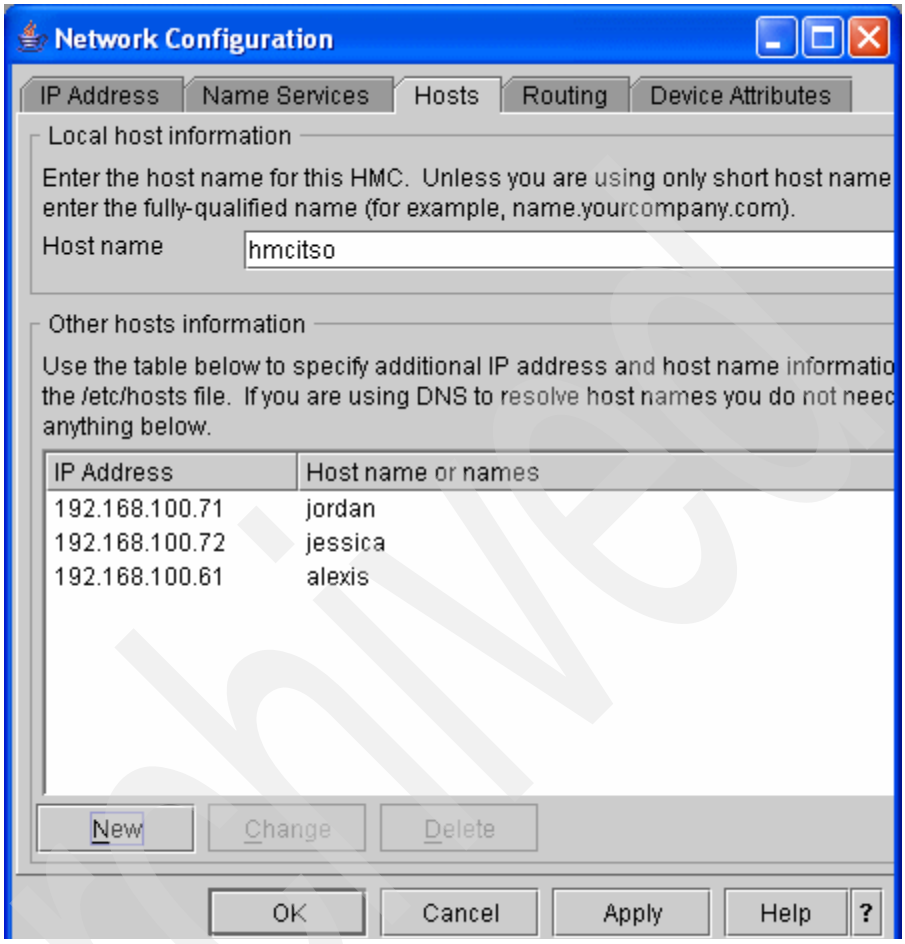

Figure 10-2 HMC Hosts

Install and configure SSH on HACMP nodes

In order to use remote command operations on the HMC it is required to have SSH installed on the HACMP nodes. The HMC must be configured to allow access from these partitions.

In this section we cover installing the ssh packages, including what order to install them in. These packages have very little flexibility in the install order. In most cases, if something is installed out of order, an error message will normally point you to which package is needed first.

With each version of SSH and HMC code these steps may differ slightly. We have documented our processes used to successfully implement our environment. Information for installing SSH on each version of AIX can be found at:

http://www-1.ibm.com/support/docview.wss?uid=isg1pTechnote0707

Installing SSH

In "Requirements" on page 462, we covered which packages are needed and where to obtain them. The following steps assume these packages have been downloaded or copied onto the HACMP nodes. We chose to put all of our images in the common install directory of */usr/sys/inst.images*.

The *rpm.rte* package must be installed prior to installing the additional rpm packages. You can use either **installp** or **smit install_all** to install it. You can verify it is installed by running **lslpp -l rpm.rte** (see Example 10-1).

Example 10-1 Checking if rpm is installed

```
Jordan / > lslpp -l rpm.rte
  Fileset                      Level  State      Description
  ----------------------------------------------------------------------------
Path: /usr/lib/objrepos
  rpm.rte                    3.0.5.36  COMMITTED  RPM Package Manager
Path: /etc/objrepos
  rpm.rte                    3.0.5.36  COMMITTED  RPM Package Manager
```

It is now possible to install the remaining prerequisites by using the **rpm** command. We installed these packages as in Example 10-2:

Example 10-2 Installing openSSH prerequisites

```
rpm -i zlib-1.2.1-2.aix5.1.ppc.rpm
rpm -i prngd-0.9.23-3.aix4.3.ppc.rpm
rpm -i openssl-0.9.7d-2.aix5.1.ppc.rpm
rpm -i openssl-devel-0.9.7d-2.aix5.1.ppc.rpm
rpm -i openssl-doc-0.9.7d-2.aix5.1.ppc.rpm
```

This fulfills the requirements needed to install SSH. In AIX 5.1 and above, the openSSH package is in **installp** format. Assuming the image has been extracted from the tar package, you can now install using **smitty install_all.** There are three core filesets to install. These filesets and the results of our install are shown in Example 10-3.

Example 10-3 Install SSH

```
smitty install_all
> openssh.base                                                        ALL ¦
  ¦    + 3.8.0.5202  Open Secure Shell Commands                          ¦
```

```
  |    + 3.8.0.5202  Open Secure Shell Server                          |
  | > openssh.license                                                ALL |
  |    + 3.8.0.5202  Open Secure Shell License                         |
  |                                                                    |
  | > openssh.man.en_US                                              ALL |
  |    + 3.8.0.5202  Open Secure Shell Documentation - U.S. English    |

Name                  Level           Part        Event       Result
-------------------------------------------------------------------------
openssh.license       3.8.0.5202      USR         APPLY       SUCCESS
openssh.base.client   3.8.0.5202      USR         APPLY       SUCCESS
openssh.base.server   3.8.0.5202      USR         APPLY       SUCCESS
openssh.base.client   3.8.0.5202      ROOT        APPLY       SUCCESS
openssh.base.server   3.8.0.5202      ROOT        APPLY       SUCCESS
openssh.man.en_US     3.8.0.5202      USR         APPLY       SUCCESS
```

Tip: Be sure to choose *yes* on the field to accept the license agreement.

Now that SSH is installed we need to configure the HACMP nodes to access the HMC without passwords for remote DLPAR operations.

Configure HMC SSH access

The document “Managing the Hardware Management Console” contains a section that describes the steps to setup a remote secure shell access. The guide can be downloaded from the IBM Web site at:

http://publib.boulder.ibm.com/infocenter/iseries/v1r2s/en_US/info/iphai/iphai.pdf

These are the steps we used in our setup to enable SSH access from our HACMP nodes

- Enable HMC SSH access
- Generate SSH keys on HACMP nodes.
- Enable non-password HMC access via *authorized_keys2* file

First, make sure the HMC is setup to allow remote operations by doing the following:

1. In the Navigation area, select **HMC Maintenance**.
2. In the Navigation area, select **System Configuration**.
3. In the Contents area, click **Enable/Disable Remote Command Execution**.
4. Select the box to enable ssh.

Note: Normally it is recommended to create a separate HMC user for remote command execution, however HACMP uses `hscroot`.

It is needed to create the SSH directory $HOME/.ssh for user root to store the authentication keys. HACMP will execute the ssh remote DLPAR operations as the root user. By default this is */.ssh,* and is what we used.

To generate public and private keys, run the following command on each HACMP node:

```
/usr/bin/ssh-keygen -t rsa
```

This will create the following files in */.ssh:*

```
private key: id_rsa
public key: id_rsa.pub
```

The write bits for both group and other are turned off. Ensure that the private key has a permission of 600.

The HMC's public key needs to be in *known_hosts* file on each HACMP node, and vice versa. This is easily accomplished by executing **`ssh`** to the HMC from each HACMP node. The first time executed, a prompt will be displayed to insert the key into the file. Answer *yes* to continue, and then you will be prompted to enter a password. It is not necessary to do so as we have not completed the setup yet to allow non-password ssh access (see Example 10-4).

Example 10-4 SSH to HMC

```
Jordan /tmp > ssh -l hscroot 192.168.100.69
The authenticity of host '192.168.100.69 (192.168.100.69)' can't be
established.
 RSA key fingerprint is 2d:50:3f:03:d3:51:96:27:5a:5e:94:f4:e3:9b:e7:78
 Are you sure you want to continue connecting (yes/no)?yes
Warning: Permanently added '192.168.100.69' (RSA) to the list of known
hosts.
```

When utilizing two HMCs, like in our test configuration, it is necessary to repeat this process for each HMC. You may also want to do this between all member nodes to allow ssh type of operations between them (i.e., scp, sftp, and ssh).

To allow non-password ssh access, we must put each HACMP node's public key into the *authorized_keys2* file on the HMC. This can be done more than one way, however here is an overview of the steps we used:

1. Create authorized_keys2 file on HMC

2. Copy (use **scp**) the public key from all nodes to one machine.
3. Concatenate (**cat**) all the key files together into authorized_keys2 file.
4. Copy (**scp**) the concatenated file over to the HMC /home/hscrtoot/.ssh

For the first step, we created the authorized_keys2 file manually. This can be done by either the command line at HMC, or remotely from the client. We executed it remote from the client. We chose to create a dummy file first, then scp the contents of our combined key files over to the HMC, essentially replacing our dummy file. To create the file with dummy information, we ran from one client:

```
ssh hscroot@hmc “mkauthkeys --add ‘*’ ”
```

Verify on the HMC that the authorized_keys2 file exists in .ssh directory. We did this from the client by executing:

```
ssh hscroot@hmc "ls -al .ssh/"
```

Note: You can run the **mkauthkeys** command, via ssh, from each AIX client and add one key at time as documented in the *Managing the Hardware Management Console* guide. However, syntactically it requires adding the key string in manually. We found this to be cumbersome, especially when having to execute on multiple system.

Next, from /.ssh on the AIX LPARs, we made a copy of the public key and renamed it to include the local nodename as part of the file name. We then copied, via scp, the public key of each machine (Jessica and Alexis) to one node (Jordan). We then ran the **cat** command to create an authorized_keys2 file that contains the public key information for all HACMP nodes. The commands run on each node are shown in Example 10-5:

Example 10-5 Scp authorized_keys2 file to HMC

```
Alexis /.ssh > cp id_rsa.pub id_rsa.pub.alexis
Alexis /.ssh > scp id_rsa.pub.alexis jordan:/.ssh/id_rsa.pub.alexis

Jessica /.ssh > cp id_rsa.pub id_rsa.pub.jessica
Jessica /.ssh > scp id_rsa.pub.jessica jordan:/.ssh/id_rsa.pub.jessica

Jordan /.ssh > cp id_rsa.pub id_rsa.pub.jordan
Jordan /.ssh > cat id_rsa.pub.alexis id_rsa.pub.jessica id_rsa.pub.jordan
>authorized_keys2

Jordan /.ssh > ls -al
total 64
drwx------   2 root     system          256 Jul 18 22:27 .
drwxr-xr-x  21 root     system         4096 Jul 14 02:11 ..
-rw-r--r--   1 root     system          664 Jun 16 16:31 authorized.keys2
```

```
-rw-------   1 root     system          883 Jun 16 14:12 id_rsa
-rw-r--r--   1 root     system          221 Jun 16 14:12 id_rsa.pub
-rw-r--r--   1 root     system          221 Jun 16 16:30 id_rsa.pub.alexis
-rw-r--r--   1 root     system          222 Jun 16 15:20 id_rsa.pub.jessica
-rw-r--r--   1 root     system          221 Jun 16 16:27 id_rsa.pub.jordan
-rw-r--r--   1 root     system         1795 Jul 14 04:08 known_hosts

Jordan/.ssh > scp authorized.keys2 hscroot@192.168.100.69:.ssh/authorized_keys2
hscroot@192.168.100.69's password:
authorized_keys2                              100% 664 0.7KB/s   00:00
```

When executing **scp** to the HMC you should be prompted to enter the password for the hscroot user. Once entered, the authorized_key2 will be copied. You can then test if the no-password access is working from each node by executing the ssh command as shown in Example 10-4 on page 477. However, this time you should end up at the HMC shell prompt as shown in Example 10-6.

Example 10-6 Test no-password ssh access

```
Alexis /.ssh > ssh -l hscroot 192.168.100.
Last login:Thur Jun 16 22:46:51 2005 from 192.168.100.61
hscroot@hmcitso:~>
```

Once each node can ssh to the HMC without a password, then this step is completed and HACMP verification of the HMC communications should succeed.

Defining HMC and managed system names

The HMC(s) IP address(es) must be specified for each HACMP node that will be utilizing DLPAR. In our example, each HACMP corresponds to an LPAR. Each LPAR is assigned to a *managed system*. Managed systems are those systems that are physically to, and managed, by the HMC. These managed systems must also be defined to HACMP.

You can obtain the managed system names through the HMC console in the navigation area. The managed system name can be a user created name or the default name is the machine type and serial number.

To define the HMC communication for each HACMP node:

1. In smit hacmp, select **Extended Configuration->Extended Resource Configuration->HACMP Extended Resources Configuration->Configure HACMP Applications->Configure HACMP for Dynamic LPAR and CUoD Resources->Configure Communication Path to HMC->Add HMC IP Address for a Node** and press Enter.

The **Add HMC IP Address** screen appears.

Tip: You can use the smit fastpath of **`smit cladd_apphmc.dialog`**

2. Fill out the following fields as appropriate:
   - **Node Name**: Select a node name to associate with one or more. Hardware Management Console (HMC) IP addresses and a Managed System.
   - **HMC IP Address(es)**: Enter one or more space-separated IP addresses for the HMC. If addresses are added for more than one HMC, HACMP tries to communicate with each HMC until a working communication path is found. Once the communication path is established, HACMP uses this path to execute the dynamic logical partition commands on that HMC.
   - **Managed System Name**: Enter the name of the Managed System that runs the LPAR that represents the node. The maximum length is 32 characters.
3. Press Enter.

Figure 10-3 on page 481 shows the smit screen to add the HMC information we listed above. We also show the HMC managed system name information as used in our test configuration.

Note: The DLPAR/CUoD Appendix in the HACMP Administration Guide states that the managed system name can *not* include underscores. However, in our example we did use underscores and it worked fine. Discussions with development reaffirmed that this should not be a stated limitation.

During cluster verification HACMP verifies that the HMC is reachable by first issuing a **`ping`** to the IP address specified. If the HMC responds, then HACMP will verify that each specified HACMP node is in fact DLPAR capable by issuing an **`lssycfg command`**, via ssh, on the HMC.

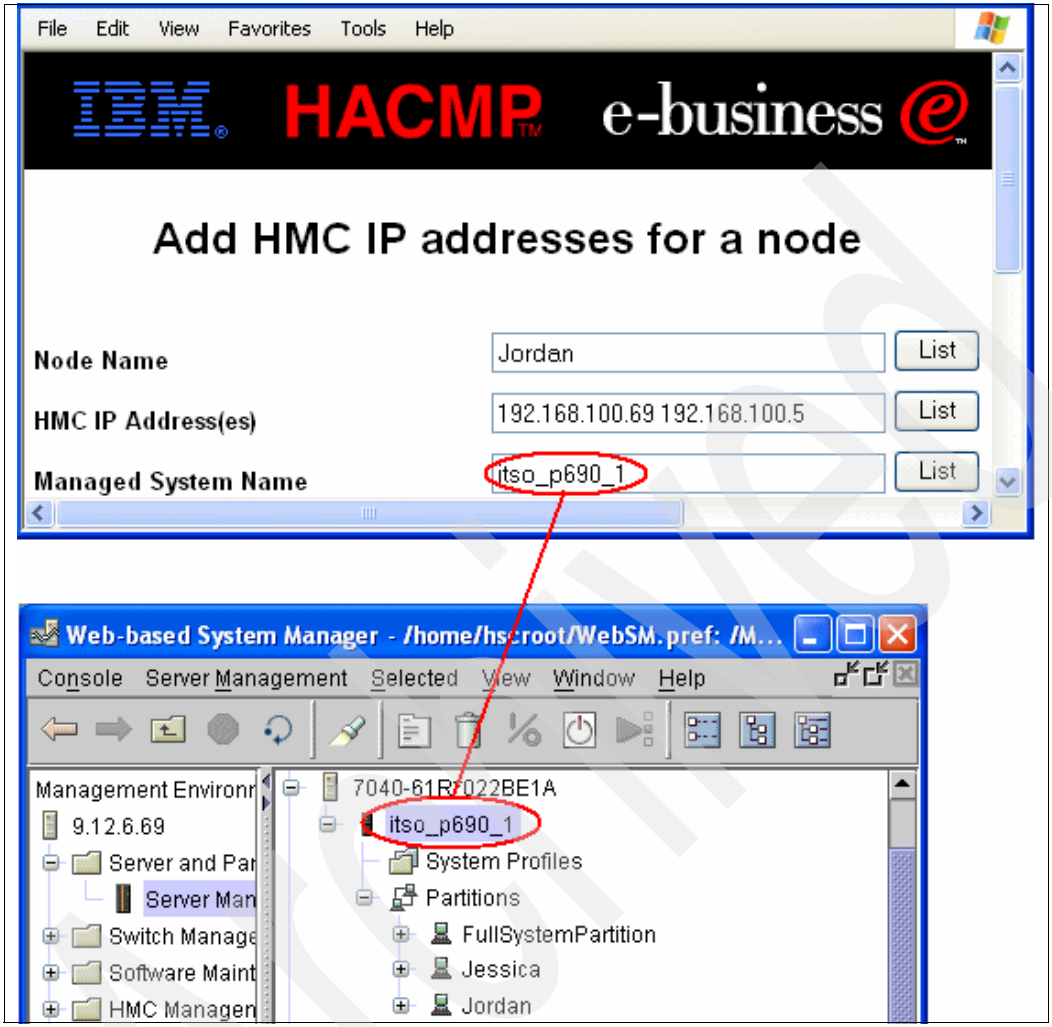

Figure 10-3 Defining HMC and Managed System to HACMP

Configure application provisioning

To configure dynamic LPAR and CUoD resources, for each application server that could use DLPAR-allocated or CUoD resources:

1. In smit hacmp, select **`Extended Configuration-> Extended Resource Configuration-> HACMP Extended Resources Configuration-> Configure HACMP Applications-> Configure HACMP for Dynamic LPAR and CUoD Resources->Configure Dynamic LPAR and CUoD Resources for Applications-> Add Dynamic LPAR and CUoD Resources for Applications`** and press Enter.

A picklist of configured application servers appears.

Tip: You can use the smit fastpath of **`smit cladd_appdlpar.dialog`**

2. Select an application server from the list and press Enter.

   The screen to specify the requirements for an application server appears. Detailed information can be found in the help screens and in “Application provisioning” on page 464

3. Fill out the following fields as appropriate:
   - **Application Server Name**: This is the application server for which you will configure Dynamic LPAR and CUoD resource provisioning that was chosen from the previous menu.
   - **Minimum Number of CPUs**: Enter the minimum number of CPUs to acquire when the application server starts. The default value is 0.To perform the application provisioning, HACMP checks how many CPUs the LPAR node currently has above its LPAR minimum value, compares this number with the minimum requested in this field and based on this, requests more CPUs, if needed.
   - **Number of CPUs**: Enter the maximum amount of CPUs HACMP will attempt to allocate to the node before starting this application on this node. The default value is 0.
   - **Minimum Amount of Memory** : Enter the amount of memory to acquire when the application server starts. Must be a multiple of 256.
   - **Use CUoD if resources are insufficient?**: The default is No. Select Yes to have HACMP use Capacity Upgrade on Demand (CUoD) to obtain enough resources to fulfill the minimum amount requested. Using CoD requires a license key (activation code) to be entered on the Hardware Management Console (HMC) and may result in extra costs due to usage of the CoD license.
   - **I agree to use CUoD resources**: The default is No. Select Yes to acknowledge that you understand that there might be extra costs involved when using CUoD. HACMP logs the answer to the syslog and smit.log files
4. Press Enter.

When the application requires additional resources to be allocated on this node, HACMP performs its calculations to see whether it needs to request only the DLPAR resources from the free pool on the frame and whether that would already satisfy the requirement, or if CUoD resources are also needed for the

application server. After that, HACMP proceeds with requesting the desired amounts of memory and numbers of CPU, if you selected to use them.

During verification, HACMP ensures that the entered values are below LPAR maximum values for memory and CPU. Otherwise HACMP issues an error, stating these requirements.

HACMP also verifies that the total of required resources for ALL application servers that can run concurrently on the LPAR is less than the LPAR maximum. If this requirement is not met, HACMP issues a warning. Note that this scenario can happen upon subsequent fallovers. That is, if the LPAR node is already hosting application servers that require DLPAR and CUoD resources, then upon acquiring yet another application server, it is possible that the LPAR cannot acquire any additional resources beyond its LPAR maximum. HACMP verifies this case and issues a warning.

An application provisioning example for our test configuration can be seen in Figure 10-4 on page 484.

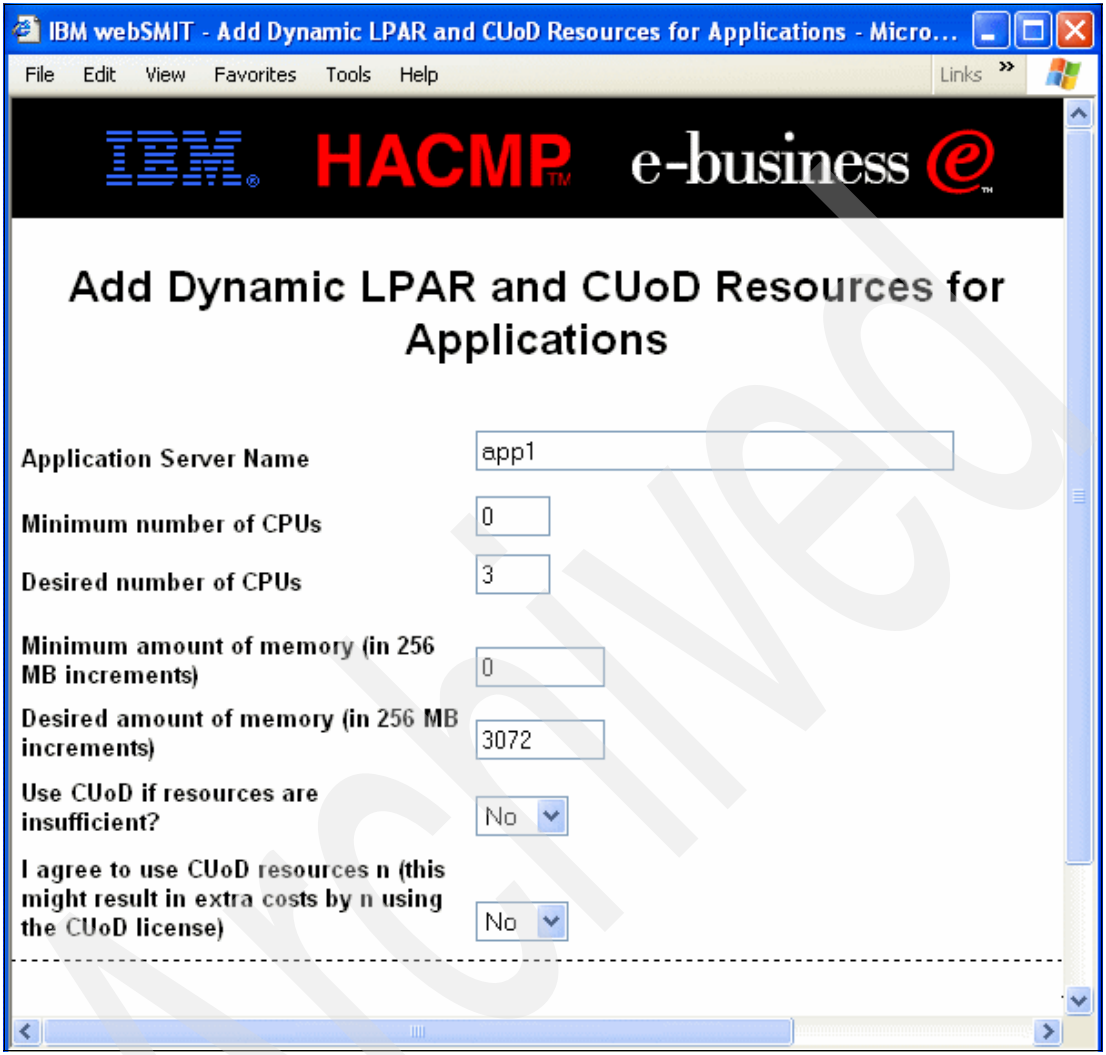

Figure 10-4 Adding application provision to HACMP

After adding both the HMC communications and application provisioning it is necessary to synchronize the cluster.

10.1.4 Troubleshooting HMC verification errors

In this section we show some errors that could be encountered during verification, along with possibilities of why the errors are generated. Though some of the error messages seem self explanatory we believe any tips in troubleshooting are normally welcome.

Example 10-7 HMC unreachable during verification

```
ERROR: The HMC with IP label 192.168.100.69 configured on node jordan is not
reachable. Make sure the HMC IP address is correct, the HMC is turned on and
connected to the network, and the HMC has OpenSSH installed and setup with the
public key of node jordan.
```

In Example 10-7 the error message itself gives good probable causes to the problem. Here are some things you can do to discover the source of the problem:

- Ping the HMC IP address
- Manually ssh to the HMC via **ssh -l hscroot hmcip**

If the ssh is unsuccessful or prompts for a password that is an indication that ssh has not been properly configured.

Example 10-8 Node not DLPAR capable verification error

```
ERROR: An HMC has been configured for node jordan, but the node does
not appear to be DLPAR capable.
```

If the message shown in Example 10-8 appears by itself, this is normally an indication that access to the HMC is working, however the particular node's matching LPAR definition is not reporting that it is DLPAR capable. You can verify this manually from the HMC command line as shown in Example 10-9.

Example 10-9 Verify LPAR is DLPAR capable

```
hscroot@hmcitso:~> lssyscfg -r lpar -m itso_p690_1 -n Jordan
Name     id   DLPAR  State    Profile      OpPanel
Jordan   001  NO     Running  Jordan_Prod
```

Note: The HMC command syntax can vary by HMC code levels and type.

This may be caused by something as simple as the node not running at least AIX 5.2 that is required for DLPAR operations. It can also be from RMC not updating properly.

During our testing, we ran several events within very short periods of time. At some point we would see that our LPAR would report it was no longer DLPAR capable. Then after a short period it would report back normally again. We believe this was due to RMC information between the LPARs and the HMC would get out of sync.

10.1.5 Test cluster configuration

Our test configuration consists of three LPARs distributed over two p690s. There are two production LPARs (Jordan, Jessica) in one p690 (itso_p690_1) and one standby LPAR (Alexis) in the second p690 (itso_p690_2). Each p690 contains eight CPUs and 8 GB of memory and are attached to two HMCs for redundancy.

Each partition have the following software levels installed:

- AIX 5.2 ML5
- HACMP 5.2.0.3
- RSCT 2.3.5.0
- rpm-3.0.5-37.aix5.1
- OpenSSH 3.8.1p1 (and the following prerequisites)
  - zlib 1.2.1-2
  - prngd 0.9.23-3
  - openssl 0.9.7d-2

Each partition have the following adapters installed:

- (2) IBM Gigabit FC Adapter (#6228)
- (2) 10/100 Ethernet (#2975)

Both HMCs have:

- HMC 3 version 3.5
- HMC build level/firmware 20050320.1

Our test LPARs, and their corresponding frames can be seen in Figure 10-5 on page 487.

For shared storage we used an ESS with eight 10GB luns, four of which were assigned to each production resource group. For purposes our testing, the shared storage was not important other than trying to set up a more complete cluster configuration by utilizing disk heartbeat.

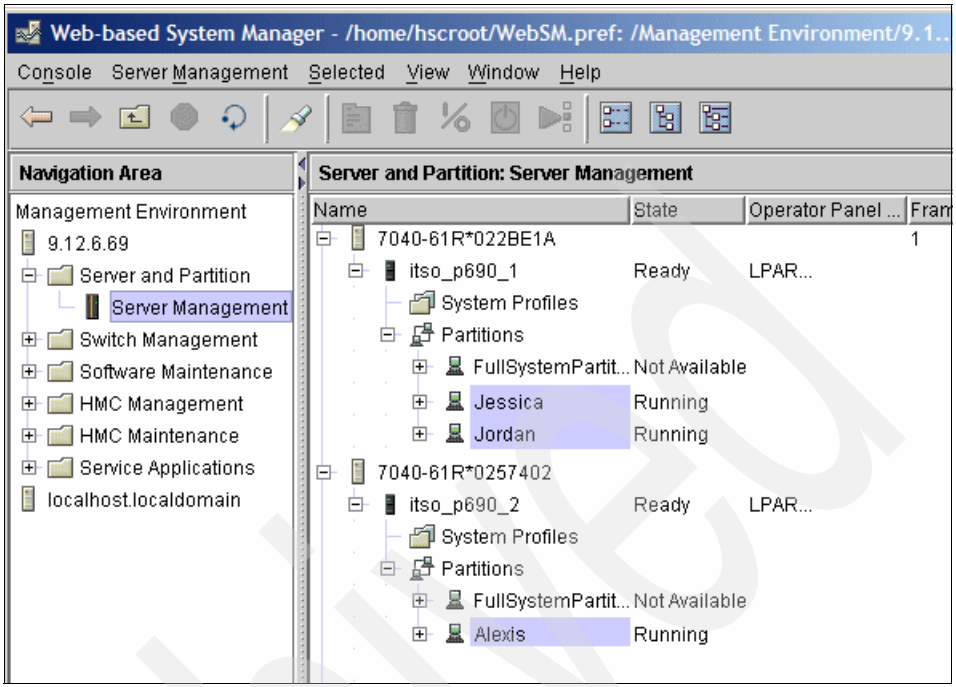

Figure 10-5 Test LPARs

These LPARs are configured with the partition profile settings shown in Table 10-5:

Table 10-5 Partition profile settings

LPAR Name	Minimum	Desired	Maximum
Jordan	1CPU - 1GB	1CPU - 1GB	4CPU - 4GB
Jessica	1CPU - 1GB	1CPU - 1GB	2CPU - 2GB
Alexis	1CPU - 1GB	1CPU - 1GB	6CPU - 6GB

We have two resource groups configured, app1_rg and app2_rg each containing their own corresponding application servers of app1 and app2 respectively. Each resource group is configured as online on home node. App1_rg has participating nodes of Jordan and Alexis. App2_rg has participating nodes of Jessica and Alexis. Making our cluster a 2+1 setup, with node Alexis as the standby node.

We have configured HMC communications for each node to include both HMCs with IP addresses of 192.168.100.69 and 192.168.100.5.

The application server DLPAR configuration settings are shown in Table 10-6 on page 488:

Table 10-6 Application server DLPAR settings

App Server	Minimum	Desired
app1	0	3
app2	0	2

We specifically chose the minimum settings of zero to always allow our resource group to be acquired.

10.1.6 Test results

Scenario 1 - Resource group acquisition

In this scenario we start off with the following:

- Jordan has 1 CPU/1GB allocated
- Free pool has 7 CPU/7GB

Upon starting cluster services on Jordan, app1 is started locally and attempts to acquire the desired amount of resources assigned. Since there are enough resources in the free pool, another 3 CPUs and 3 GB are acquired as shown in Figure 10-6.

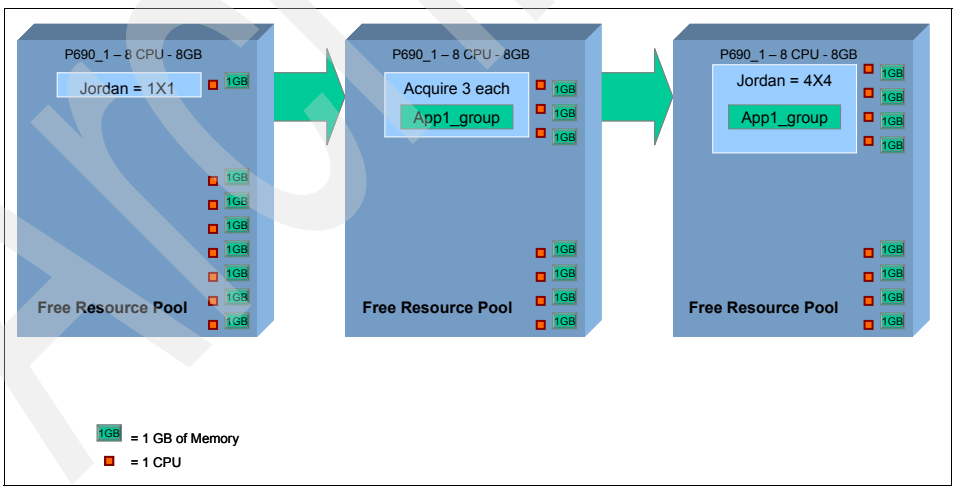

Figure 10-6 DLPAR Resource acquisition

Scenario 2 - Resource group release

In this scenario node Jessica is online in the cluster with app2 running on its partition maximum settings of 2 CPUs and 2 GB.

Upon stopping cluster services on Jessica, app2 is stopped locally and releases resources back to the free pool. When releasing resources, HACMP will not release more resources than it originally acquired.

In Figure 10-7, we show the releasing of resources and their re-allocation back to the free pool.

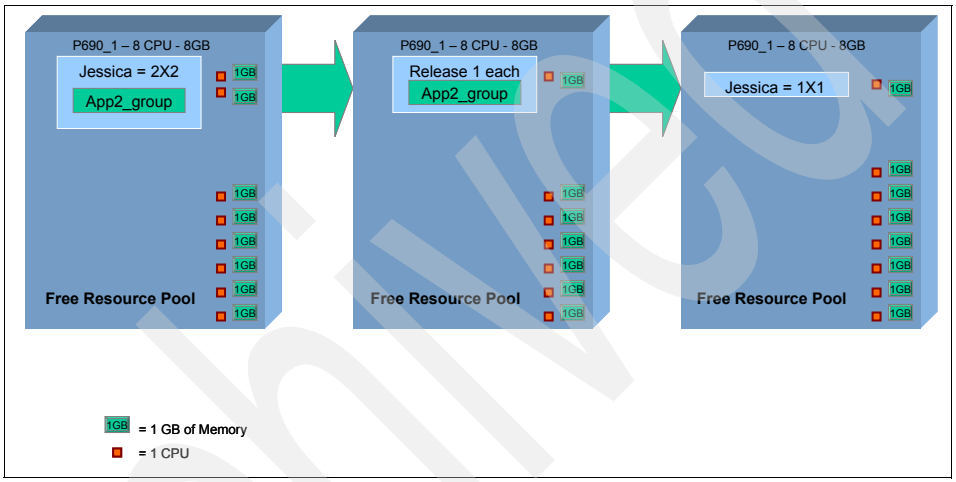

Figure 10-7 DLPAR resource release

Scenario 3 - Fallover each LPAR sequentially

This is a two part scenario that we will go through falling over each partition, node Jordan and then node Jessica. This demonstrates how resources are acquired on fallover, similarly to local resource group acquisition.

Also between this scenario and Scenario 4 - Fallover production LPARs in reverse order, we show how each individual fallover differs in the total amount of resources the standby node.

For the first part, we fail node Jordan by executing **`reboot -q`**. This results in a fallover to occur to node Alexis. Alexis acquires the app1 resource group and allocates the desired amount of resource as shown in Figure 10-8 on page 490.

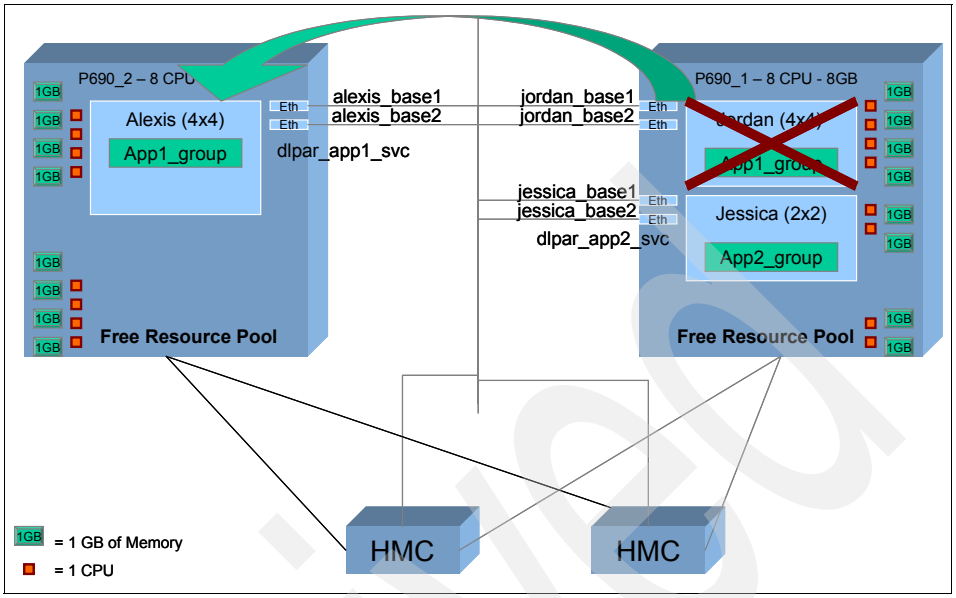

Figure 10-8 First production LPAR fallover

Node Alexis now has the same amount resources as the original failing node.

The second part of this scenario we continue on from the following:

- Jordan is offline
- Jessica has 2 CPUs and 2GB memory
- Alexis has 4 CPUs and 4GB memory
- Free pool (frame 2) has 4 CPUs and 4 GB memory

We now fail node Jessica via **`reboot -q`**. Node Alexis takes over the app2 resource group and acquires the desired resources as shown in Figure 10-9 on page 491.

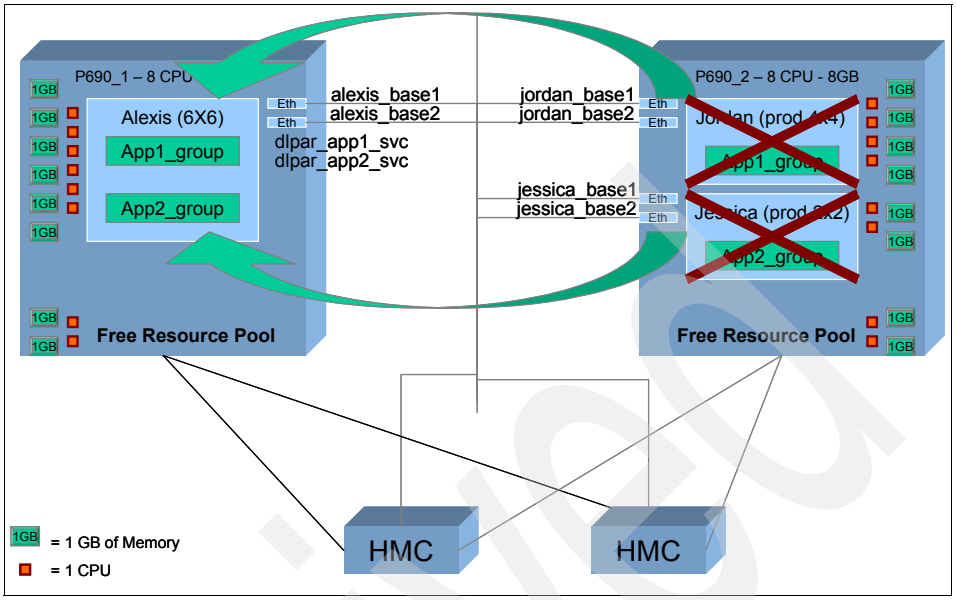

Figure 10-9 Second production LPAR fallover

Alexis ends up with its maximum partition setting of 6 CPUs and 6 GB memory.

Scenario 4 - Fallover production LPARs in reverse order

This is also a two part scenario. We start off exactly the same way we did with scenario 3 as follows:

We start off with cluster services running on all nodes and they currently have the following resources assigned to each:

- Jordan (frame 1) has 4 CPUs and 4GB memory
- Jessica (frame 1) has 2 CPUs and 4GB memory
- Alexis (frame 2) has 1 CPU and 1 GB memory
- Free pool (frame 2) has 7 CPUs and 7 GB memory

This time the we fail node Jessica first via our preferred method of **`reboot -q`**. This results in node Alexis acquiring the app2 resource group and the desired amount of resources as shown in Figure 10-10 on page 492.

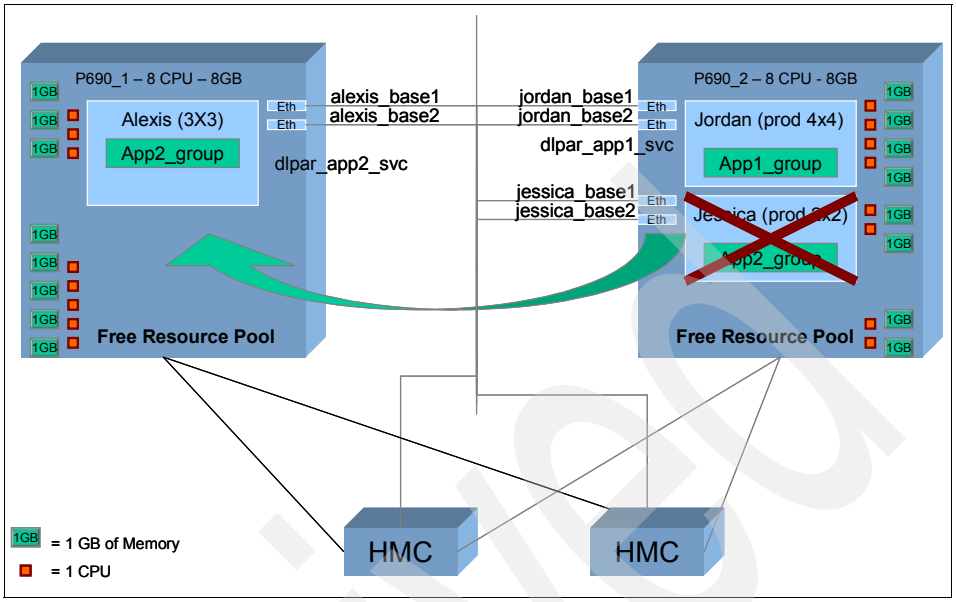

Figure 10-10 Fallover second production LPAR first

The point to note here is now Alexis has 3 CPUs and 3GB memory. Normally the app2 resource group only has 2 CPUs and 2GB memory on node Jessica. Technically this may be more resources than necessary. This is a direct result of how application provisioning can end up with a different amount of resources depending on which LPAR/partition profile the resource group ends up on.

The second part we continue from here with:

- Jordan (frame 1) has 4 CPUs and 4GB memory
- Jessica offline
- Alexis (frame 2) has 3 CPU and 3GB memory
- Free pool (frame 2) has 5 CPUs and 5 GB memory

We now fail node Jordan via **`reboot -q`**. Node Alexis takes over the app1 resource group and acquires the desired resources as shown in Figure 10-11 on page 493. The end result is exactly the same as the previous scenario.

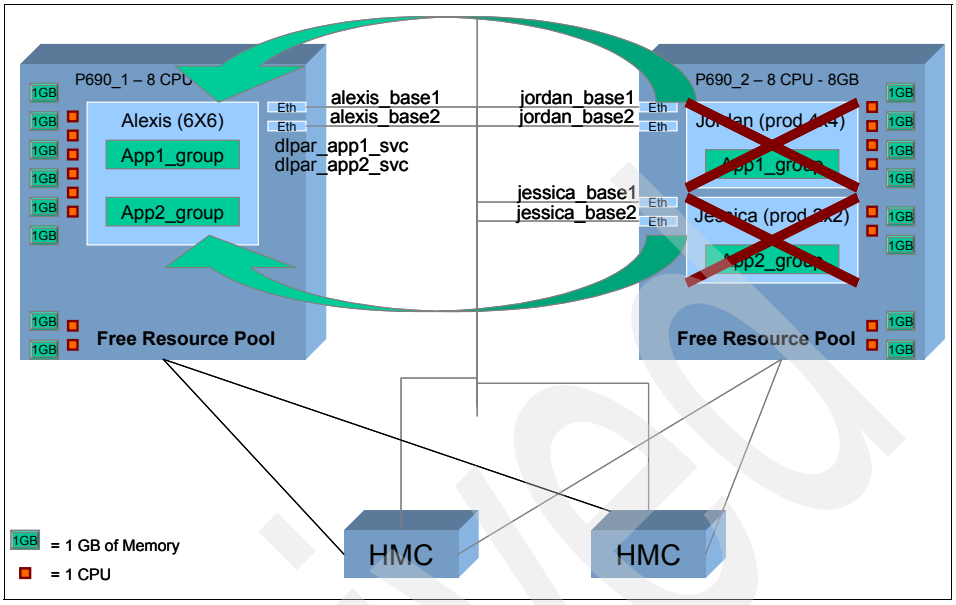

Figure 10-11 Results after second fallover

Scenario 5 - Production re-acquisition via rg_move

In this scenario we continue on from where we left off at in scenario 4 after restarting nodes Jordan and Jessica back into the cluster. We start with the following:

- Jordan (frame 1) has 1 CPU and 1 GB memory
- Jessica (frame 1)has 1 CPU and 1 GB memory
- Alexis (frame 2) has 6 CPUs and 6 GB memory
- Free pool (frame 1) has 6 CPUs and 6 GB memory

Node Alexis is currently hosting both resource groups. We run an rg_move of app2_rg from node Alexis back to its home node of Jessica. Alexis releases 2 CPUs and 2GB memory, while Jessica *only* acquires 1 CPU and 1 GB of memory as shown in Figure 10-12 on page 494. This again is a direct result of the combination of application provisioning and the LPAR profile settings.

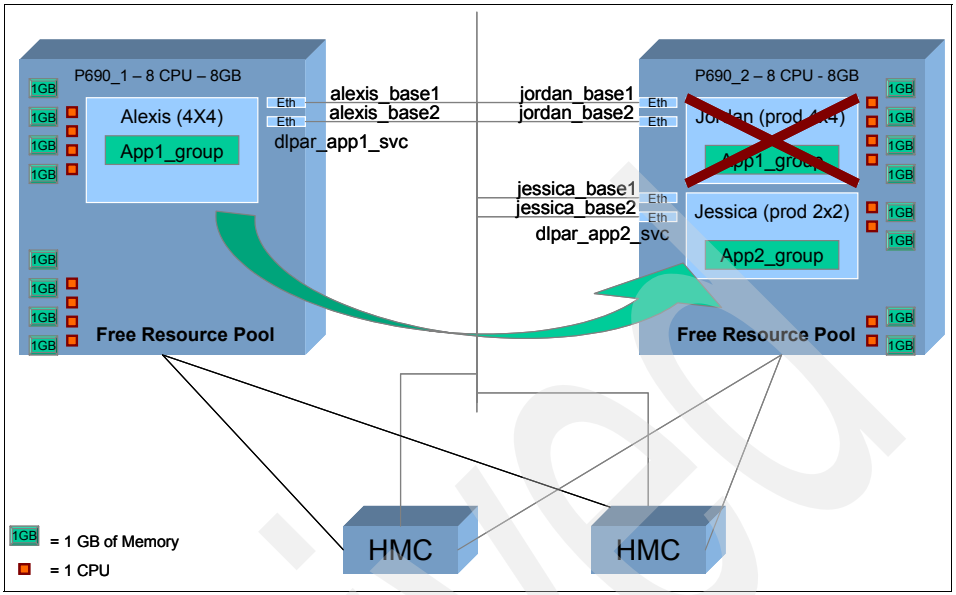

Figure 10-12 Resource group release and acquisition from rg_move

Scenario 6 - Test HMC Redundancy

In this scenario we test the HMC redundancy by physically unplugging the network connection of one of the HMCs. We start off with cluster services running on all nodes and they currently have the following resources assigned to each:

- Jordan (frame 1) has 4 CPUs and 4GB memory
- Jessica (frame 1) has 2 CPUs and 4GB memory
- Alexis (frame 2) has 1 CPU and 1 GB memory
- Free pool (frame 2) has 7 CPUs and 7 GB memory

We physically pulled the ethernet cable from the HMC we have listed first of 192.168.100.69. We then failed node Jordan to cause a fallover to occur. We show this in Figure 10-13 on page 495.

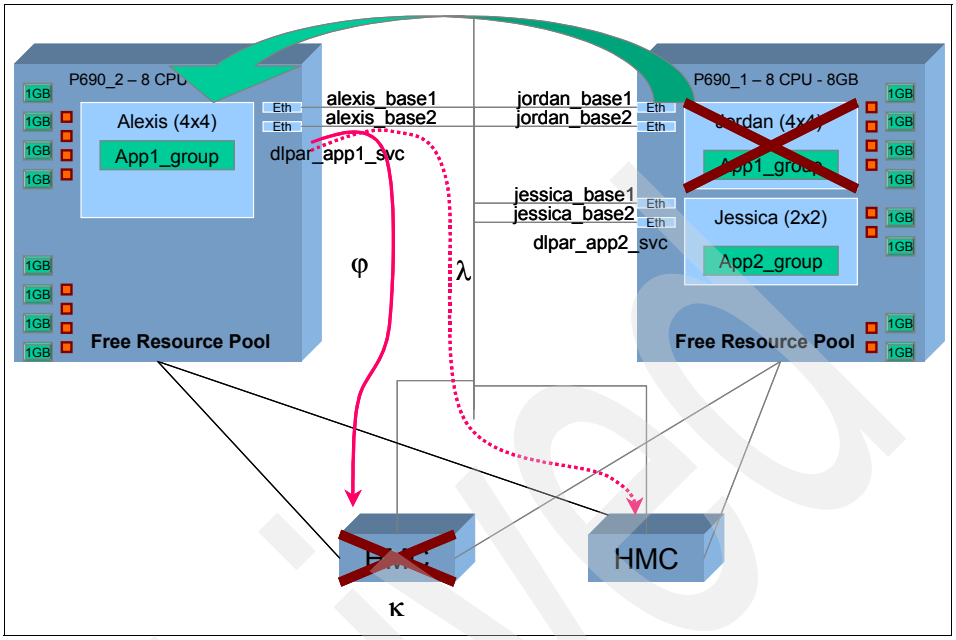

Figure 10-13 HMC redundancy test

During fallover and trying to access the HMC(s) the following occurs:

1. HACMP issues ping to first HMC
2. First HMC is offline and does not respond
3. HACMP issues ping to second HMC which is successful and continues to process DLPAR command line operations.

The HMC test actions can be seen in /tmp/hacmp.out via the event utility *clhmcexec*.

10.2 HACMP and virtualization

During writing, the formal support announcement of HACMP and virtualization was released. While we do cover the support details in the following sections, the announcement can be found here:

http://w3-1.ibm.com/sales/systems/portal/_s.155/254?navID=f220s240&geoID=All&prodID=IBM%20eServer%20And%20TotalStorage%20Products&docID=hacmpvio063005

10.2.1 Requirements

To use the integrated virtualization functions, and/or CUoD, of HACMP on Power5, all LPAR nodes in the cluster should have at *least* the following levels installed:

- AIX 5.3 Maintenance Level 5300-002 with APAR IY70082 and eFIX IY72974.
- HACMP
  5.1 with APAR IY66556 (or higher).
  5.2 with APAR IY68370 (or higher) and APAR IY68387.
  5.3.0.1with APAR IY73051 for ... support.
- RSCT
  rsct.basic.hacmp.2.4.2.1
  rsct.basic.rte.2.4.2.2
  rsct.compat.basic.hacmp.2.4.2.0
- OpenSSH 3.4p1
  The OpenSSH software can be obtained from any of the following sources, AIX 5.3 expansion pack, Linux Toolbox CD, downloaded from:

  `http://sourceforge.net/projects/openssh-aix`

  OpenSSh for AIX has its own prerequisites cf. 10.1.Requirements
- Virtual I/O Server Version 1.1.2 with VIOS fixpack 6.2 and eFIX IY71303.062905.epkg.Z
  Fixpack 6.2 is available at:

  `http://techsupport.services.ibm.com/server/vios/download/home.html`

  eFIX IY72974 is available at ftp.software.ibm.com/.../efixes
- OpenSSL SSL Cryptographic Libraries (OpenSSL)

  `http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html`

HMC attachment to the LPARs is required for proper management and DLPAR capabilities. The HMC must also have at least the following levels installed:

- HMC Version 4 Release 5 Build 20050519.1 or greater.

Important: APAR IY73051 for HACMP V5.3 is required to support micropartitioning, CUoD and CBU functionality on Power5 systems.

10.2.2 Application provisioning

All listed considerations in “Application provisioning” on page 32 are also available for the micropartitioning configuration.

The granularity of what HACMP can manage is the physical CPU for dedicated processors, and virtual processors for shared processors configuration. You can see some scenarios of what you can do below.

With micropartitioning, HACMP works with virtual processors, and not physicals processors. In a mixed environment, HACMP does the verification of the possibility to add the resources to respect the maximum value. For dedicated processor mode the maximum value unit is the physical processor. For shared processor mode, the maximum value unit is the virtual processor (Figure 10-14). The free pool resources is calculated by adding the desired capacity entitlement (CE) values of each partition in shared processor mode, and the physical processor in dedicated processor mode.

You can't use capacity entitlement (10th processor precision) as value to define application provisioning in HACMP menu.

HACMP does not verify if the partition is capped or uncapped. Depending on set up, you have to verify all cases.

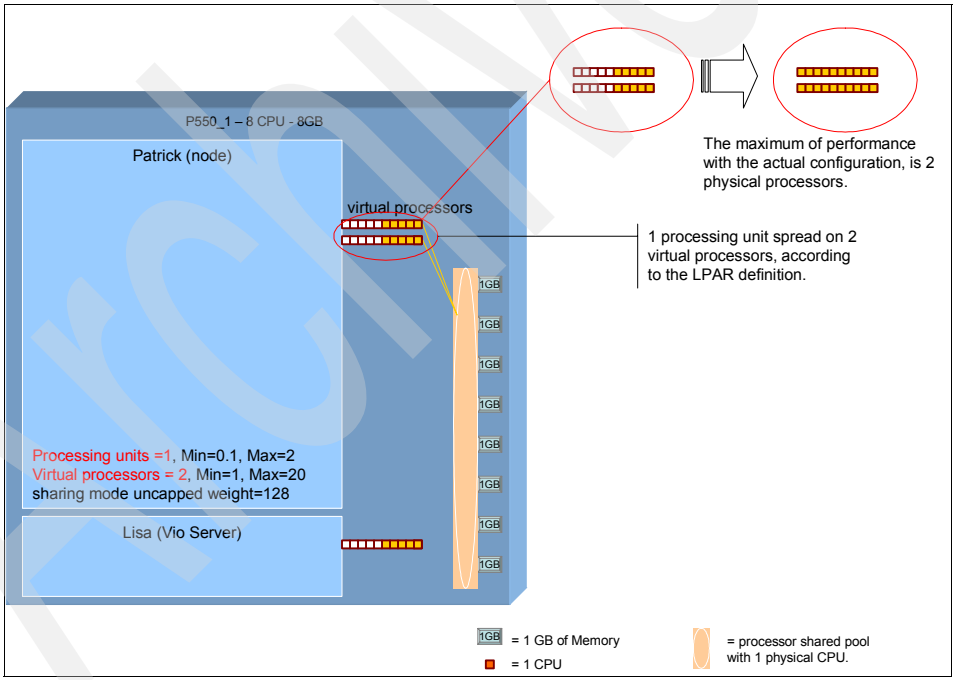

Figure 10-14 The attribution of capacity entitlement to a partition

The maximum value for one virtual processor is one physical processor (Figure 10-14). So the number of virtual processors attributed to a partition, determine the maximum of processing power that you can have in this partition.

That's why it could be a good idea to use virtual processor with HACMP. Indeed instead of configuring the number of virtual processors in the partition, for the worse case (lots of applications fall over on one node). In this case the number of virtual processors is not optimized (more processing consumption for hypervisor).

If you are on a capped environment you have to anticipate the number of virtual processors you can have in the partition to have the right performance for your application. Indeed, if you define a CE of 1 and you create 4 virtual processors, you have always 1 CE attributed, so the equivalent of 1 physical processor not 4. In an uncapped environment you don't have this problem (Figure 10-15). The only way to limit the partition is the number of virtual processors, and the weight.

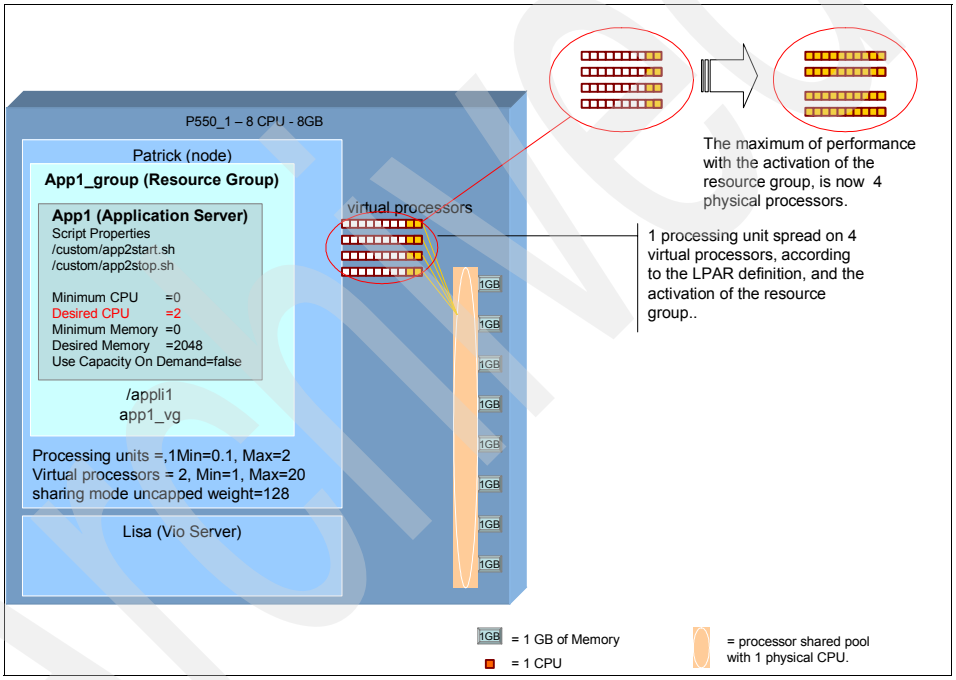

Figure 10-15 Resource group activation

Here is some example with micropartitioning to understand how it works. We take only the processors information. In this example the minimum values for HACMP application provisioning is 0. We recommend this value to be 0 to start the application server even if we haven't got the resources available.

Table 10-7 LPAR parameter value

LPAR name	LPAR values min/desired/max	Virtual processors min/desired/max	HACMP values	Processor	Processor mode
patrick	0.1 / 1.0 / 2.0	1 / 2 / 20	app1 / 0 / 2		shared
lisa	0.1 / 0.5 / 2.0	1 / 2 / 20	N/A		shared
maelle	1.0 / 1.0 / 1.0	N/A	N/A	1	dedicated
shawn	0.1 / 0.3 / 4.0	1 / 3 / 40	app2 / 0 / 1		shared
lee	0.1 / 0.5 / 2.0	1 / 2 / 20			shared
smokey	0.1 / 0.5 / 1.0	1 / 1 / 10			shared

Table 10-8 displays real active values.

Table 10-8 Value used for calculation

LPAR name	LPAR values ①	Virtual processors min/desired/max	HACMP values	Processors
patrick	1	2	app1 / 0 / 2	
lisa	0.5	2	N/A	
maelle	1.0	N/A	N/A	1
shawn	0.3	3	app2 / 0 / 1	
lee	0.5			
smokey	0.5	1		

Note: These are sample calculations with the software version available at the time of writing this material. Check for PTF1 before you actually implement this type of configuration in production.

Free resources value is subtract 2.8 (the sum of column ① for one machine) from the total number of CPU on the machine.

Free resource on patrick machine: 4 - 2.8 = 1.2

Free resource on shawn machine: 4 - 1.3 = 2.7

Calculation with one resource on node Patrick:
1 VP (minimum LPAR) + 2 VP(AS desired) = 3 VPs actives when app2 is on.

Calculation to add one moreover application app1:
1 VP (desired AS) + 3 VPs (current activity) - 1VP (minimum LPAR) - 2VP = 1 VP to add. If this value is lower than free resources (1.2) then HACMP adds it.

Attention: If you want the application to start every time even if you don't have enough resources on the target machine, you have to put 0 as the minimum value on the resource (CPU and memory) in the menu of application server DLPAR definition.

This is applicable for DLPAR with dedicated mode processor.

HACMP code permits to adjust the LMB size to the optimal size. The LMB size is gotten from HMC. But smit menu is capped to the only increment of 256 MB.

The free resources is calculated by addition of the number of dedicated processors and the capacity entitlement specified as desired for each LPAR configured.

10.3 HACMP and virtualization configuration scenarios

The first is designed to assist you in basic configuration of HACMP in a virtual environment.
The second is designed to bring out the functionalities of virtualization. These are complementary to the high availability mechanism that offer HACMP.

- Scenario 1 - Configuration with two HACMP nodes in mutual takeover. Shows a basic configuration of HACMP.
- Scenario 2 - Configuration with two HACMP clusters including two nodes each.

10.3.1 Scenario 1

HACMP and virtual component outlook.

HACMP could be used as usual in an environment virtualized. Of course you have to respect the above-mentioned constraints.

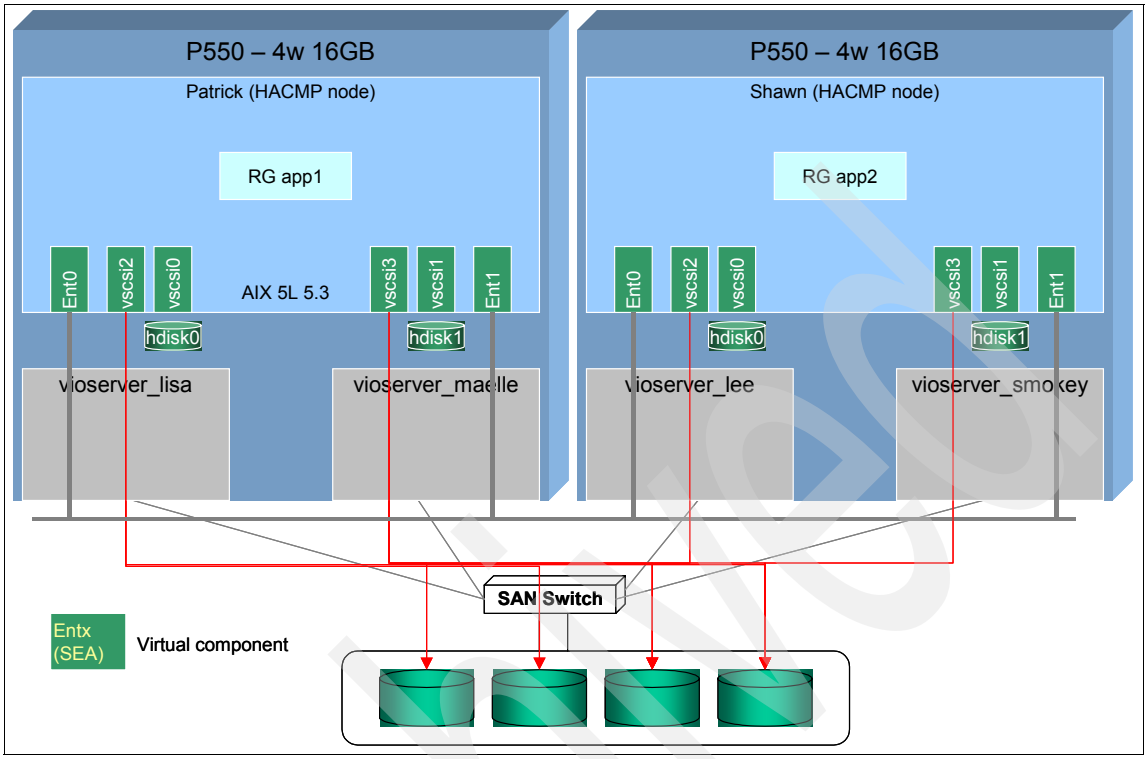

Figure 10-16 HACMP logical diagram outlook

The diagram in Figure 10-16 shows the components as seen by HACMP. For redundancy, we define two Virtual I/O server per machine. The AIX client's partition points through virtual adapter to the VIOS partition. When you use the virtual scsi to access the disks (vscsi0, vscsi1, ...) automatically multi path I/O (MPIO) is used. That's why shared disks are available by two path. The **`lspath`** command shows you the path managed by MPIO (Figure 10-17 on page 502). There is no hardware errors reported to AIX by the virtual adapter. But HACMP uses RSCT to communicate the status of all communicating components.

```
patrick / > lspath
Enabled hdisk1 vscsi1
Enabled hdisk3 vscsi2
Enabled hdisk4 vscsi2
Enabled hdisk5 vscsi2
Enabled hdisk6 vscsi2
Enabled hdisk6 vscsi3
Enabled hdisk5 vscsi3
Enabled hdisk4 vscsi3
Enabled hdisk3 vscsi3
Enabled hdisk2 vscsi0
Enabled hdisk0 vscsi0
```

Figure 10-17 lspath command

Example 10-10 shows the HACMP configuration in our test cluster.

Example 10-10 Using cldump utility

```
patrick / > cldump
_____________________________________________________________________________
Cluster Name: app2_cluster
Cluster State: UP
Cluster Substate: UNSTABLE
_____________________________________________________________________________

Node Name: patrick              State: UP

  Network Name: net_diskhb_01       State: UP

    Address:                  Label: patrick2shawn      State: UP

  Network Name: net_ether_01        State: UP

    Address: 10.10.5.2        Label: patrick_base1      State: UP
    Address: 10.10.6.2        Label: patrick_base2      State: UP
    Address: 192.168.101.143 Label: vio_svc2            State: UP

Node Name: shawn                State: UP

  Network Name: net_diskhb_01       State: UP

    Address:                  Label: shawn2patrick      State: UP

  Network Name: net_ether_01        State: UP

    Address: 10.10.5.1        Label: shawn_base1        State: UP
    Address: 10.10.6.1        Label: shawn_base2        State: UP
```

```
    Address: 192.168.101.142 Label: vio_svc1          State: UP

Cluster Name: app2_cluster

Resource Group Name: app2_group
Startup Policy: Online On Home Node Only
Fallover Policy: Fallover To Next Priority Node In The List
Fallback Policy: Fallback To Higher Priority Node In The List
Site Policy: ignore
Priority Override Information:
    Primary Instance POL:
Node                          Group State
----------------------------- ---------------
shawn                         ONLINE
patrick                       OFFLINE

Resource Group Name: app1_group
Startup Policy: Online On Home Node Only
Fallover Policy: Fallover To Next Priority Node In The List
Fallback Policy: Fallback To Higher Priority Node In The List
Site Policy: ignore
Priority Override Information:
    Primary Instance POL:
Node                          Group State
----------------------------- ---------------
patrick                       ONLINE
shawn                         OFFLINE
```

The virtual I/O (VIO) server has to be configured to offer the virtualized components to the partition.

Here is on the next diagram (Figure 10-18 on page 504) the details of hardware components that were used. Thanks to the legend, you can distinguish the virtual components from physical components. Note that in our test AIX and HACMP are based only on virtual components. On each node we use four virtual SCSI, two of which for the mirrored system disk, and two for the shared disks. All of the access are spread on two VIO servers.

The shared disks for both clients partitions have to be defined as hdisk on the target definition in the VIO server.

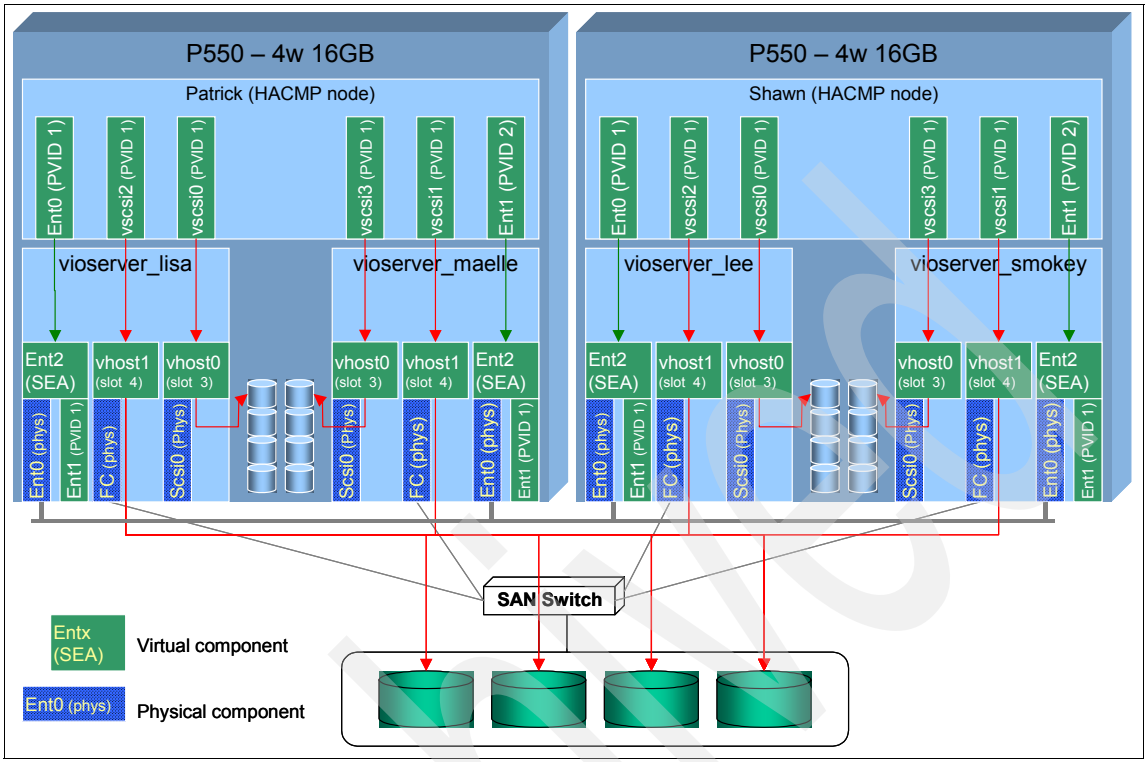

Figure 10-18 Virtualization architecture diagram

Virtualization set up

First of all, as defined in the redbook *Advanced POWER Virtualization on IBM @server p5 Servers: Introduction and Basic Configuration,* SG24-7940, you have to plan your operations. Here are the operations that we have done to implement our virtualization configuration:

1. Planning the operations. See Figure 10-19 on page 505.
2. Create the LPAR configuration on the HMC.

   Each VIOS partition is defined with one virtual ethernet and two virtual scsi. The virtual ethernet is defined as trunk adapter (the future ent1). Both virtual scsi are defined as server (the future vhost0 and vhost1).
3. Install the VIOS partition and then update it with the **`updateios`** command.

Machine	Client Name	type SCSI or Ethernet	intrface id	Slot		remote partition	Remote Partition Vslot/PVID	VIOS name	type SCSI or Ethernet	interface id	Slot number	remote partition	remote slot / PVID
p550-1	**patrick**	ent	0	2	10.10.5.2		1	**lisa**	ent	4	6		1
p550-1	**patrick**	ent	1	4	10.10.6.2		2	**maelle**	ent		2		2
p550-1	**patrick**	scsi		7		maelle	7	**maelle**	scsi		7	patrick	7
p550-1	**patrick**	scsi		6		lisa	4	**lisa**	scsi		4	patrick	6
p550-1	**patrick**	scsi		5		maelle	5	**maelle**	scsi		5	patrick	5
p550-1	**patrick**	scsi		3		lisa	3	**lisa**	scsi		3	patrick	3
p550-2	**shawn**	ent	1	4	10.10.6.1		2	**smokey**	ent		2		2
p550-2	**shawn**	ent	0	2	10.10.5.1		1	**smokey**	ent		3		1
p550-2	**shawn**	scsi		7		smokey	7	**smokey**	scsi		7	shawn	7
p550-2	**shawn**	scsi		5		smokey	5	**smokey**	scsi		5	shawn	5
p550-2	**shawn**	scsi		6		lee	6	**lee**	scsi		6	shawn	6
p550-2	**shawn**	scsi		3		lee	3	**lee**	scsi		3	shawn	3

Figure 10-19 The excel tool to plan the Virtualization operations.

4. Set up the shared ethernet adapter (SEA) - see Example 10-11

Example 10-11 Creating the shared ethernet adapter

```
mkvdev -sea ent0 -vadapter ent1 default ent1 -defaultid 1

mktcpip -hostname vioserver_lisa -inetaddr 192.168.100.220 -netmask
255.255.255.0 -gateway 192.168.100.60 -start
```

5. Create the virtual disks.
   In our cluster we create two types of disks for the client. On node shawn we create both target disks on VIOS partition as logical volume (example 10-2). On node patrick we create both target disks on VIOS partition as hdisk (Example 10-12).

Example 10-12 Creating the virtual disks devices

```
mkvg -f -vg shawnvg hdisk1
mklv -lv shawn_lv shawnvg 30G
mkvdev -vdev shawn_lv -vadapter vhost0 -dev vshawn_disk
mkvdev -vdev hdisk1 -vadapter vhost0 -dev vpatrick_disk
```

6. Install the system and HACMP in the client partition.
7. Mirror the system disks, and change the bootlist.
8. If you want to use DLPAR or CUoD function, install and configure SSH as described in “Install and configure SSH on HACMP nodes” on page 474.

Test results

We performed the following tests and obtained these test results:

rootvg mirroring

By doing a shutdown of VIOS, the access on one part of the disk is inoperative. The mirror enables to continue working. At the level of AIX that we have, to reestablish the situation when disks are available, you have to extract the disk from the rootvg definition and then add the same disk and do the mirroring. Then you have to run the **bosboot** command.

adapter swap

HACMP uses RSCT to track the state of the communication interfaces or devices. Figure 10-20 shows the identification of a failure on a virtual adapter. In this test we disconnect an ethernet cable to simulate a physical network failure.

```
07/05 18:32:45.542: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:33:05.542: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:33:25.548: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:33:40.789: Heartbeat was NOT received. Missed HBs: 1. Limit: 10
07/05 18:33:42.789: Heartbeat was NOT received. Missed HBs: 2. Limit: 10
07/05 18:33:42.789: Starting sending ICMP ECHOs.
07/05 18:33:42.789: Invoking netmon to find status of local adapter.
07/05 18:33:44.789: Heartbeat was NOT received. Missed HBs: 3. Limit: 10
07/05 18:33:46.789: Heartbeat was NOT received. Missed HBs: 4. Limit: 10
07/05 18:33:47.497: netmon response: Adapter seems down
...
07/05 18:33:58.809: Heartbeat was NOT received. Missed HBs: 10. Limit: 10
07/05 18:33:58.809: Local adapter is down: issuing notification for local adapter
07/05 18:33:58.809: Adapter status successfully sent.
07/05 18:33:58.945: Dispatching netmon request while another in progress.
07/05 18:33:58.945: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:33:59.497: netmon response: Adapter seems down
07/05 18:33:59.497: Adapter status successfully sent.
07/05 18:41:42.982: netmon response: Adapter is up
07/05 18:41:42.982: Bind to broadcast address succeeded.
07/05 18:41:42.982: Adapter status successfully sent.
07/05 18:41:45.849: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:46.859: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:46.859: Received a STOP MONITOR command.
07/05 18:41:46.861: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:46.861: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:46.861: Received a START HB command. Destination: 10.10.5.2.
07/05 18:41:46.861: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:46.861: Received a START MONITOR command.
07/05 18:41:46.861: Address: 10.10.5.2 How often: 2000 msec Sensitivity: 10 Configuration Instance: 11
07/05 18:41:46.862: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:46.862: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:47.660: netmon response: Adapter is up
07/05 18:41:51.871: Received a SEND MSG command. Dst: 10.10.5.2.
07/05 18:41:57.871: Received a SEND MSG command. Dst: 10.10.5.2.
```

Figure 10-20 The nim.topsvcs.en2.app2_cluster

In this example, be fore 18:33 the network connection was available. At 18:33:30 we disconnect the ethernet cable. The topology services identifies by missed heartbeat, that the adapter en2 is down at 18:33:59. At 18:34:02 HACMP does a **swap_adapter**.

A VIO server down

In this test the `vioserver_lisa` is down. All the operation continue to work by the second path. MPIO does its work by preserving the access to the data by the second VIO server `vioserver_maelle`. Of course rootvg is always active on the mirrored disk (Figure 10-21). The **lspath** command shows that one path is failed for the shared disks access.

As described in “rootvg mirroring” on page 506, when the VIO server is up, you have to perform the operation manually. The **lspath** command shows you the status of each path (Figure 10-13 on page 508).You can force the try with the **chpath** command or by **smit mpiopath_enable_all**, here is an example after running this command on Figure 10-21. The virtual ethernet adapter is joined automatically by the **join_interface** event.

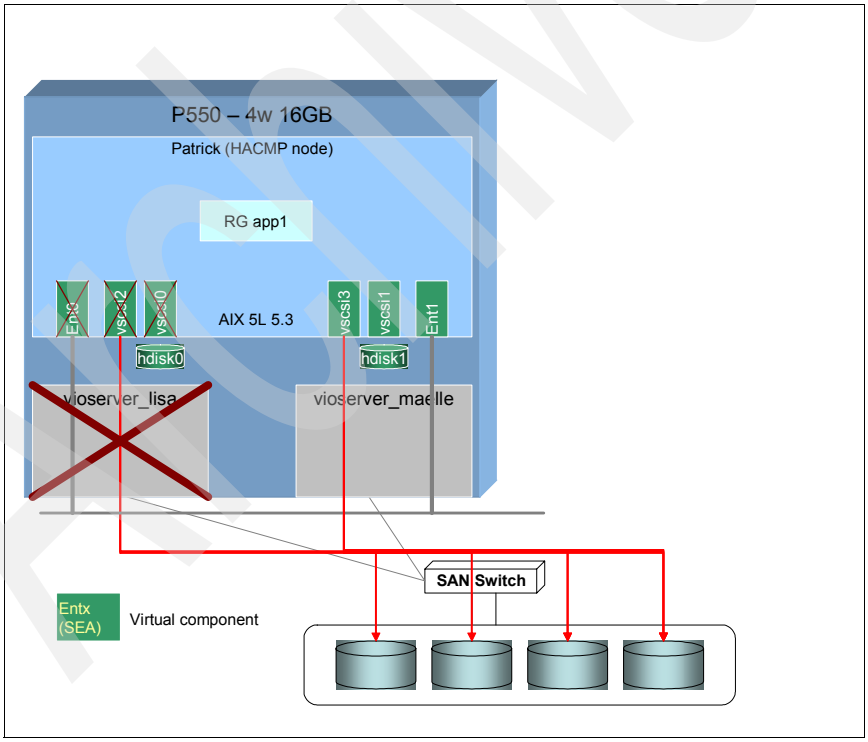

Figure 10-21 VIO server failure

The following Example 10-13 displays the missing path information for the virtual disks:

Example 10-13 lspath command with path missing

```
patrick / > lspath
Enabled hdisk1 vscsi1
Missing hdisk3 vscsi2
Missing hdisk4 vscsi2
Missing hdisk5 vscsi2
Missing hdisk6 vscsi2
Failed  hdisk6 vscsi3
Failed  hdisk5 vscsi3
Failed  hdisk4 vscsi3
Failed  hdisk3 vscsi3
Enabled hdisk2 vscsi0
Missing hdisk0 vscsi0
```

When one VIO server is down, the lspath command looks similar to the output presented in Example 10-14.

Example 10-14 A result example of lspath command when one VIO server is down.

```
patrick / > lspath
Enabled hdisk1 vscsi1
Missing hdisk3 vscsi2
Missing hdisk4 vscsi2
Missing hdisk5 vscsi2
Missing hdisk6 vscsi2
Enabled hdisk6 vscsi3
Enabled hdisk5 vscsi3
Enabled hdisk4 vscsi3
Enabled hdisk3 vscsi3
Enabled hdisk2 vscsi0
Missing hdisk0 vscsi0
```

One node fallover

The final HACMP behavior is the same for these tests:

- Two VIO servers down
- Each ethernet adapter disconnected from both VIO servers.
- Halt of one node

HACMP has finished its logical processing.

10.3.2 Performance and architecture considerations (scenario 2)

We try to go further and then add another cluster on the same infrastructure. This scenario include two HACMP clusters. We define two virtual scsi servers moreover (Figure 10-22).

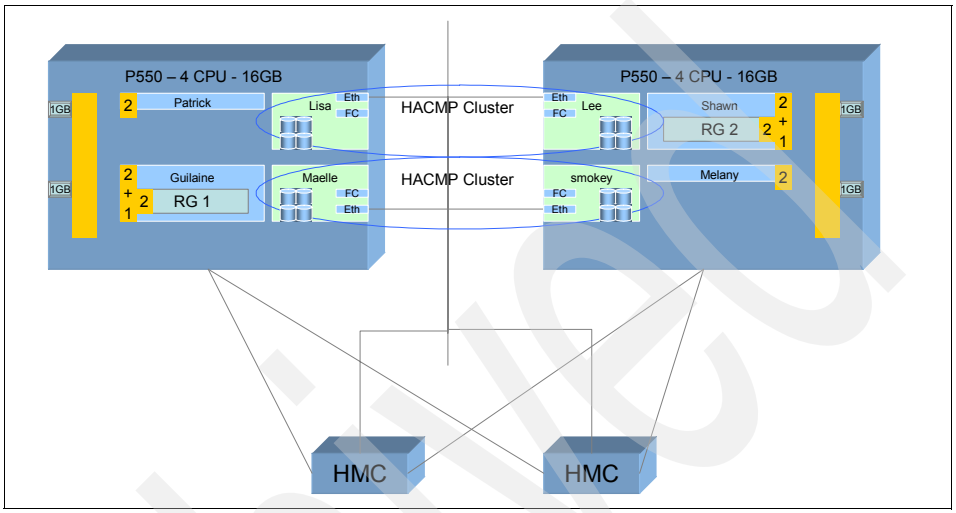

Figure 10-22 Two clusters with two CECs and two HMCs

The advantage of this kind of architecture is that you manage two simple clusters with two machines. Indeed, the principal of having a standby node to secure the production is good to limit the risk to break it. Each application has its own cluster.The administration is simplified, so you master better your environment. By configuring the virtualization you can manage the application provisioning. To configure this cluster by using HACMP config assist, in no time it is done.

The resource group RG1 is the production that need more CPU resource. By configuring the uncapped weight value on each partition you can support an environment compared to the other. HACMP activates the resources associated with the application server (the number of virtual processors). The calculation mechanism is the same as we have presented in 10.1.2, “Application provisioning” on page 464.

In this configuration we use two functionalities, one from HACMP with the application provisioning feature and second with the micropartitioning feature that comes with Advance Power Virtualization (APV). Instead of implementing a stop of another partition to free some resources, we use the uncapped weight feature, to leave the priority to the most important partition.

The micropartitioning parameters (shown in Table 10-9) are independent of the HACMP parameters.

Table 10-9 Parameters list of cluster configuration

LPAR name	CE min/desired/max	HACMP DLPAR min/desired	uncapped weight	Virtual Processors min/desired/ax
Patrick	0.5 / 2.0 / 4.0	app2 0 / 2	200	1 / 2 / 40
Guilaine	0.5 / 2.0 / 4.0	app1 0 / 2	10	1 / 2 / 40
Maelle	0.1 / 0.5 / 1	N/A	128	1 / 2 / 10
Lisa	0.1 / 0.5 / 1	N/A	128	1 / 2 / 10

Here is the test that we do. We fall over one node to the other one. At the same time we take some collect of performance on each partition. Uncapped weight value of 200 for LPAR patrick and 10 for LPAR guilaine, give more priority for the partition patrick.

Uncapped weight is a number in the range of 0 through 255 that you set for each uncapped partition in the shared processor pool. 255 is the highest weight. Any available unused capacity is distributed to partitions in proportion to the other uncapped partitions.

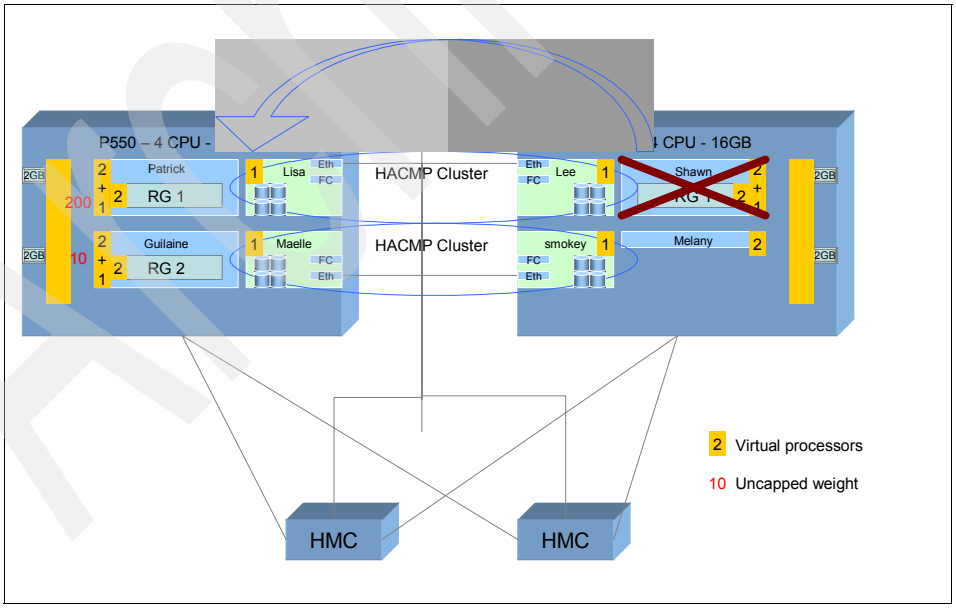

Figure 10-23 Takeover on one machine with privileged processing of RG1

The graph on Figure 10-24 represents the processing units activated on each partition on one machine of the configuration.

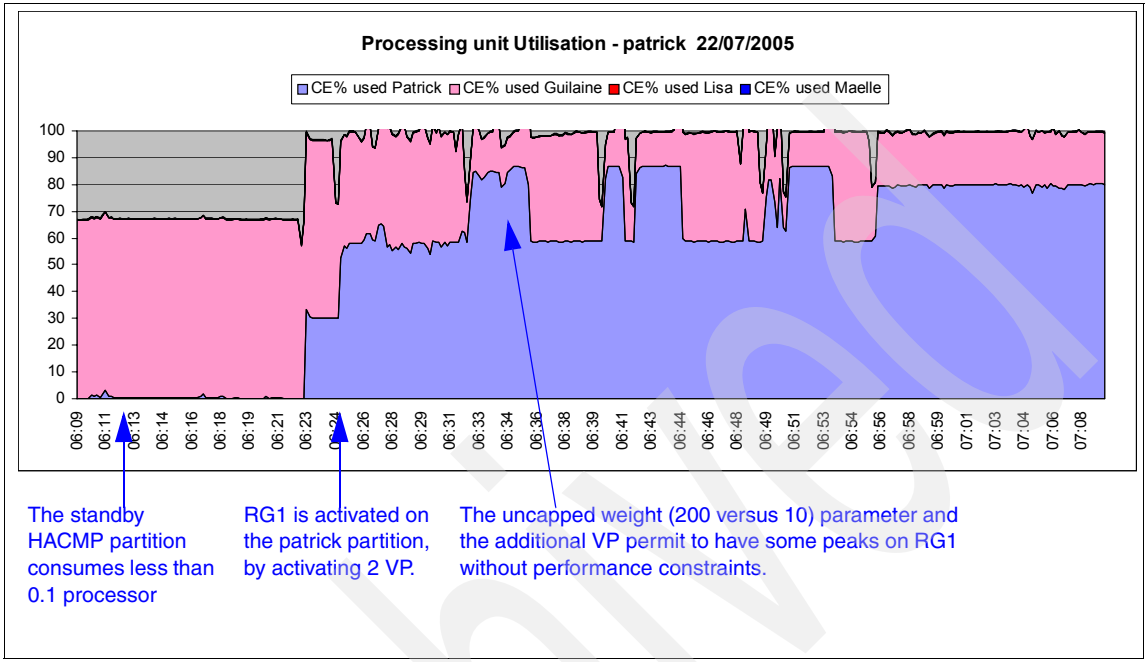

Figure 10-24 Graph of the effective processing unit activated in each partition

11

Extending resource group capabilities

In this chapter we describe — by identifying resources and defining resource group policies — how HACMP makes it possible to fulfill various individual requirements. We discuss how to configure these options and our testing experiences with these options in HACMP 5.3. The following policies and attributes dictate how a resource group's startup, fallover, and fallback behavior will be managed.

Table 11-1 Resource Group Attribute Behavior Relationships

Attribute	Startup	Fallover	Fallback
Settling Time	Yes		
Node Distribution Policy	Yes		
Dynamic Node Priority		Yes	
Priority Override Location	Yes		Yes
Delayed Fallback Timer			Yes
Resource Group Parent / Child Dependency	Yes	Yes	Yes
Resource Groups Location dependency	Yes	Yes	Yes

11.1 Settling time

The startup behavior for a resource group that is currently offline is to start up on the first available node with the highest priority that joins the cluster. The settling time feature gives you the ability to delay the acquisition of a resource group so in event that a higher priority node enters the cluster the resource group will be brought online on the higher priority node.

Settling time behavior

- If configured, it affects the startup behavior of all resource groups in the cluster for which you selected the `Online on First Available Node` startup behavior.
- The only time that this attribute is ignored is if the node joining is the highest priority node. In this situation the resource group is acquired immediately.
- If a resource group is currently in the ERROR state, HACMP will wait for the settling time period before attempting to bring the resource group online.
- The current settling time continues to be active until the resource group moves to another node of goes offline. A DARE operation may result in the release and re-acquisition of a resource group, in which case the new settling time values take effect immediately.

Configuring settling time for resource groups

. To configure a settling time for resource groups, do the following:

1. Enter `smit hacmp`
2. Select `Extended Configuration > Extended Resource Configuration > Configure a Resource Group Run-Time Policies > Configure Settling Time for Resource Group` and press Enter.
3. Enter field values as follows:

- **Settling Time (sec.**)
  Enter any positive integer number in this field. The default is zero.

  If this value is set and the node that joins the cluster is not the highest priority node, the resource group will wait the duration of the settling time interval. When this time expires, the resource group is acquired on the node which has the highest priority among the list of nodes that joined the cluster during the settling time interval.

  Remember that this is only valid for resource groups using the `Online of First Available Node` startup policy.
  Displaying the current settling time

To display the current settling time in a cluster already configured you can execute the **clsettlingtime list** command.

```
#/usr/es/sbin/cluster/utilities/clsettlingtime list
#SETTLING_TIME
120
```

During the acquisition of the resource groups on cluster startup you can also see the settling time value by running the **clRGinfo -t** command (Example 11-1).

Example 11-1 Displaying the RG settling time

```
#/usr/es/sbin/cluster/utilities/clRGinfo -t
-----------------------------------------------------------------------------
Group Name     Group State                  Node            Delayed
Timers
-----------------------------------------------------------------------------
settling_rg1   OFFLINE                      cobra          120 Seconds
               OFFLINE                      python         120 Seconds

settling_rg2   OFFLINE                      viper          120 Seconds
               OFFLINE                      python         120 Seconds
```

Note that this value will only be there during the acquisition period, it will return to blank in the **clRGinfo** output after the settling time period expires and the resource group is acquired on the appropriate node.

Settling time test scenario

In an effort to test this feature we configured a 3-node cluster using two resource groups. In our test we tried to prove two things:

1. The settling time period is enforced and the resource group is not acquired on the node startup (as long as he is not the highest priority node) until the time is expired.
2. If the highest priority node enters the cluster while in the settling time period it does not wait for the full settling time period to expire and acquires the resource group immediately.

We set out to perform our test by making the settling time a value of 10 minutes, or 600 seconds. Our two resource groups: `Settling1_RG` and `Settling2_RG` were configured to use the `Online on First Available Node` startup policy. We set the nodelist order in each one so that nodes **kaitlyn** and **thrish** failed over to node **mike**. Figure 11-1 on page 516 shows a diagram of our configuration and the sequence in which we started HACMP one node at a time to test the settling time feature.

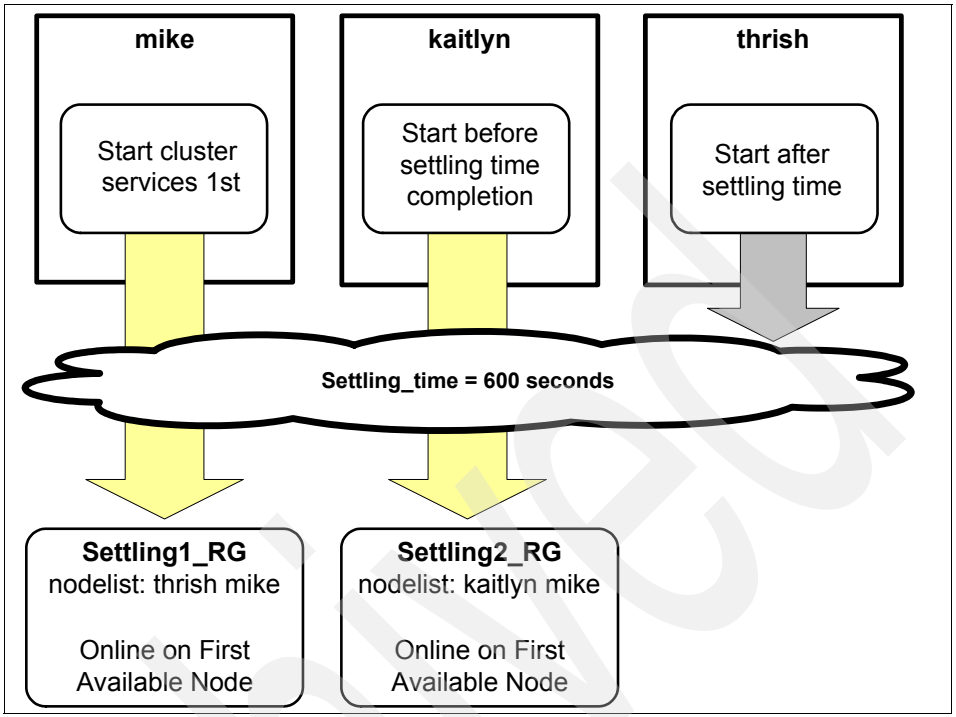

Figure 11-1 Settling time test scenario environment

Our testing comprised of the following steps:

1. With HACMP inactive on all 3 nodes, defined the settling time value to 600 seconds.
2. Synchronized the cluster.

   During the cluster verification/synchronization we logged the following:

   ```
   The Resource Group Settling time value is: 120 secs.
   The Resource Group(s) affected by the settling time are:
      settling_rg1
      settling_rg2
   ```

3. Started cluster services on node **mike**.

   We started HACMP services on this node because he was the last one on the nodelist for both of the resource groups. After the startup of the daemons neither resource group was acquired on the node. By running the **clRGinfo -t** command the 600 second settling time was visible.

   In the hacmp.out file we logged the following Example 11-2 on page 517:

Example 11-2 Checking settling time in /tmp/hacmp.out

```
#tail -f /tmp/hacmp.out
No action taken on resource group 'settling_rg1'
The Resource Group 'settling_rg1' has been configured
to use 600 seconds Settling Time. This group will be
processed when the timer expires.

No action taken on resource group 'settling_rg2'
The Resource Group 'settling_rg2' has been configured
to use 600 seconds Settling Time. This group will be
processed when the timer expires.
```

4. Started cluster services on node **kaitlyn**.

   We started up cluster services about 2 minutes into the settling time period. Immediately after the node was started the Settling2_RG was acquired. We did not have to wait for the remaining 8 minutes to pass for the resource group to be acquired and come online. These were the desired results.

5. Waited for the settling time to expire.

   Upon the completion of the 600 second wait the Settling1_RG was acquired on node **mike**. Since the first node in the nodelist (**thrish**), did not become available within the settling time period the resource group was acquired on the next node in the nodelist (**mike**). These were also the desired results for this test.

6. Started up cluster services on node **thrish**.

   We started up cluster services on the last node after the settling time period expired and no resources were acquired on the node.

Overall our test proved that the settling time feature was working as expected and that the resource groups were distributed appropriately among the nodes.

11.2 Node distribution policy

One of the startup policies that you can configure for a resource group in a cluster is Online Using Node Distribution. This distribution policy causes resource groups to distribute themselves in a way that only one resource group is acquired on a node during startup. This allows for the balancing of CPU-intensive applications on different nodes.

The only node specific distribution policy supported in HACMP 5.3 is node-based distribution.

Note: If you upgrade from a previous release that allowed network-based distribution, that configuration is automatically changed to the node-based distribution.

If two or more resource groups are offline when a node joins, the policy sorts the resource groups in the following order:

1. Resource group with the least number of participating nodes
2. Alphabetic sort of resource group name

If one or more resource groups is a parent resource group HACMP will give preference to the parent resource group. Refer to section “Resource group dependencies” on page 541 for more details about this.

11.2.1 Configuring a RG node-based distribution policy

To configure this type of distribution policy follow these steps:

1. Enter `smit hacmp`
2. In SMIT, select `Extended Configuration > Extended Resource Configuration > HACMP Extended Resource Group Configuration > Add a Resource Group` and press Enter.
3. Type a resource group name.
4. Select a startup policy of `Online Using Distribution Policy` and press Enter.

Example 11-3 Configuring resource group node-based distribution policy

```
Add a Resource Group (extended)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Resource Group Name                                 []
* Participating Nodes (Default Node Priority)         []

  Startup Policy                                       Online On Home Node O>
  Fallover Policy                                      Fallover To Next Prio>
  Fallback Policy                                      Fallback To Higher Pr>
+--------------------------------------------------------------------------+
¦                            Startup Policy                                ¦
¦                                                                          ¦
¦ Move cursor to desired item and press Enter.                             ¦
¦                                                                          ¦
```

```
¦   Online On Home Node Only                                                ¦
¦   Online On First Available Node                                          ¦
¦   Online Using Distribution Policy                                        ¦
¦   Online On All Available Nodes                                           ¦
¦                                                                           ¦
¦ F1=Help                F2=Refresh             F3=Cancel                   ¦
¦ F8=Image               F10=Exit               Enter=Do                    ¦
¦ /=Find                 n=Find Next                                        ¦
+---------------------------------------------------------------------------
```

11.2.2 Node-based distribution policy test scenario

When using resource groups with this policy the documentation states that only one resource group per node will be acquired during startup. In this test scenario we wanted to prove that any additional resource groups within the cluster using other startup policies would not be affected by this restriction and that their acquisition behavior would not change.

In an effort to test the node based distribution policy we configured a 2-node cluster with five resource groups:

- three using the `Online Using Distribution Policy`
- two using the `Online On Home Node Only Policy`

This was done in an effort to exceed the number of resource groups using the `Online Using Node Distribution` policy that could be brought online within the cluster. Figure 11-2 on page 520 has a diagram of our configuration:

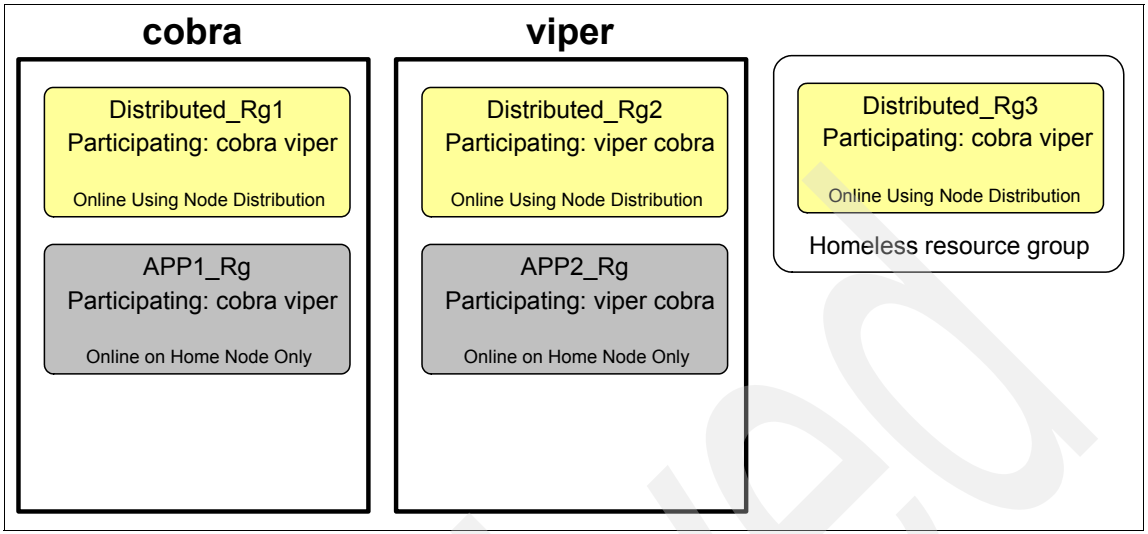

Figure 11-2 Online Using Node Distribution policy test scenario

After starting cluster services one node at a time only the `Distributed_Rg1` and `Distributed_Rg2` were acquired out of the resource groups using the `Online Using Node Distribution` policy. The third resource group, `Distributed_Rg3`, was left homeless in OFFLINE state. However, the restriction did not apply to the remaining resource groups using a different startup policy. Resource groups `APP1_Rg` and `APP2_Rg` were brought online on nodes **`cobra`** and **`viper`** respectively.

Our test results confirmed that the utilizing the node based distribution policy restricts only one resource group of this type to be acquired per node during startup. It does not affect the acquisition behavior for resource groups using other policies.

Note: The only means to bring the `Distributed_Rg3` to ONLINE state in this environment would be to go through the HACMP smit panels and to select the option to bring a resource group online every time that the cluster is started or to use the **`clRGmove`** equivalent CLI command.

11.3 Dynamic node priority (DNP)

The default node priority order for a resource group is the order in the participating nodelist. Implementing a dynamic node priority for your resource groups allows you to go beyond the default HACMP fallover behavior and calculate the destination of a resource group upon fallover based on the following three RMC pre-configured attributes:

```
cl_highest_free_mem - node with highest percentage of free memory
cl_highest_idle_cpu - node with the most available processor time
cl_lowest_disk_busy - node with the least busy disks
```

In order for DNP to be effective please note the following:

- DNP is irrelevant for cluster made up of fewer than three nodes.
- DNP is irrelevant for concurrent resource groups.
- DNP is most useful in a cluster where all nodes have equal processing power and memory.

Attention: The highest free memory calculation is performed based on the amount of paging activity taking place. It does not take into consideration whether one node has less actual physical memory than another.

At the time that this publication was produced the combination of vpath devices with the `cl_lowest_disk_busy` DNP policy was not supported. The support for such a configuration may be added later in the release in form of a fix update.

11.3.1 Configuring the dynamic node priority policy

In order to set up DNP for your resource group, no resources may already be a part of it. You will need to assign the fallover policy of dynamic node priority at the time that the resource group is created. In order for your resource group to use one of the three DNP policies you must run the following (see Example 11-4):

1. Enter `smit hacmp`
2. In SMIT, select `Extended Configuration > Extended Resource Configuration > Extended Resource Group Configuration > Add a Resource Group >` and press Enter.

Example 11-4 Adding a resource group using DNP

```
Add a Resource Group (extended)

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                  [Entry Fields]
* Resource Group Name                            []
* Participating Nodes (Default Node Priority)    []                      +

  Startup Policy                                  Online On Home Node O> +
  Fallover Policy                                 Fallover To Next Prio> +
  Fallback Policy                                 Fallback To Higher Pr> +
```

Set the Fallover Policy field to Dynamic Node Priority.

3. Assign the resources to the resource group by selecting `Extended Configuration > Extended Resource Configuration > Extended Resource Group Configuration > Change/Show Resources and Attributes for a Resource Group` and press Enter (as in Example 11-5).

Example 11-5 Selecting the dynamic node priority policy to use

```
Change/Show All Resources and Attributes for a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                [Entry Fields]
  Resource Group Name                               DNP_test1
  Participating Nodes (Default Node Priority)       alexis jessica jordan
* Dynamic Node Priority Policy                      []                    +

  Startup Policy                                    Online On Home Node O>
  Fallover Policy                                   Fallover Using Dynami>
  Fallback Policy                                   Fallback To Higher Pr>
```

4. Select one of the three available policies from the pull-down list:
   - cl_highest_free_mem
   - cl_highest_idle_cpu
   - cl_lowest_disk_busy

   Continue selecting the resources that will be part of the resource group.

5. Verify and synchronize the cluster.

   If trying to view the current DNP policy for an existing resource group in your configuration you may run the following:

```
#odmget -q group=APP1_RG HACMPresource | more

HACMPresource:
        group = "APP1_RG"
        name = "NODE_PRIORITY_POLICY"
        value = "cl_highest_free_mem"
        id = 1
```

Note: Using the information retrieved directly from the ODM is for informational purposes only as the format within the stanzas may change with updates, and/or new versions.

Hardcoding ODM queries within user defined applications is not supported and should be avoided.

11.3.2 Changing an existing resource group to use DNP policy

You cannot change the fallover policy to DNP if there are any resources currently part of the resource group. The SMIT path, to change a resource group will report an error if you attempt to do so without first removing the resources:

In order to change the policy you can:

- Remove the resource group and recreate it selecting the `Dynamic Node Priority` as the `Fallover` policy.

  or

- Enter the screen to `Change/Show Resources and Attributes for a Resource Group` screen, zero out all of the resources part of the resource group and press Enter. Then, go into the SMIT path for `Extended Configuration > Extended Resource Configuration > Extended Resource Group Configuration > Change a Resource Group` and set the Fallover policy to DNP. You can then read the resources into the resource group and synchronize the cluster for the changes to take effect.

11.3.3 How dynamic node priority works

Starting in HACMP 5.2 the dynamic node priority calculation no longer polls Event Management Services (`emsvcs`). Instead, the `clstrmgrES` polls the Resource Monitoring and Control (`ctrmc`) daemon every two minutes, and maintains a table that stores the current memory, CPU and disk I/O state of each node.

The resource monitors that contain the information for each policy are:

- IBM.PhysicalVolume
- IBM.Host

Each of these monitors may be queried during normal operation by running the commands shown in Example 11-6:

Example 11-6 Verifying resource values

```
#lsrsrc -Ad IBM.Host | grep TotalPgSpFree
        TotalPgSpFree       = 123076
        PctTotalPgSpFree    = 93.8995

#lsrsrc -Ad IBM.Host | grep PctTotalTimeIdle
        PctTotalTimeIdle    = 99.6649

#lsrsrc -Ap IBM.PhysicalVolume
resource 1:
        Name             = "vpath7"
```

```
        PVId             = "0x000685bf 0x85a0f03a 0x00000000 0x00000000"
        ActivePeerDomain = ""
        NodeNameList     = {"p630n02"}

# lsrsrc -Ad IBM.PhysicalVolume
resource 1:
        PctBusy   = 0
        RdBlkRate = 0
        WrBlkRate = 0
        XferRate  = 0
```

We can display the current table maintained by clstrmgrES by running the command shown in Example 11-7:

Example 11-7 DNP values as known to cluster manager

```
#lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#)36   1.135.1.37   src/43haes/usr/sbin/cluster/hacmprd/main.C,
hacmp.pe, 51haes_r530, r5300525a 6/20/05 14:13:01"
i_local_nodeid 1, i_local_siteid -1, my_handle 2
ml_idx[1]=0     ml_idx[2]=1     ml_idx[3]=2
There are 0 events on the Ibcast queue
There are 0 events on the RM Ibcast queue
CLversion: 8
local node vrmf is 5300
cluster fix level is "0"
The following timer(s) are currently active:
Current DNP values
DNP Values for NodeId - 1  NodeName - cobra
    PgSpFree = 130771  PvPctBusy = 0  PctTotalTimeIdle = 99.947917
DNP Values for NodeId - 2  NodeName - python
    PgSpFree = 130741  PvPctBusy = 0  PctTotalTimeIdle = 99.879167
DNP Values for NodeId - 3  NodeName - viper
    PgSpFree = 124489  PvPctBusy = 0  PctTotalTimeIdle = 99.941667
```

The values in the table are used for the DNP calculation in the event of a fallover. If clstrmgrES is in the middle of polling for the current state and a fallover occurs the last value of when the cluster was in a stable state is used for the DNP calculation.

11.3.4 Dynamic node priority test scenario

In an effort to test the dynamic node priority functionality we set up a 3-node cluster with two resource groups each using a different DNP policy. The first resource group A (APP1_RG) used the cl_highest_idle_cpu value and resource group B (APP2_RG) used the cl_highest_free_mem option to calculate its fallover

location. We tested different fallover scenarios with each one and documented the results.

Below in Figure 11-3 is a diagram of the configuration that we used:

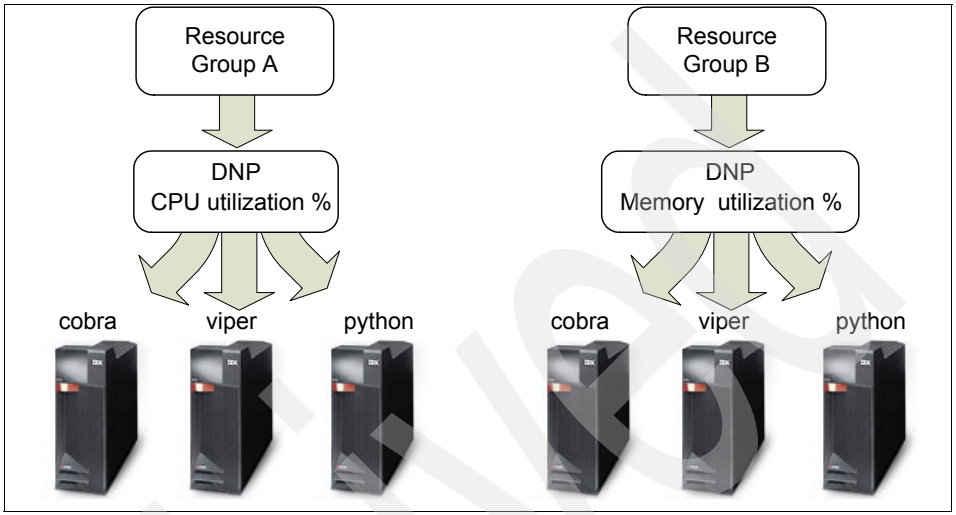

Figure 11-3 Dynamic node priority test scenario

For our environment we set each resource group to use the following startup/fallover/fallback policies:

```
Startup = Online Using Distribution Policy
Fallover = Fallover Using Dynamic Node Priority
Fallback = Never Fallback
```

We set the node order on both resource groups to: `cobra viper python`

Note: We made the nodelist in each resource group the same to show how a fallover would not follow the list order. Instead it will calculate the node with the most idle CPU, memory, or disk resources to host the resource groups depending on the DNP policy specified.

Our topology consisted of an ethernet network using IP Aliasing. This meant that in the event of a fallover multiple resource groups could be hosted on the same node. This would only occur if the DNP calculation of the most idle CPU and

memory criteria matched on the same node.

Note: To restrict the coexistence of multiple resource groups on the same node while using DNP and IP Aliasing you can implement a resource group location dependency of `Online on Different Nodes`.

An alternative is to use IPAT via Replacement to prevent a host from being able to host multiple service IPs after a fallover.

DNP scenario fallover test steps

In order to simplify our testing we did not utilize actual application workloads to impose bottlenecks on the node CPUs or memory. Instead, we ran a loop to bind two of the CPUs and utilized the **rmss** command to logically reduce the amount of memory visible to the system and then generated paging activity.

The following is the list of steps that we followed to test that the DNP policies were enforced:

1. Started cluster services on node **cobra**.
   The first resource group (`APP1_RG`) was the only one to come online as a result of the startup `Online Using Distribution Policy` in place.
2. Entered `smit hacmp > System Management (C-SPOC) > HACMP Resource Group and Application Management > Bring a Resource Group Online > selected APP2_RG > selected Restore_Node_Priority_Order` and pressed Enter.
   This brought the second resource group (`APP2_RG`) online onto node **cobra** as shown in Example 11-8:

Example 11-8 Resource group information

```
#/usr/es/sbin/cluster/utilities/clRGinfo
APP1_RG         ONLINE                      cobra
                OFFLINE                     viper
                OFFLINE                     python

APP2_RG         ONLINE                      cobra
                OFFLINE                     viper
                OFFLINE                     python
```

This was done to free up nodes (`viper` and `python`) giving us the ability to have two possible nodes to fallover to. Doing this gave us the ability to test DNP idle memory and CPU calculation in the next step.

3. We executed the following sequence of commands on node **viper** to lower the available memory and generate paging activity, as shown in Example 11-9:

Example 11-9 Inducing paging activity

```
cobra-#rmss -p
Simulated memory size is 8192 Mb.

viper-#rmss -p
Simulated memory size is 8192 Mb.

viper-#rmss -c 2000
Simulated memory size changed to 2000 Mb.

viper-#lptest 80 1000000000 > /app1/garbage_file.out

viper-#lsps -s
Total Paging Space   Percent Used
      512MB              7%

viper-#lsrsrc -Ad IBM.Host | grep PctTotalPgSpFree
      PctTotalPgSpFree    = 93.8995
```

We first downgraded the memory to 2GB on node **viper**, thus reducing the available memory. We then utilized the **lptest** command to generate large amounts of characters to a file, thus resulting in some paging activity. We monitored the value of TotalPgSpFree as it changed within the IBM.Host class. The memory load that we placed on **viper** made node **python** the next logical choice in the DNP calculation for resource group APP1_RG.

4. We executed two ksh scripts to generate a loop to bind two CPUs on node **viper** (Example 11-10):

Example 11-10 Inducing CPU load

```
#vi cpu_loop1
while true
do
:
done

#./cpu_loop1
#./cpu_loop2
```

Executing two instances of the script in the background resulted in two of the CPUs getting bottlenecked. The results are outlined in Example 11-11 on page 528:

Example 11-11 Checking CPU load

```
#sar -P ALL 5
AIX p630n02 3 5 000685BF4C00    07/08/05
System configuration: lcpu=4

11:56:46 cpu    %usr    %sys    %wio   %idle
11:56:51  0        0       0       0     100
          1      100       0       0       0
          2      100       0       0       0
          3        0       0       0     100
          -       50       0       0      50
```

We bottlenecked two of the CPUs on node **viper** since he was the next node in the resource group nodelist. Thus reducing the available idle processors we made node **python** the next logical choice in the DNP calculation for resource group APP2_RG.

5. Stopped HACMP graceful with takeover on node **cobra**.
   When we stopped cluster services on the node the DNP calculation was performed against the values in the current clstrmgrES table. The values at the time of the fallover can be seen in Table 11-2:

Table 11-2 DNP target node calculation for first scenario fallover

Cluster Nodes	DNP Highest Free CPU	DNP Highest Free Memory
viper	93.8995	93.8995
python	93.8995	93.8995
DNP Best Target Node	**python**	**python**

As seen in the results the best target node for the resources groups is **python.** The processing of the DNP calculation can be viewed in the /tmp/clstrmgr.debug at the time of the fallover (Example 11-12).

Example 11-12 Cluster manager debug messages during DNP fallover

```
#more /tmp/clstrmgr.debug
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: Using resource
attribute IBM.Host.PctTotalTimeIdle
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: for nodes 3, 2.
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: using values ,
0.0000, 0.0000.
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: condition is
DNP_largest
Wed Jul 13 10:40:53 In largest_comparison
```

```
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: Computed node order -
3, 2.
Wed Jul 13 10:40:53 For Resource Group APP1_RG, BestNode got node order
Wed Jul 13 10:40:53 NodeList::showNodeList: Got the following 2 node IDs:
Wed Jul 13 10:40:53 NodeList::showNodeList: 3 2
Wed Jul 13 10:40:53 The best node for group APP1_RG is python.
Wed Jul 13 10:40:53 RGPA got viper as highest priority node.
.............
.............
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: Using resource
attribute IBM.Host.TotalPgSpFree
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: for nodes 3, 2.
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: using values ,
0.0000, 0.0000.
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: condition is
DNP_largest
Wed Jul 13 10:40:53 In largest_comparison
Wed Jul 13 10:40:53 NodeList::RmcComputeNodePriority: Computed node order -
3, 2.
Wed Jul 13 10:40:53 For Resource Group APP2_RG, BestNode got node order
Wed Jul 13 10:40:53 NodeList::showNodeList: Got the following 2 node IDs:
Wed Jul 13 10:40:53 NodeList::showNodeList: 3 2
Wed Jul 13 10:40:53 The best node for group APP2_RG is python.
Wed Jul 13 10:40:53 RGPA got viper as highest priority node.
```

6. Reintegrated node **cobra** into the cluster.
7. We made changes to the memory and CPU parameters a second time. This time setting up nodes **cobra** and **viper** so that the DNP calculation would distribute the two resource groups on node **python** among the two different nodes (see Example 11-13).

Example 11-13 Inducing CPU and memory load

```
viper-#rmss -r
Simulated memory size is 8192 Mb.

viper-#./cpuloop1
viper-#./cpuloop2

viper-#sar -P ALL 5
AIX p630n02 3 5 00065BF4C00 07/08/05
System configuration: lcpu=4

18:03:20 cpu    %usr    %sys    %wio   %idle
18:03:25  0        0       0       0     100
          1      100       0       0       0
          2      100       0       0       0
          3        0       0       0     100
```

```
           -       50       0       0       50

viper-#lsps -s
Total Paging Space   Percent Used
      512MB               1%


cobra-#rmss -c 2000
Simulated memory size changed to 2000 Mb.

cobra-#lptest 80 1000000000 > /app1/garbage_file.out

cobra-#lsps -s
Total Paging Space   Percent Used
      512MB              6%
```

We bottlenecked two of the CPUs on node **viper** and generated paging activity on node **cobra** so that the DNP policies in each resource group distributed them onto a different node.

8. Executed a **halt -q** on node **python**.
   This resulted in a fallover of the APP1_RG resource group over to node **cobra** since the CPU were all available on it. The APP2_RG instead moved to node **viper** because it had no paging activity more available free memory.
   Table 11-3 shows the current clstrmgrES values at the time of the fallover.

Table 11-3 DNP target node calculation for second scenario fallover

Cluster Nodes	DNP Highest Free CPU	DNP Highest Free Memory
viper	93.8995	93.8995
cobra	93.8995	93.8995
DNP Best Target Node	**cobra**	**viper**

DNP scenario test results

In our DNP test environment scenario we were able to successfully prove that the default nodelist order was ignored during a fallover. After reintegrating the failed node we altered the memory and CPU loads on the two idle nodes and tested failing the node currently hosting the resource groups. The DNP calculation the second time around successfully distributed the resource groups onto the two different standby nodes.

In our environment we used practical commands to alter the CPU and memory load on the systems. We expect the results to be same when the load is imposed by a running application. When setting up your own DNP environment be sure to

test all possible fallover scenarios and use the documented **`lsrscr -Ad`** and **`lssrc -ls clstrmgrES`** commands documented in this manual to monitor the current CPU, memory and disk I/O load on the cluster nodes.

11.4 Priority override location (POL)

The concept of a priority override location was introduced in HACMP 5.1 to replace the old “sticky bit” function. One major difference from releases prior to that is that the POL policy is now always implicitly set whenever you manually move a resource group. This is due to the thought that whenever you explicitly move your resource group you want it to stay there.

When the POL gets set, the resource group is bound to that node. The policy will remain there until you reboot all cluster nodes or issue another resource group move specifying the `Restore_Node_Priority_Order` option. There is an additional setting called Persist Across Cluster Reboot. If enabled the POL will be retained even after rebooting all cluster nodes.

Note: Anytime that you explicitly move a resource group you should always either remember to reset the POL when finished, or be aware that it is present since the default fallover behavior is indirectly changed.

The POL attribute also comes into play when bringing a resource group offline or online. If you choose to bring your resource group offline the POL will reflect a state of OFFLINE. Whenever you bring the resource group back online and you select an explicit target node the POL will get set for that node.

Moving a resource group

Note that this option is not available for non-concurrent resource groups.

To move a resource group:

1. Proceed into `smit hacmp > System Management (C-SPOC) > HACMP Resource Group and Application Management > Move a Resource Group to Another Node / Site > Move Resource Groups to Another Node` > select resource group and press Enter.
2. Select one of the following two options:
   - `Restore_Node_Priority_Order`

     Or,
   - `Destination node`

3. In the next screen choose the appropriate value for this setting:
   – Persist Across Cluster Reboot?
     The default value is false. If 'true' is selected, then the Priority Override Location will be retained after the entire cluster reboots. If 'false' is selected, then the Priority Override Location will not be retained after the entire cluster reboots and the resource group will go back to it's default behavior after a cluster reboot.
4. After your resource group is moved you can confirm that the move was successful and display that the POL is set by running **clRGinfo -p**. The sample result is available in Example 11-14.

Example 11-14 Checking for the priority override location

```
# /usr/es/sbin/cluster/utilities/clRGinfo -p

Cluster Name: migration2

Resource Group Name: C1ORG2
Priority Override Information:
    Primary Instance POL:
Node                         State
---------------------------- ---------------
viper                        ONLINE
cobra                        OFFLINE

Resource Group Name: C1ORG1
Priority Override Information:
    Primary Instance POL:   viper
Node                         State
---------------------------- ---------------
cobra                        OFFLINE
viper                        ONLINE
```

Whenever the POL is set the `/usr/es/sbin/cluster/etc/clpol` file is generated on both nodes and will remain there until there is a cluster reboot or a resource group move is issued with the `Restore_Node_Priority_Order` option selected. The file contains a representation of all priority override locations and may be interpreted with the following format: [RG id] [node id] [pol] [per?]

```
3 2 2 1      // RG 3 on node 2 is OFFLINE persistent
3 1 2 1      // RG 3 on node 1 is OFFLINE persistent
1 1 1 0      // RG 1 on node 1 is ONLINE non-persistent
```

The file data is numeric, and is not intended to be viewed or manipulated by an end user.

Attention: The manual removal of the clpol file does NOT clear the POL setting on an active cluster.

Bringing resource group online or offline

There are a couple of additional operations where you may implicitly set the priority override location on a resource group. If you bring a resource group online or offline using the following menu options and select a destination node the POL attribute will get set:

- `smit HACMP > System Management (C-SPOC) > HACMP Resource Group and Application Management > Bring a Resource Group Online`
- `smit HACMP > System Management (C-SPOC) > HACMP Resource Group and Application Management > Bring a Resource Group Offline`

When bringing the resource group online if you select the `Restore_Node_Priority_Order` the resource group will come online on the node designated as the highest priority node (based on nodelist) and will not set the POL attribute.

When you bring a resource group offline using the path noted above keep in mind that a different type of POL is set. The resource group will show to have a POL of OFFLINE. This means that if any subsequent attempts to restart cluster services will not allow you to bring the resource group online. The way to clear this POL is to move or bring the resource group online by selecting the `Restore_Node_Priority_Order` option.

Resetting the priority override location

Once the POL is in effect, in order to clear it and restore the default resource group behavior you will need to initiate another resource group move and select the Restore_Node_Priority_Order option. Example 11-15 shows how the menu option will show up:

Example 11-15 Resetting priority override location

```
Move a Resource Group to Another Node / Site

Move cursor to desired item and press Enter.

  Move Resource Groups to Another Node
  Move Resource Groups to Another Site

  +--------------------------------------------------------------------------+
  ¦                        Select a Destination Node                         ¦
  ¦                                                                          ¦
  ¦ Move cursor to desired item and press Enter.                             ¦
```

```
   # To choose the highest priority available node for the
   # resource group, and to remove any Priority Override Location
   # that is set for the resource group, select
   # "Restore_Node_Priority_Order" below.
   Restore_Node_Priority_Order

   # To choose a specific node, select one below.
   viper

 F1=Help                 F2=Refresh              F3=Cancel
 F8=Image                F10=Exit                Enter=Do
F1/=Find                 n=Find Next
```

When you select this option:

- The RG moves to the highest priority node currently available
- Any persistent priority override location previously set is removed
- If already on the highest priority node the POL is cleared but no move actions are taken.

When using resource groups with a startup policy of `Online Using Distribution Policy` the menu option to reads differently. Instead of `Restore_Node_Priority_Order` the option to clear the POL is `Reset_Any_Priority_Overrides`. For this type of resource group the restoring the node priority should never cause an actual resource move since there is no concept of a higher priority node.

Priority override location recommendations

Whenever manipulating the location of your resource groups be specially mindful of where the POL setting is and its current value. The instances where we saw the most potential for confusion or for potential for the default behavior to not occur were:

- Bringing a resource group online

  If trying to bring RG back online on the higher priority node it is easy to accidentally set the POL by choosing the nodename and not the `Restore_Node_Priority_Order` option.

- Bringing a resource group offline

  When bringing an RG offline the operation will also set a POL. It will leave the resource group down in an OFFLINE state. A subsequent start of cluster services will not bring the RG back online unless you reissue an acquire and select the `Restore_Node_Priority_Order` option

- Moving a RG back to the highest priority node

  You must be specially mindful if the `Never Fallback` option or the `Online Using Distribution Policy` are set and you ever manually move a resource group back to the higher priority node. If instead of selecting `Restore_Node_Priority_Order` you select the specific <nodename>, the POL will get set on that node. In this scenario it is easy to forget that the policy is in place and that it could potentially cause a resource group to not behave as expected during later operations.

- Setting the `Persist Across Cluster Reboot` option to yes.

  Use this option with caution. Whenever you manipulate the location of a resource group always make a note if you choose to set this option. Someone not familiar with the priority override location behavior could forget that it is set and encounter mixed results.

11.5 Delayed fallback timer

This setting allows you to configure the fallback behavior of a resource group to occur at one of the predefined recurring times: daily, weekly, monthly, yearly, or on a specific date and time. This is useful for scheduling fallbacks to occur during off-peak business hours. The diagram in Figure 11-5 on page 549 displays the three different phases involved when utilizing delayed fallback timers.

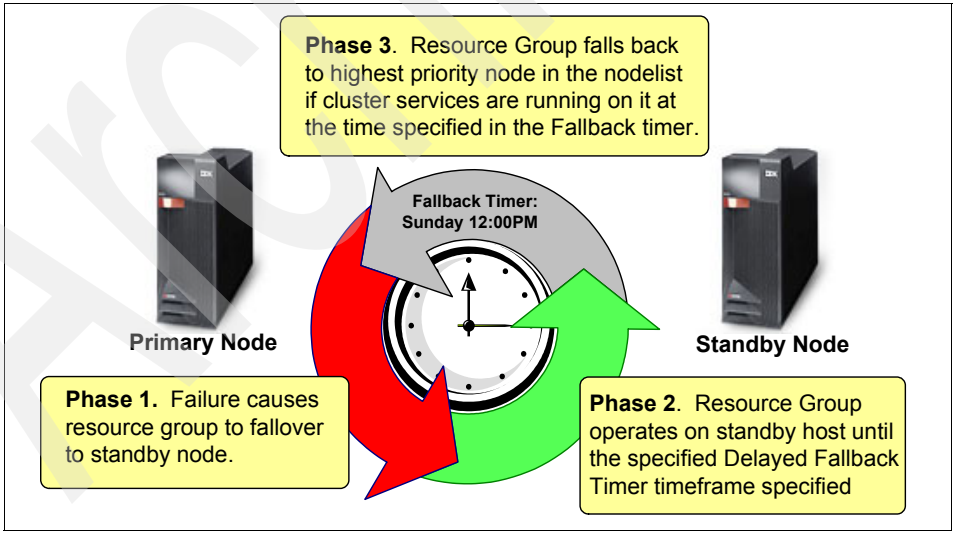

Figure 11-4 Delayed fallback timer phases

In the event of a failure your resource group will fallover to a standby node. The resource group will remain there until the specified time frame in the delayed fallback timer. If cluster services are active on the primary node at that time, the resource group will fallback to the highest priority node. If a node with higher priority is not available during that time, the fallback timer will be reset and the fallback will be delayed until the next iteration of that time setting.

Delayed fallback timer behavior

In order to effectively implement an environment using delayed fallback timers consider the following points:

- The delayed fallback timer behavior is only specific to resource groups using the fallback policy of `Fallback To Higher Priority Node In The List`.
- If there is no higher priority node available at the time frame when the timer is set to fallback no resource group action will take place.
- When using a specific date value for a fallback timer the check for a higher priority node will not reoccur. Only using any of the other values will the fallback timer recycle and continue to check on the next iteration.
- If a priority override location (POL) is set on a particular node for your resource group at the time that the fallback timer performs its check the resource group will be moved back to the node specified within the POL setting.
- When using a `Same Node Dependency` set, if one resource group in the set has a fallback timer, it applies to the set. For resource groups using the `Same Site Dependency` policy, if a fallback timer is used it must be identical for all resource groups in the set. See section “Resource group dependencies” on page 541 for more details on this.
- You cannot remove a fallback timer if a resource group is currently using it.
- You cannot configure delayed fallback timers through the `Initialization and Standard Configuration` screens. They may only be configured through the `Extended Configuration` path.

Configuring fallback timer for resource groups

To configure this feature you must do the following:

1. Enter smit hacmp.
2. Select `Extended Configuration > Extended Resource Configuration > Configure Resource Group Run-Time Policies > Configure Delayed Fallback Timer Policies > Add a Delayed Fallback Timer Policy` and press Enter.
3. Select the policy from the picklist: daily, weekly, monthly, yearly, specific date

4. Enter field values as follows:
   - Name of Fallback Policy
     Specify the name of the policy using no more than 32 characters. Use alphanumeric characters and underscores only. Do not use a leading numeric value or any reserved words.
   - Policy selected
     Select applicable values based on policy selected. The fields will include: day, hour, minutes, year, or a combination.
5. To then define it to the resource group enter `smit hacmp > Extended Configuration > Extended Resource Configuration > HACMP Extended Resource Group Configuration` > select the desired resource group from the list and press Enter.
6. Press the F4 key to select one of the policies configured in the previous steps. The following screen will be displayed (Example 11-16):

Example 11-16 Defining a fallback timer policy into a resource group

```
Change/Show All Resources and Attributes for a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                              [Entry Fields]
  Resource Group Name                             fallback1_rg
  Participating Nodes (Default Node Priority)     cobra viper python

  Startup Policy                                  Online On Home Node Only
  Fallover Policy                                 Fallover To Next Priority>
  Fallback Policy                                 Fallback To Higher Priori>
  Fallback Timer Policy (empty is immediate)      []                        +

  Service IP Labels/Addresses                     []                        +
  Ap+-------------------------------------------------------------------------+ +
    ¦              Fallback Timer Policy (empty is immediate)               ¦
  Vo¦                                                                       ¦ +
  Us¦ Move cursor to desired item and press Enter.                          ¦ +
  Au¦                                                                       ¦ +
  Fi¦   july12                                                              ¦ +
  Fi¦   july1_test                                                          ¦ +
[MOR¦                                                                       ¦
    ¦ F1=Help                F2=Refresh              F3=Cancel              ¦
F1=H¦ F8=Image               F10=Exit                Enter=Do               ¦
F5=R¦ /=Find                 n=Find Next                                    ¦
F9=S+-------------------------------------------------------------------------+
```

7. Select the Fallback Timer Policy desired from the picklist and press Enter.
8. Add any additional resources to the resource group and press Enter.
9. Run a verification and synchronization on the cluster to propagate the changes to all nodes.

Displaying delayed fallback timers in a resource group

If coming into an environment that has a running cluster you can check for any existing fallback timer policies for the resource groups by looking at the output of the **clshowres** command, as shown in Example 11-7 on page 524:

Example 11-17 Cluster resources

```
#/usr/es/sbin/cluster/utilities/clshowres
Resource Group Name                          fallback1_rg
Participating Node Name(s)                   cobra viper python
Startup Policy                               Online On Home Node Only
Fallover Policy                              Fallover To Next Priority Node In
The List
Fallback Policy                              Fallback To Higher Priority Node
In The List
Delayed Fallback Timer                       july12
```

Note: Remember that the `Delayed Fallback Timer` attribute will only be displayed in the menus if the fallback policy is set to Fallback To Higher Priority Node In The List.

To display the values within an existing fallback timer policy you may enter the **smit hacmp > Extended Configuration > Extended Resource Configuration > Configure Resource Group Run-Time Policies > Configure Delayed Fallback Timer Policies > Change/Show a Delayed Fallback Timer Policy** and select a policy from the list.

An alternative is to query the `HACMPtimer` object class by running the following:

```
#odmget HACMPtimer
   HACMPtimer:
        policy_name = "july12"
        recurrence = "once"
        year = 105
        month = 6
        day_of_month = 12
        week_day = 0
        hour = 17
        minutes = 47
```

Attention: Using the information retrieved directly from the ODM is for informational purposes only as the format within the stanzas may change with updates, and/or new versions.

Hardcoding ODM queries within user defined applications is not supported and should be avoided.

Delayed Fallback timer test scenario

In order to test the functionality of this feature we configured a two node cluster with one resource group using a fallback timer. For the purposes of our test we used a fallback timer that held a specific date.

The following are the attributes for the resource group that we used:

```
Resource Group Name: fallback1_rg
Participating Nodes: cobra viper
Startup Policy: Online On Home Node Only
Fallover Policy: Fallover To Next Priority Node In The List
Fallback Policy: Fallback To Higher Priority Node In The List
Delayed Fallback Timer: july12
```

In the process of running our cluster verification/synchronization we noted the following message in the clverify output:

```
Resource Group 'fallback1_rg' is configured to use 'july12' fallback timer
policy.
```

Once the policy was in place we simulated the 3 phases outlined in Figure 11-5. The following are the steps we took:

1. Started cluster services on both nodes: **cobra viper**
2. We simulated a failure (Phase 1 in Figure 11-4 on page 535) for the primary node **cobra.**

   To do this, we stopped HACMP services selecting the graceful with takeover option on the node. We elected to move the resource group in this fashion to avoid setting a POL with a resource group move operation.
3. We confirmed that the resource group came online on node **viper**.
4. Reintegrated node **cobra** (the higher priority node) into the cluster.

   When we started cluster services on the node we noted the following message in the /tmp/hacmp.out file:

   ```
   No action taken on resource group 'fallback1_rg'
   The Resource Group 'fallback1_rg' has been configured
   to fallback using 'july12' Timer Policy.
   ```

Even though the higher priority node came into the cluster the timer policy was enforced and the resource group was not immediately failed back. In Phase 2 in Figure 11-4 on page 535 we can see how the resource group will remain on the standby node until the fallback time period occurs.

5. Once the timer policy encountered the time period it was set for, a resource group move was called and the `fallback1_rg` resource group moved back to the primary node (**`cobra`**). Since the primary node for the resource group was available at the time of the fallback timer check (Phase 3 in Figure 11-4 on page 535) the resource group was immediately moved back to it.

Delayed fallback timer test results

In our test we proved that the fallback timer policy was accurately enforced and that the resource group did not return to the primary node until the time specified. The one observation worth mentioning was the method in which the time was calculated after we implemented the fallback policy, with respect to daylight savings time. With the daylight savings time enabled on our machines the fallback calculation was not accurate as reported in the `hacmp.out` log file. We discovered this while monitoring that the fallback timer policy was going to be enforced. Below is an example of the discrepancy that we saw:

```
#date
Wed Jul 13 17:18:09 EST 2005

#tail -f /tmp/hacmp.out
No action taken on resource group 'fallback1_rg'
The Resource Group 'fallback1_rg' has been configured
to fallback on 'Wed Jul 13 18:25:00 2005'
```

The cluster reported that the fallback would be taking place at `18:25:00`, however, the fallback timer policy that we set was set for `17:25:00` as can be seen below:

```
#odmget HACMPtimer
HACMPtimer:
        policy_name = "july13"
        recurrence = "once"
        year = 105
        month = 6
        day_of_month = 13
        week_day = 0
        hour = 17
        minutes = 25
```

The same test with the daylight savings time option disabled worked without any calculation discrepancy.

Note: The discrepancy with the daylight savings time calculation was tested with early version of the HACMP 5.3 code and will be addressed with later fixes in this release.

In the mean time, if you are going to set a delayed fallback timer and have daylight savings enabled on your machines, check your environment for the same discrepancy to avoid a fallback at a different time than you expected.

11.6 Resource group dependencies

Introduced in HACMP 5.2, the concept of a resource group parent/child dependency allows administrators to have greater control of multi-tiered applications where one application depends on the successful startup of another.

Business configurations that use multi-tiered applications can benefit from the ability to set up these dependencies. For instance, in an environment where the database must be online before the application server, if the database goes down and fails over to a different node, the resource group containing the application server would also be brought down and back up on any of the available cluster 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 and remain offline.

An location dependency feature added in HACMP 5.3 gives you control over the type of resource group distribution policy during the acquisition and release of a resource group.

The new concepts introduced with dependencies include:

- **Parent resource group**
  The parent resource group is the first one to be acquired during the resource group acquisition.
- **Child resource group**
  The child resource group is a dependant of the parent and will not come online unless the parent is available. In the event that the parent resource group falls over or is taken offline the child resource group will also be taken offline and follow the parent.
- **Child dependency**
  A child dependency gives you the ability to bind resource groups in a hierarchical fashion up to three levels deep. A set of resource groups part of parent/child dependency will always be acquired and released together. In addition to this, you may also configure a location dependency to manage the collocation of your resource groups.

- **Location Dependency**
  New with HACMP 5.3, this is an extension to managing your resource groups that enables you to specify a policy that will determine how your resource groups will get distributed among nodes during acquisition and fallover events. You can select to have a set of resource groups all coexist on the same node, or choose to distribute them among the cluster nodes.

11.6.1 Resource group child dependency

The implementation of a resource group child dependency will give you added flexibility when trying to configure applications that are dependant on each other. Always make new people managing or trying to understand the cluster configuration aware about these dependencies since enabling this feature changes the default behavior of HACMP standard policies.

You can display whether any dependencies are present by using the **`clrgdependency`** command.

Planning for resource group child dependencies

There are various configuration considerations when preparing to use resource group dependencies:

- Plan which resource groups will contain which applications. Ensure that any applications that require sequencing are placed into different resource groups. Once they are separated, dependencies may be built among the resource groups.
- Be aware of the following restrictions:
  - Dependencies can only be three levels deep
  - You cannot specify circular dependencies between resource groups
- For each application that is going to be included in dependent resource groups, configure application servers and application monitors. In general, we recommend that you configure a monitor that will check the running process for an application in the child resource group, and a monitor that will check the running process for the application in the parent resource group.

  For the parent resource group it is also advisable to configure a startup application monitor to ensure its success. This ensures that after the parent resource group is acquired the child is also able to come online.
- To minimize the change of data loss during the application stop and restart process, you should 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.

Configuring a resource group child dependency

Be aware that the dependencies that you configure are:

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

The following are the steps to configure a resource group child dependency:

1. Enter `smit hacmp`
2. In SMIT, select `Extended Configuration > HACMP Extended Resource Configuration > Configure Resource Group Run-Time Policies > Configure Dependencies between Resource Groups > Configure Parent/Child Dependency > Add Parent/Child Dependency between Resource Groups` and press Enter.
3. Fill in the fields as follows:
   - **Parent Resource Group**
     Select the parent resource group from the list. During resource group acquisition HACMP will acquire the parent resource group before the child resource group is acquired.
   - **Child Resource Group**
     Select the child resource group from the list and press Enter. During release, HACMP will bring the child resource group offline before the parent resource group.

     HACMP will prevent you from specifying a circular dependency.
4. Use the SMIT Verify and Synchronize HACMP Configuration option to guarantee that the desired configuration if feasible given the dependencies selected, and that the changes are propagated to all nodes in the cluster.

11.6.2 Resource group location dependency

New with HACMP 5.3 is the implementation of resource group location dependencies. The goal of these is to control the location where dependent resource groups will come online. The available policies include the option to have all dependent resource groups on the same node, or to distribute them across different nodes. If sites are being used there is also a policy that will collocate the resource groups within the same site.

The following are the three location dependency policies:

- Online On Same Node Dependency
- Online On Same Site Dependency
- Online On Different Nodes Dependency

You can combine a resource group child dependency with a location dependency. In doing so, you can specify that a set of resource groups will always be online on the same node, or that a set or resource groups will always be online on different nodes.

Note: Make sure the review the planning notes for each of these policies before trying to implement them in your configuration.

Planning for Online On Same Node Dependency

In order to effectively implement this policy be aware of the following:

- All resource groups part of the same dependency must have the same nodelist (participating nodes in the same order)
- All non-concurrent resource groups in the same dependency must have the same startup/fallover/fallback policies
  - Online Using Node Distribution Policy is not allowed for startup
  - If dynamic node priority is being used as the fallover policy then all resource group in the dependency set must be using the same DNP policy.
  - If one resource has a fallback timer configured it will apply to the entire set of resource group in the dependency. All resource groups within the set must have the fallback time setting.

Configuring Online on Same Node location dependency

The same-node location dependency allows you to specify a set of resource groups to always be acquired onto the same node. You configure this policy by following these steps:

1. Enter `smit hacmp`
2. In SMIT, select `Extended Configuration > HACMP Extended Resource Configuration > Configure Resource Group Run-Time Policies > Configure Dependencies between Resource Groups > Configure Online on the same node Dependency > Add Online on the same node Dependency between Resource Groups` and select the resource groups that will be part of that set.

   Remember to ensure that all the node participating nodelists in each of the resource groups are identical, otherwise this operation will report and error and fail.
3. In order to propagate the change across all cluster nodes remember to verify and synchronize your cluster.

Planning for Online On Different Nodes Dependency

In order to effectively implement this policy be aware of the following rules and restrictions:

- Only one Online On Different Nodes dependency is allowed per cluster
- Each resource group set should have a different home node for startup
- When using this policy you can assign three different priorities:
  - **High**
  - **Intermediate**
  - **Low**

  The higher priority resource groups take precedence over lower priority resource groups at startup, fallover, and fallback:

  - If a resource group with high priority is ONLINE on a node, then no other resource group in a different node dependency set can come online on that node.
  - If a resource group in this set is ONLINE, but a resource with a higher priority fails over or falls back to this node it will be the one brought online, and the lower priority resource group will be taken OFFLINE or relocated to another node if possible.
  - Resource groups with the same priority cannot come be brought ONLINE on the same node. The priority of resource groups within the same set that have the same priority level is determined by the alphabetical order of the groups.
  - Resource groups with the same priority do not cause one another to be moved from the node after a fallover or fallback.
  - If a parent/child dependency is specified, then the child cannot have a higher priority than its parent.

Configuring Online on Different Node location dependency

To configure this type of location dependency follow the screens below:

1. Enter smit hacmp.
2. In SMIT, select `Extended Configuration > HACMP Extended Resource Configuration > Configure Resource Group Run-Time Policies > Configure Dependencies between Resource Groups > Configure Online on the same node Dependency > Add Online on Different Nodes Dependency between Resource Groups and press Enter.`
3. Fill in the fields as follows and press Enter:

- **High Priority Resource Group(s)**
  Select the resource groups in this set to be acquired and brought ONLINE

before lower priority resource groups.

On fallback and fallover, these resource groups are processed simultaneously and brought ONLINE on different target nodes before any other groups are processed. If different target nodes are unavailable for fallover or fallback, these groups (same priority level) can remain on the same node.

The highest relative priority within this list is the group listed first (on the left), as for the nodelist.

- **Intermediate Priority Resource Group(s)**
  Select the resource groups in this set to be acquired and brought ONLINE after the high priority resource groups and before the low priority resource groups are brought ONLINE.

  On fallback and fallover, these resource groups are processed simultaneously and brought ONLINE on different target nodes before low priority resource groups are processed. If different target nodes are unavailable for fallover or fallback, these groups (same priority level) can remain on the same node.

  The highest relative priority within this list is the group listed first (on the left), as for the nodelist.

- **Low Priority Resource Group(s)**
  Select the resource groups in this set to be acquired and brought ONLINE after the higher priority resource groups are brought ONLINE.

  On fallback and fallover, these resource groups are brought ONLINE on different target nodes after the higher priority resource groups are processed.

  Higher priority resource groups moving to a node may cause these groups to be moved or be taken OFFLINE.

4. Continue configuring run-time policies for other resource groups or verify and synchronize the cluster.

Planning for Online On Same Site Dependency

When you configure two or more resource groups to use a location dependency they belong to a set for the particular dependency. The following rules and restrictions are applicable for site dependencies:

- All resource groups in a `Same Site Dependency` 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 a set.

- 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 is supported.
- Both concurrent and non-concurrent resource groups are allowed.
- 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 same may be OFFLINE or in the ERROR state.
- If you add a resource group included in a `Same Node Dependency` set to a `Same Site Dependency` set, then you must add all the other resource groups in the `Same Node Dependency` set to the `Same Site Dependency` set.

Configuring Online on Same Site location dependency

To configure a set of resource groups to use the same site location dependency do the following:

1. Enter smit hacmp
2. In SMIT, select `Extended Configuration > HACMP Extended Resource Configuration > Configure Resource Group Run-Time Policies > Configure Dependencies between Resource Groups > Configure Online on the same node Dependency > Add Online on the same Site dependency between Resource Groups` and press Enter.
3. Select the resource groups from the list to be part of this set.
   During acquisition these resource groups will be brought ONLINE on the same site according to the site and node startup policy specified in the resource group. On fallback or fallover the resource groups are processed simultaneously and brought ONLINE on the same site.
4. Verify and synchronize the cluster.

11.6.3 Limitations for combinations of dependencies

The following limitations apply to configuration 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 at the same time.
- If a resource group belongs to both an `Online on Same Node` dependency set and an `Online on Different Node` dependency set, all nodes in the Online of Same Node dependency set have the same priority as the shared resource group.

- 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.

11.6.4 Displaying resource group dependencies

If you acquire an pre-existing HACMP cluster and want to display the current dependencies in place you may use the **`clrgdependency`** command. Below are some examples of what you would expect to see return:

```
#clrgdependency -?
usage: clrgdependency -t <ANTICOLLOCATION> -u -hp <high priority RG list> -ip
<intermediate priority RG list> -lp <low priority RG list>
usage: clrgdependency -t <PARENT_CHILD | NODECOLLOCATION | SITECOLLOCATION |
ANTICOLLOCATION > -sl

#clrgdependency -t PARENT_CHILD -sl
#Parent                          Child
DB2_1Rg                          Child_1Rg
DB2_2Rg                          Child_2Rg
```

At the time of this publication there was no man page available for the **`clrgdependency`** command.

An alternative is to look at the ODM stanzas `HACMPrg_loc_dependency` and `HACMPrgdependency` by running an odmget against them:

```
#odmget HACMPrgdependency
HACMPrgdependency:
         id = 0
         group_parent = "DB2_1Rg"
         group_child = "Child_1Rg"
         dependency_type = "PARENT_CHILD"
         dep_type = 0
```

Attention: Using the information retrieved directly from the ODM is for informational purposes only as the format within the stanzas may change with updates, and/or new versions.

Hardcoding ODM queries within user defined applications is not supported and should be avoided.

11.6.5 Resource group dependency test scenario

There are many possible combinations for administering your resource groups when implementing resource group dependencies. In an effort to test some of these, we set up a configuration utilizing a combination of the different policies

available. Note that the configuration that we used goes beyond the default behavior of HACMP and assumes that you are familiar with how resource group dependencies work.

We selected to use a 3-node cluster hosting multiple instances of DB2 databases each with child resource groups hosting WebSphere applications. We made the third node a standby hosting the Tivoli application for backups and a resource group hosting an application testing ground. Figure 11-5 shows the configuration that we used:

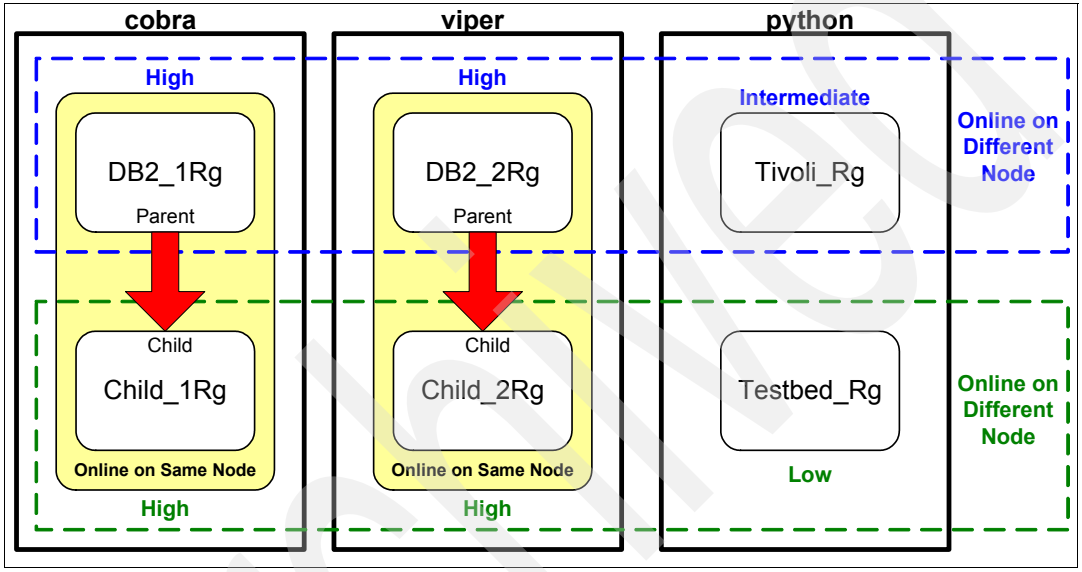

Figure 11-5 Resource group dependency test scenario

We implemented six different dependencies, including: `Parent/Child` (2), `Online on Same Node` (2 sets), and `Online on Different Node` (2 sets) `dependencies`. These dependencies are also outlined on Table 11-4 on page 550. We implemented a configuration where the parent and corresponding child resource groups always come online on the same node, but where each parent and the Tivoli resource group always come online on a different node in order to distribute the load. We included a repulsion dependency between the child resource groups and the `Testbed_Rg` so that if the production resources were to ever move to the standby node **`python`**, the resource group hosting the testing bed had a LOW priority and was taken offline.

Table 11-4 Test scenario resource group dependency and distribution policies

	Resource Groups defined to cluster					
Policies	DB2_1Rg	Child_1Rg	DB2_2Rg	Child_2Rg	Testbed_Rg	Tivoli_Rg
Parent / child	YesYes	Yes				
Online on Same Node	Yes	Yes				
Parent / child			Yes	Yes		
Online on Same Node			Yes	Yes		
Online on Different Node		High		High	Low	
Online on Different Node	High		High			Intermediate

In Table 11-5, we outline the different resource group attributes and start/fallover/fallback policies that we implemented for our resource groups. Review the concepts section of this manual if you are not familiar with the acronyms listed in each of the resource group policies. We also included the participating nodes, the service IP labels, and the application servers used for each resource group.

Table 11-5 Test scenario resource group attributes

	Resource Groups defined to cluster					
Attributes	DB2_1Rg	Child_1Rg	DB2_2Rg	Child_2Rg	Testbed_Rg	Tivoli_Rg
Startup	OHNO	OHNO	OHNO	OHNO	OHNO	OHNO
Fallover	FNPNL	FNPNL	FNPNL	FNPNL	FNPNL	FNPNL
Fallback	NF	NF	NF	NF	NF	NF
Participating Nodes	cobra python viper	cobra python viper	viper python cobra	viper python cobra	python cobra viper	python cobra viper
Service IP	app1svc		app2svc		app3svc	app4svc
Application Server	db2_1	db2_child1	db2_2	db2_child2	testbed_app	tivoli_app

Test scenario operations and results

For our testing, we tried to recreate failures and resource group operations that would be common in a production environment. We found that halting a node or doing a graceful stop with takeover were the methods that truly enforce the dependency policies. In our environment, since there were always high priority resource groups on all nodes the resource group move operations were not allowed, as to be discussed in the test results below.

Our test sequence was as follows:

1. Stopped HACMP graceful with takeover on node **viper**.

   After stopping cluster services with takeover the resource groups successfully moved over to the standby node **python**. The repulsion distribution policy between the child resource groups and the Testbed_Rg was enforced, and the test resource group was brought OFFLINE. The Tivoli_Rg was left ONLINE because priority setting within the policy was set to Intermediate. The output of **clRGinfo** displayed the following:

```
#/usr/es/sbin/cluster/utilities/clRGinfo
DB2_2Rg         OFFLINE                     viper
                ONLINE                      python
                OFFLINE                     cobra

Child_2Rg       OFFLINE                     viper
                ONLINE                      python
                OFFLINE                     cobra

Testbed_Rg      OFFLINE due to lack of node python
                ERROR                       cobra
                OFFLINE                     viper

Tivoli_Rg       ONLINE                      python
                OFFLINE                     cobra
                OFFLINE                     viper
```

2. Attempted to bring Testbed_Rg back online.

   While in this state, we attempted to bring the test resource group online using the HACMP menus. We found that there were no available nodes to bring the resource group online. We then attempted to select the Restore_Node_Priority_Order option to bring the resource group online and the command executed with no errors, but no action was taken as we would expect.

3. Reintegrated node **viper** into the cluster.

   We reintegrated the node back into the cluster and the Testbed_Rg was brought online on it.

Note: This occurred because the `Testbed_Rg` was a LOW priority resource group that was currently homeless.

4. Attempted to move `DB2_2Rg` back from **`python`** to the original node **`viper`**.

   While in the failed over state, we attempted to move the original resources for node **`viper`** back to its home node by using the HACMP menus. This operation did not show the node as an available node to move back to. The only way that we were able to bring the resources back to node **`viper`** was to stop node **`python`** graceful with takeover.

5. Reintegrated node **`python`** into cluster.

   We noticed a delay during the reintegration of this node back into the cluster, but the overall operation was successful and the `Testbed_Rg` returned to node **`python`**. During the processing lag nothing was logged to the `hacmp.out` or `clstrmgr.debug` files. The `Tivoli_Rg` remained online on node **`viper`** and we opted to leave it there.

The overall testing with the resource group dependencies proved to be successful. The main observation was that the resource group move operations were not allowed for any of the resource groups after reintegrating the node that had been previously failed over. This was applicable for the resource groups using the HIGH and LOW priorities.

At the time of the testing we were unable to test dynamic DARE operations of changes to the resource group dependency policies due to lack of functionality. However, dynamic changes to resource group dependency policies will be supported with fixes beyond base level code later in this release.

12

Customizing events

This chapter provides information about custom events and how to use AIX error notification to detect hardware or software errors which are out of HACMP's scope. Here we discuss two ways to intercept and handle non cluster related errors:

- **HACMP pre/post-event command:** Designed to manage extra configuration issues that HACMP cannot handle by default.
- **HACMP error notification:** Uses AIX error log facility to capture errors and perform the appropriate response.

12.1 Writing scripts for custom events

The HACMP custom event and error notification solution requires writing scripts. Please consider the following:

- Test all possible input parameters.
- Test all conditional branches, e.g., all “if”, “case” branches.
- Handle the error codes and return values of all external commands.
- Provide correct return value: 0 for success, any other number for unsuccessful run.
- Terminate within a reasonable amount of time.
- Test as many scenarios as possible.
- If your script fails, your cluster will fail too.
- A recovery program should be able to recover from an event failure, otherwise the cluster will fail.
- Store your scripts in a convenient separate directory, e.g., /usr/ha.

Important: Your cluster will not continue processing events until your custom pre/post-event script has not finished.

12.2 HACMP pre/post-event commands

There may be some special hardware, software components in your configuration, what HACMP cannot manage by default. In this case you can add your own pre/post-event scripts. Typical application is managing remote disk to disk copy on third party storage subsystems.

For all predefined HACMP event you can define a pre-event, a post-event, a notification method and a recovery command:

- **Pre-event script**: Runs *before* the HACMP event executed
- **Post-event script**: Runs *after* the HACMP event executed
- **Notify method**: The notification method runs before and after the HACMP event. It sends a message to the system administrator about an event is starting or finishing.
- **Recovery command**: Runs only if the HACMP event failed. If the recovery method runs successfully, then the event will success too, regardless of the result of the event script.

HACMP passes a number of arguments for the event scripts, that you can use in your program. Table 12-1 shows pre/post-event script arguments.

Table 12-1 Arguments for pre/post-event script

Script	Arguments
pre-event	event name, *trailing arguments*
post-event	event name, event exit status, *trailing arguments*
notify method	event name, keyword: *start* or *complete*, exit status, if the keyword is complete, *trailing arguments*
recovery command	event name, *trailing arguments*

Trailing arguments: the arguments of the predefined HACMP event script. They are defined in the event command file header. The HACMP predefined event command files are stored in /usr/es/sbin/cluster/events directory. For example the let see the swap adapter event file in Example 12-1:

Example 12-1 /usr/es/sbin/cluster/events/swap_adapter event file

```
/usr/es/sbin/cluster/events# more swap_adapter
#                                                                    #
#       Arguments:      nodename network ip_address1 ip_address2     #
#                                                                    #
#                       ip_address1 - the new available address      #
#                       this script swaps the service adapter to     #
#                       ip_address2 - the failed address             #
```

As you can see, the swap_adapter event has the following arguments:

- nodename
- network name
- ip_address1: the new available address
- ip_address2: the failed address

The same trailing arguments are passed to the user defined pre/post-events.

Attention: Do not modify the built-in event files. This is not supported, nor safe. Always use pre- or post-event scripts.

Selecting the event

HACMP processes the resource groups in parallel by default, which results in fewer cluster events. Some events run only once, while others don't run at all.

There are certain events that run only if the resource group processing mode is set to serial. We advice that you do a few cluster start/stop and failover test and examine the /tmp/hacmp.out file. This will give you an idea about the event you intend to modify really runs in a certain configuration/scenario.

12.2.1 Setting up a pre/post-event scripts

We show you how to set up a pre/post-event script through an example. We have an third party disk subsystem which requires some additional configuration before we can varyon a volume group. We want to run a pre event script for the **get_disk_vg_fs** HACMP event. Also we want to send a notification to the system administrator when this event have started or completed.

Tip: HACMP supports the online configuration of pre/post-event script.

1. Write and carefully test your event script. Copy the files to all nodes under the same path and name. It would ba a good practice to include the files you created under the control of HACMP file collections (5.2 and up).

   In our example we created two shell scripts: a pre-event script (/usr/ha/pre_get_disk_vg_fs, see Example 12-2) to run the OEM disk configuration and a notification command (/usr/ha/notify_get_disk_vg_fs, Example 12-3 on page 557).

Example 12-2 Sample pre-event script (/usr/ha/pre_get_disk_vg_fs)

```
#!/bin/ksh
# Checking the arguments
# Event name
EVENT=$1
# Check the event name
if [ "$EVENT" != "get_disk_vg_fs" ]
then
        echo Ops! We were not supposed to run in event $EVENT!
        exit 0
fi
# Run the OEM disk setup command
/oem_setup_disk
if [ $ ?!= 0 ]
then
        echo Error setting up OEM disk!
        exit -1
fi
exit 0
```

   This script checks the event arguments then calls a program to configure our third party disk subsystem.

Example 12-3 Sample notification method (/usr/ha/notify_get_disk_vg_fs)

```
#!/bin/ksh
# Sample event notification command
# Processing the arguments
# Event name
EVENT=$1
#Event status: start or complete
STATUS=$2
# Result of the event, if status=complete
RESULT=$3
# Notify root that the has event started
if [ "$STATUS" == "start" ]
then
mail -s "Event notification" root <<EOF1
Event $EVENT started.
EOF1
fi
# Notify the root that the event has completed and send the return code
if [ "$STATUS" == "complete" ]
then
mail -s "Event notification" root <<EOF2
Event $EVENT completed, the result is $RESULT
EOF2
fi
exit 0
```

This sample notification program sends an email to the root with the return code of the get_disk_vg_fs event.

2. Define your pre/post-event commands
   a. Start **smit hacmp**.
   b. Select **Extended Configuration**.
   c. Select **Extended Event Configuration**.
   d. Select **Configure Pre/Post-Event Commands**.
   e. Select **Add a Custom Cluster Event**.
   f. Supply the following information:
      - **Cluster Event Name:** short name for this event, later on you will use this name but filename. Our sample pre event is called pre_get_disk_vg_fs.
      - **Cluster Event Description:** short description of your script.
      - **Cluster Event Script Filename:** the full path name of your pre/post-event file. See SMIT screen on Figure 12-1 on page 558.

```
                        Add a Custom Cluster Event

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                              [Entry Fields]
* Cluster Event Name                          [pre_get_disk_vg_fs]
* Cluster Event Description                   [Configure OEM disk]
* Cluster Event Script Filename               [/usr/ha/pre_get_disk_vg_fs]



F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 12-1 Add a custom cluster event

You can define only pre/post-events here. The notification and recovery scripts should be defined individually in SMIT **Change/Show Pre-Defined HACMP** Events panel. See Step 3. for more details.

3. Connect your pre/post-event with the HACMP predefined event:
   a. Start **smit hacmp**.
   b. Select **Extended Configuration**.
   c. Select **Extended Event Configuration**.
   d. Select **Change/Show Pre-Defined HACMP Events**.
   e. Select the event that you want to change. See Figure 12-2 on page 559.

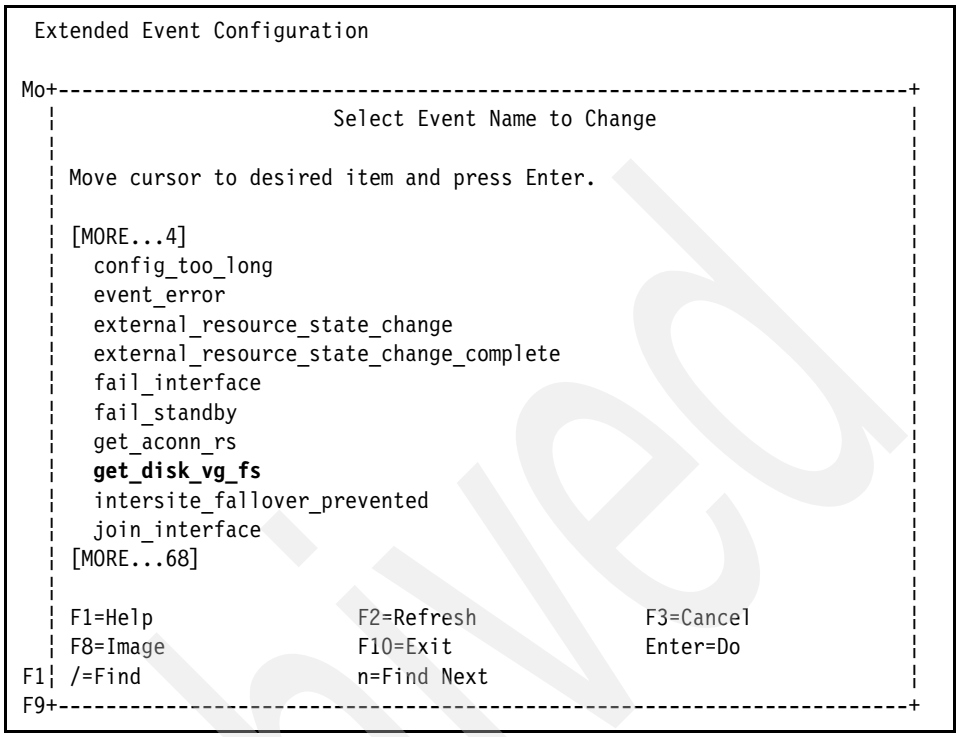

Figure 12-2 Select event to change

f. Enter the following values (see Figure 12-3 on page 560 below):

- **Notify Command** (optional): The full path name of the notification script, if you have one. In our sample, this is /usr/ha/notify_get_disk_vg_fs.
- **Pre-event Command** (optional): The name of the custom cluster event that you want to run as a pre-event. Press F4 for the already defined custom cluster event list. In our example, this is pre_get_disk_vg_fs.
- **Post-event Command** (optional): The name of a custom cluster event that you want to run as a post-event. Press F4 for the already defined custom cluster event list.
- **Recovery Command** (optional): The full path name of the recovery script.
- **Recovery Counter:** Number of times to try running the recovery command. By default it is 0. If this number is greater then zero and the recovery command finished successfully then the event runs again.

```
                        Change/Show Cluster Events

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                              [Entry Fields]
  Event Name                                  get_disk_vg_fs

  Description                                 Script run to acquire>

* Event Command                              [/usr/es/sbin/cluster/>

  Notify Command                             [/usr/ha/notify_get_disk_vg_fs]
  Pre-event Command                          [pre_get_disk_vg_fs]    +
  Post-event Command                         []                      +
  Recovery Command                           []
* Recovery Counter                           [0]                      #

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 12-3 Change a predefined cluster event

4. Verify and synchronize the cluster.

Tip: If you set up HACMP_Files file collection for automatic update then your custom event files will be propagated automatically among the cluster nodes. See 8.2, “File collections SV” on page 356 for more information about File collection.

12.3 Error notification

HACMP error notification is an excellent tool to monitor applications and devices which support and utilize AIX error log facility. When an application or device encounter an error it sends a message to the error logger daemon (errdemon) with the relevant information, like date, device or application name, type of error, error description and other debug information. The errdemon put this data to the error log file. If a HACMP error notification is set up for this type of error, then the errdemon runs the user-defined notify/recovery method.

12.3.1 Automatic error notification

HACMP can configure error notification and recovery action for several resources and error types:

- Rootvg disks
- All disks defined in a HACMP resource group
- SCSI adapters used by HACMP resources or rootvg
- Fibre Channel adapters used by HACMP resources
- SP switch adapter

The hard, non-recoverable errors are configured to initiate a takeover by **/usr/es/sbin/cluster/diag/cl_failover** command. For the non-critical errors the **/usr/es/sbin/cluster/diag/cl_logerror** sends an email to the root and logs the error in the /tmp/hacmp.out file.

Disk monitoring consideration

HACMP monitors only “traditional” standalone disks, like built-in SCSI disk or SSA storage. This kind of disks use the AIX standard error log labels, such as DISK_ERR1 or SCSI_ERR3 when an error is detected. Additionally HACMP can monitor fully mirrored volume groups regardless of the disk type. In this case when the loss of quorum is detected (LVM_SA_QUORCLOSE error log entry) HACMP can initiate a takeover.

ESS, FAStT, DS4000 Series and other modern disk subsystems are designed with high availability in mind, so they have built-in error recovery mechanisms, like hardware RAID function and multipath connection. Because they have their own device drivers, they use different error log entry labels in case of a hardware error. In 12.3.3, “Monitoring shared disks with HACMP error notification” on page 566, we have an example how to monitor such a disk subsystem.

Set up automatic error notification

Attention: You cannot configure automatic error notification while the cluster is running.

1. Start **`smit hacmp`**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Configure Automatic Error Notification**.
5. Select **Add Error Notify Methods for Cluster Resources**.

HACMP will add automatic error notification on all nodes.

Attention: You have to add again automatic error notification every time you verify and synchronize the cluster.

List automatic error notification

1. Start **smit hacmp**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Configure Automatic Error Notification**.
5. Select **List Error Notify Methods for Cluster Resources**.

Figure 12-4 shows an example list of automatic error notification.

```
                                COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.

p650n01:
p650n01: HACMP Resource         Error Notify Method
p650n01:
p650n01: hdisk0         /usr/es/sbin/cluster/diag/cl_failover
p650n01: scsi0          /usr/es/sbin/cluster/diag/cl_failover
p650n01: fcs0           /usr/es/sbin/cluster/diag/cl_logerror
p650n02:
p650n02: HACMP Resource         Error Notify Method
p650n02:
p650n02: hdisk0         /usr/es/sbin/cluster/diag/cl_failover
p650n02: scsi0          /usr/es/sbin/cluster/diag/cl_failover
p650n02: fcs0           /usr/es/sbin/cluster/diag/cl_logerror

F1=Help             F2=Refresh          F3=Cancel           F6=Command
F8=Image            F9=Shell            F10=Exit            /=Find
n=Find Next
```

Figure 12-4 List of automatic error notification

Remove automatic error notification

1. Start **smit hacmp**.
2. Select **Problem Determination Tools**.

3. Select **HACMP Error Notification**.
4. Select **Configure Automatic Error Notification**.
5. Select **Remove Error Notify Methods for Cluster Resources**.
6. Press Enter to confirm.

12.3.2 Using error notification

You can add manually error notification object for all possible kind of resource and error types. When you create a notification object, you define a set of rules or criteria for what kind of error you like to monitor. This rules can include the error log label, resource name, etc. If an error occurs, the errlogger match the specified criteria in the error notification ODM class, and if any of them matches, then errdemon will run the appropriate notification method.

Add a notify method

1. Start **`smit hacmp`**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Add a Notify Method**.
5. Define the notification object:
   - **Notification Object Name**
     User supplied name that identifies the error notification object.
   - **Persist across system restart?**
     Yes: the error notification will be used permanently. **No**: the error notification will be used until the next reboot only.
   - **Process ID for use by Notify Method**
     The error notification will be send on behalf of the selected process ID. The default is 0 (root), we suggest that you that use 0 here. You should set Persist across system restart to No if you specify any non-zero process ID here.
   - **Select Error Class**
     None: no error class to match, **All**: match all error classes, **Hardware**: hardware errors, **Software**: software errors, **Errlogger**: operator notifications and messages from the **`errlogger`** program.
   - **Select Error Type**
     None: no error type to match, **All**: match all error types, **PEND**: impending loss of availability, **PERF**: performance degradation, **PERM**: permanent errors, **TEMP**: temporary errors, **UNKN**: unknown error type.

– **Match Alertable errors?**
This field is provided for use by system management application's alert agents. **None**: ignore this entry, **All**: alert all errors, **TRUE**: match alertable errors, **FALSE**: match non-alertable errors. If you don't have remote management application, leave this field on **None**.

– **Select Error Label**
Press F4 to select the error label. See the /usr/include/sys/errids.h file for a short description of the error labels.

– **Resource Name**
The name of the failing resource. For a hardware error class, this is the device name. For software class, this is the name of the failing executable. Select **All** to match all resource type.

– **Resource Class**
For the hardware resource class, this is the device class. It is not applicable for software errors. Specify **All** to match all resource classes.

– **Resource Type**
The device type by which a resource is know in devices object. It is only applicable for hardware errors. Specify **All** to match all resource classes.

– **Notify Method**
The full-path name of the program to run whenever an error is logged, that matches the above defined criteria. You can pass the following variables to the executable:
$1: Error log sequence number
$2: Error identifier
$3: Error class
$4: Error type
$5: Alert flag
$6: Resource name of the failing device
$7: Resource type of the failing device
$8: Resource class of the failing device
$9: Error log entry label

6. Press Enter to create the error notification object.

Figure 12-5 on page 565 shows how to add a notify method in SMIT.

```
                                Add a Notify Method

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
* Notification Object Name                             [disk_error]
* Persist across system restart?                        No +
  Process ID for use by Notify Method                  [] +#
  Select Error Class                                    Hardware +
  Select Error Type                                     PERM +
  Match Alertable errors?                               None +
  Select Error Label                                   [VPATH_PATH_REMOVED] +
* Resource Name                                        [All] +
* Resource Class                                       [All] +
* Resource Type                                        [All] +
* Notify Method                                        [/usr/ha/vpath_error]

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 12-5 Add an error notification object

Change / show a notify method

1. Start **smit hacmp**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Change/Show a Notify Method**.
5. Select notify method to change from the pop-up list.
6. Change the notification object. For the explanation of the fields see“Add a notify method” on page 563.

Remove a notify method

1. Start **smit hacmp**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Remove a Notify Method**.
5. Select notify method to remove from the pop-up list.
6. Press Enter to remove.

Attention: HACMP verification and synchronization does not support error notification. You have to configure it manually on all nodes.

Test a notify method

You can easily test your notification objects. HACMP can emulate error log entry with the selected error label. The error label appears in the error log and the notification method is run by errdemon.

1. Start **`smit hacmp`**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Emulate Error Log Entry**.
5. Select the error label or notify method name from the pop-up list. Only that notify methods shows up here that has an error label defined.
6. SMIT shows up the error label, notification object name and notify method Press Enter to confirm error log entry emulation.

12.3.3 Monitoring shared disks with HACMP error notification

In this example we show how you can use HACMP error notification for monitoring your shared disks. This example is based on our hardware configuration. You can easily modify this example to work with your own disk subsystem.

Our shared disks reside in an FC connected ESS 2105-800 disk subsystem. We use SDD device driver, the volume groups are on vpath disk devices. Our goal is to monitor the vpath devices. If a vpath link (hdiskX) fails, we want to send a notification to the system administrator. If a vpath disk device (vpathX) fails, then we want to start a HACMP takeover.

In our example we have a shared volume group, called db2vg. This is located on vpath1 device which has two underlying paths: hdisk3 and hdisk11 (same LUN via two different HBAs).

Selecting the error label

First check what does happen if we remove the fibre channel cables from the adapters for a few minutes. Of course, our I/O stops and the SDD device driver logs the errors into the AIX error log.

Important: We do not recommend that you perform this test on purpose, specially if your application is running with real data. Even though HACMP and SDD can handle this failure, you have also to make sure that your application is able to handle this type of failure.

Now let's have a look on our error log (**errpt** command), see Figure 12-4:

Example 12-4 AIX error log

```
#errpt
3074FEB7   0624164705 T H fscsi0         ADAPTER ERROR
3074FEB7   0624164705 T H fscsi0         ADAPTER ERROR
D7B7FF7E   0624164705 I O SYSJ2          USER DATA I/O ERROR
D7B7FF7E   0624164705 I O SYSJ2          USER DATA I/O ERROR
613E5F38   0624164705 P H LVDD           I/O ERROR DETECTED BY LVM
F4D25312   0624164705 P H vpath1         UNABLE TO COMMUNICATE WITH DEVICE
A7212C7B   0624164705 P H hdisk11        DEVICE ACCESS PROBLEM
A7212C7B   0624164705 P H hdisk3         DEVICE ACCESS PROBLEM
3074FEB7   0624164705 T H fscsi0         ADAPTER ERROR
3074FEB7   0624164705 T H fscsi0         ADAPTER ERROR
3074FEB7   0624164705 T H fscsi0         ADAPTER ERROR
3074FEB7   0624164705 T H fscsi0         ADAPTER ERROR
B8113DD1   0624164705 T H fcs0           LINK ERROR
AFA89905   0624163505 I O grpsvcs        Group Services daemon started
```

There are two kind of error log entries of interest (see Example 12-4):

- vpath1: **unable to communicate with device**: we get this error, when all paths belonging to a certain vpath device are failed. The vpath device fails too, so the volume group is no more accessible. This error very seriously affects the cluster, so we want to have a takeover in this case.
- hdisk11 and hdisk13: **device access problem**: it means that one of the communication path to a vpath device is failed. The vpath device is still accessible but the performance and the availability is affected. In this case a notification to the sysadmin is adequate response.

All of this errors occurred only once, and they correctly pinpointed the failed device. The other related error messages appears several times and do not show the real cause of the problem. Now let see this error log entries in detail: VPATH_OUT_SERVICE (Example 12-5).

Example 12-5 Vpath error log entries

```
#errpt -a|more
LABEL:          VPATH_OUT_SERVICE
IDENTIFIER:     F4D25312
```

```
Date/Time:       Fri Jun 24 16:47:29 CDT
Sequence Number: 222
Machine Id:      000197BA4C00
Node Id:         p650n02
Class:           H
Type:            PERM
Resource Name:   vpath1
Resource Class:  disk
Resource Type:   vpath
Location:

Description
UNABLE TO COMMUNICATE WITH DEVICE

Probable Causes
DISK
SCSI ADAPTER
SCSI CABLE

Failure Causes
DISK
SCSI ADAPTER
CABLE LOOSE OR DEFECTIVE

Recommended Actions
PERFORM PROBLEM DETERMINATION ON SCSI TARGET DEVICE
PERFORM PROBLEM DETERMINATION ON HOST SCSI ADAPTER
REPLACE SCSI CABLE

Detail Data
SENSE DATA
021F 4664 0000 0000 0029 0001 0000 0004 0000 0000 0000 0000 0000 0000 0001
```

See VPATH_DEVICE_OFFLIN error log record in Example 12-6:

Example 12-6 Vpath device error log entry

```
#errpt -a|more
LABEL:          VPATH_DEVICE_OFFLIN
IDENTIFIER:     A7212C7B

Date/Time:       Fri Jun 24 16:47:29 CDT
Sequence Number: 221
Machine Id:      000197BA4C00
Node Id:         p650n02
Class:           H
Type:            PERM
Resource Name:   hdisk11
```

```
Resource Class:  disk
Resource Type:   2105
Location:        U0.1-P2-I4/Q1-W5005076300C99589-L5209000000000000
VPD:
        Manufacturer................IBM
        Machine Type and Model......2105800
        Serial Number...............20922513
        EC Level....................1.62
        Device Specific.(Z0)........10
        Device Specific.(Z1)........002C
        Device Specific.(Z2)........0013
        Device Specific.(Z3)........16602
        Device Specific.(Z4)........05
        Device Specific.(Z5)........00

Description
DEVICE ACCESS PROBLEM

Probable Causes
DISK
SCSI ADAPTER
SCSI CABLE

Failure Causes
DISK
SCSI ADAPTER
CABLE LOOSE OR DEFECTIVE

        Recommended Actions
        PERFORM PROBLEM DETERMINATION ON SCSI TARGET DEVICE
        PERFORM PROBLEM DETERMINATION ON HOST SCSI ADAPTER
                REPLACE SCSI CABLE

Detail Data
SENSE DATA
0000 0000 0000 0000 0028 0002 0000 0002 0000 0680 0000 0001 0000 0000
```

From the **errpt -a** output we can collect all the required information (they are bold in our examples) to configure HACMP error notification.

Define the error notification objects

Set up error notification in SMIT for vpath failure on all cluster hosts. We use the values from **errpt -a** command shown in Example 12-5 on page 567.

1. Start **smit hacmp**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.

4. Select **Add a Notify Method**.
5. Define the notification method with the following values (see SMIT screenshoot on Figure 12-6):
   - Notification Object Name: **vpath_failed**
   - Persist across system restart? Yes/No: **Yes**
   - Process ID for use by Notify Method: **0** (root) would be fine
   - Select Error Class: **Hardware** (See “Class: H”).
   - Select Error Type: **Permanent** (See “Type: PERM”).
   - Match Alertable errors?: **None**
   - Select Error Label: **VPATH_OUT_OF_SERVICE**
   - Resource Name: **All**. We want to setup this error notification for all vpath devices.
   - Resource Class: **disk** (See “Resource Class: disk”)
   - Resource Type: **vpath** (See “Resource Type: vpath”)
   - Notify Method: **/usr/es/sbin/cluster/diag/cl_failover $6 $9**. This script is provided by HACMP and it starts the HACMP **errnotify event error** which performs a takeover. Argument $6 is the device name, $9 is the error label.

```
                              Add a Notify Method

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
* Notification Object Name                           vpath_failed
* Persist across system restart?                     Yes +
  Process ID for use by Notify Method                [0] +#
  Error Class                                         Hardware +
  Class Type                                          PERM +
  Match Alertable errors?                             None +
  Select Error Label                                 [VPATH_OUT_SERVICE] +
  Resource Name                                      [All] +
  Resource Class                                     [disk] +
  Resource Type                                      [vpath] +
* Notify Method                                      [/usr/es/sbin/cluster/>

F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 12-6 Configuring error notification for vpath error

Set up error notification in SMIT for vpath path communication error on all nodes. We use the values from **errpt -a** command shown in Example 12-6 on page 568.

1. Start **smit hacmp**.
2. Select **Problem Determination Tools**.
3. Select **HACMP Error Notification**.
4. Select **Add a Notify Method**.
5. Define the notification method with the following values:
   - Notification Object Name: **path_offline**
   - Persist across system restart? Yes/No: **Yes**
   - Process ID for use by Notify Method: **0** (root) would be fine
   - Select Error Class: **Hardware** (See “Class: H”).
   - Select Error Type: **Permanent** (See “Type: PERM”).
   - Match Alertable errors?: **None**
   - Select Error Label: **VPATH_DEVICE_OFFLIN**
   - Resource Name: **All**. We want to setup this error notification for all vpath devices.
   - Resource Class: **disk** (See “Resource Class: disk”)
   - Resource Type: **2105** (See “Resource Type: 2105”)
   - Notify Method: **/usr/ha/notify_root $6 $9**. See Example 12-7 for the source of /usr/ha/notify_root script. We pass argument $6 and $9: the device name and the error label. The script sends an email to the root user regarding the encountered error.

Example 12-7 /usr/ha/notify_root script

```
#!/bin/ksh
# Processing the arguments
# device name
DEVICE=$1
# Error log label
ERRLABEL=$2
mail -s "Vpath communication error" root <<EOF
The $DEVICE device encountered an $ERRLABEL error. The disk subsystem
performance and availability is degraded. Perform problem determination!
EOF
exit 0
```

Test the error notification

There are two ways for testing our error notification object:

1. Unplug the FC cables from the HBA adapters one by one. This will really test our solution, if everything configured properly HACMP will start a takeover.
2. Simulate the error log entries:

   g. Start **`smit hacmp`**.

   h. Select **Problem Determination Tools**.

   i. Select **HACMP Error Notification**.

   j. Select **Emulate Error Log Entry**.

   k. Select **VPATH_OUT_OF_SERVICE** or **VPATH_DEVICE_OFFLIN** error label from the pop-up list.

   l. SMIT shows up on the error label, notification object name and notify method. Press Enter to confirm error log entry emulation. See SMIT Figure 12-7.

   Now HACMP sends the requested error log entry to the AIX error log daemon and starts the notify method. If everything is configured properly, HACMP will initiate a takeover. Monitor /tmp/hacmp.out file to see what's going on.

```
                        Emulate Error Log Entry

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  Error Label Name                                  VPATH_OUT_SERVICE
  Notification Object Name                          vpath_failed
  Notify Method                                     /usr/es/sbin/cluster/>



F1=Help           F2=Refresh          F3=Cancel           F4=List
F5=Reset          F6=Command          F7=Edit             F8=Image
F9=Shell          F10=Exit            Enter=Do
```

Figure 12-7 Emulate VPATH_OUT_SERVICE error log entry

13

Storage related considerations

This chapter contains more detailed information about the following:

- Volume group types
- Disk reservations
- Forced varyon of volume groups
- Fast disk takeover
- Disk heartbeat

13.1 Volume group types

It is important to understand the different types of volume groups and how HACMP utilizes each type. We cover the following volume group types:

- Enhanced concurrent
- Non-concurrent
- Concurrent
- RAID concurrent

There are other additional volume group attributes that can coexist with these listed above. Such as, “big” and “scalable”. Generally speaking, you can combine this attributes with the additional types listed above. For example you can have a big enhanced concurrent volume group. However, these additional attributes do not affect how HACMP activates the volume groups. HACMP activates a big enhanced concurrent volume group using the same method as though it were only an enhanced concurrent volume group.

13.1.1 Enhanced concurrent

Enhanced concurrent volume groups were first introduced in AIX 5.1. Unlike concurrent that is for SSA only, it is supported for use on any disk subsystem that is supported in a shared, AIX, pSeries, HACMP configuration. In AIX 5.2 and above, enhanced concurrent volume group type is the only concurrent type available.

Enhanced concurrent volume groups use the Group Services Concurrent Logical Volume Manager (gsclvmd) daemon, which communicates over IP to other cluster member nodes.

Utilizing gsclvmd, most LVM changes can be made dynamically, even from the command line. For these dynamic changes to work correctly, it is required to have gsclvmd, topsvcs, groupsvcs, and emsvcs running while performing maintenance. This is easily done by having the HACMP cluster up and running with your volume groups online in concurrent mode.

Note: C-SPOC is the recommended best practice for all cluster LVM administration. Unlike the command line, it is not dependent on cluster services (other than clcomdES) to be running on the member nodes.

Enhanced concurrent volume groups can be used in both concurrent and non-concurrent environments. Additional facilities within HACMP (i.e fast disk takeover and disk heartbeat) are dependent on it.

Attention: When configuring enhanced concurrent volume groups in the cluster, ensure that multiple networks (IP and non-IP) exist for communication between the nodes in the cluster, to avoid cluster partitioning. When fast disk takeover is used, the normal SCSI reserve is not set to prevent multiple nodes from accessing the volume group.

Existing non-concurrent volume groups can be converted to enhanced concurrent without losing any additional storage space. The volume group must be online to change it and can be done by executing **chvg -c vgname**. For this change to take affect on other nodes, the volume group must be offline, and either exported and re-imported, or you can use the *learn* option of importvg via **importvg -L vgname pvname**

To create a new enhanced concurrent volume group on a local node from the command line, simply run **mkvg -C vgname pvname.**

You can determine if a volume is enhanced concurrent by running **lsvg vgname** and checking the “Concurrent:” field; it should says “Enhanced Capable”, as shown in Figure 13-1.

```
Maddi / > lsvg app1vg
VOLUME GROUP:   app1vg                  VG IDENTIFIER: 0022be2a00004c48
VG STATE:       active                  PP SIZE:        16 megabyte(s)
VG PERMISSION:  read/write              TOTAL PPs:     1190
MAX LVs:        256                     FREE PPs:      1180
LVs:            0                       USED PPs:       10
OPEN LVs:       0                       QUORUM:         2
TOTAL PVs:      2                       VG DESCRIPTORS: 3
STALE PVs:      0                       STALE PPs:      0
ACTIVE PVs:     2                       AUTO ON:        no
Concurrent:     Enhanced-Capable        Auto-Concurrent: Disabled
VG Mode:        Non-Concurrent
MAX PPs per PV: 1016                    MAX PVs:        32
LTG size:       128 kilobyte(s)         AUTO SYNC:      no
HOT SPARE:      no                      BB POLICY:      relocatable
```

Figure 13-1 Enhanced concurrent volume group example

13.1.2 Non-concurrent

A non-concurrent volume group is the default when creating a new volume group. It is also referred to as a *standard volume group*. The inherent nature of non-concurrent volume groups is that the volume group will not be accessed by

more than one system at any time. Full read/write access is possible only by the system that activated the volume group with varyonvg vgname.

Non-concurrent is not an LVM designated type of volume group. It is a designation of the mode of operation in which the volume group is to be used. When running the lsvg command against a volume group, you can tell it is a non-concurrent volume group by the omission of the “Concurrent” field that is shown in Figure 13-1 on page 575

13.1.3 Concurrent

Note: AIX 5.1, using the 32-bit kernel, is the last version to support concurrent volume groups. AIX 5.2 and above utilize “Enhanced Concurrent” volume groups as stated in section 13.1.1, “Enhanced concurrent” on page 574.

This type of volume group, also referred to as “Concurrent Capable”, is specific to SSA disks in a “concurrent access” configuration with HACMP. This combination provided the first true concurrent mode volume group.

The unique serial connectivity of SSA disks allows communications access over something called the covert channel. This covert channel is utilized by the Concurrent Logical Volume Manager (CLVM). CLVM is capable of keeping LVM related ODM information in sync automatically using the CLVM daemon (clvmd). This allows for online LVM maintenance of volume groups.

Clvmd gets started automatically when the volume group is varied on in concurrent mode via **`varyonvg -c.`**

13.1.4 RAID concurrent

Note: Raid concurrent volume groups rarely exist anymore and are considered obsolete today.

A raid concurrent volume group is a non-concurrent volume group that is assigned to a concurrent (or now called “online on all nodes” in HA 5.2 and above) resource group within HACMP. There is not a designation within LVM that makes it a type “raid concurrent”. Raid concurrent is actually an internal designation from HACMP.

When a concurrent resource group is brought online, the volume group type is checked. If the volume group is a type non-concurrent, then HACMP must determine what type of device(s) comprise the volume group. It is important to

know the device type as only certain storage devices were supported in this configuration.

The supported devices are stored in **/usr/es/sbin/cluster/diag/clconraid.dat** When the device type is found, HACMP continues to bring the volume group online in full read/write mode to each member node by utilizing the **convaryonvg** command. You may see references of concurrent mode as "mode 3" in both older documentation and log files like /tmp/hacmp.out.

When a raid concurrent volume group is online in concurrent access mode, it is not possible to make LVM related changes (including C-SPOC). To make any LVM changes involves bringing the volume groups offline on all nodes, then online to only one node, make the desired changes, re-import the volume group to every other member node again. This obviously is not convenient as almost any LVM change requires an outage.

13.2 Disk reservations

When a volume group is varied on in AIX, a disk reserve is placed against each member disk. This is to ensure that no other systems can access these drives to maintain data integrity.

These reserves are often called SCSI reserves and they are based on SCSI standards. Most of today's newest FC disks are using FSCSI protocol and still utilize a SCSI reserve.

The SCSI standards define two different types of reservations:

- SCSI-2 "traditional" reservation
- SCSI-3 persistent reservation (PR)

A SCSI-2 reservation allows access along a single path only, so this reservation could not be used for general multipathing access to the storage. SCSI-2 reservations are not persistent and they do not survive node reboots.

SCSI-3 PR (persistent reservation) supports device access through multiple nodes, while at the same time blocking access to other nodes. SCSI-3 PR reservations are persistent across SCSI bus resets or node reboots, and they also support multiple paths from host to disk.

SCSI-3 PR uses a concept of registration and reservation. Systems that participate, register a 'key' with the SCSI-3 device. Each system registers its own key. Registered systems can then establish a reservation. With this method, blocking write access is as simple as removing the registration from a device. When a system wants to eject another system, it issues a 'pre-empt and abort'

command, which ejects another node. Once a node is ejected, it has no key registered so it cannot eject others. This method effectively avoids the split-brain condition.

Another benefit of the SCSI-3 PR method is that since a node registers the same key on each path, ejecting a single key blocks all I/O paths from that node. For example, SCSI-3 PR is implemented by EMC Symmetrix, Sun™ T3, and Hitachi Storage systems. ESS SDD uses persistent reservations while LVM cmds use "traditional" reservation.

This reserve will normally stay in place until removed by a varying off the volume group. Even if the AIX system is halted, if the disks maintain power, the reserve normally stays set. For this reason, HACMP must break disk reserves during fallover to bring the volume group online to the standby node.

Disk reserves are not used for concurrent, or enhanced concurrent mode volume groups used when the volume groups are online in concurrent access mode. This is also true when using enhanced concurrent volume groups in a fast disk takeover configuration.

More information about fast disk takeover can be found in 13.4, “Fast disk takeover” on page 579.

13.3 Forced varyon of volume groups

This ability is very important when using mirrored volume groups. It is an attribute of the varyonvg command is represented by the -f flag. Utilizing this flag enables a volume group to be brought online even when a quorum of disks is not available. During the varyon process, each logical volume is checked and at least one complete copy of each must be found for the varyon to succeed.

It is standard practice to disable the quorum setting when configuring a mirrored volume group. When quorum is disabled, it allows the volume group to stay online even after disk failure(s). As long as one disk is available the volume group will stay online. However, to initially varyon the volume group all disks must be available. In order to varyon on a volume group which does not have all member disks available, the force attribute must be used.

This setting is most commonly used when mirroring across storage subsystems, and/or mirroring between locations via cross-site LVM mirroring. This allows for site redundancy so in the event of a site outage, (a site consists of a server and one copy of storage), a server at the remote site can active the volume group off of the local lvm copy. More information about cross-site LVM mirroring can be found in Chapter 16.

In HACMP 5.1 and above, the user may set this attribute in the resource group definition. Prior to 5.1 users could set the environment variable, `HACMP_MIRROR_VARYON=true` in /etc/environment, or use custom event scripts.

13.4 Fast disk takeover

This section explains the following in regards to fast disk takeover:

- Prerequisites
- How fast disk takeover works
- How to enable fast disk takeover
- Advantages
- Disadvantage

13.4.1 Prerequisites

The following are required to implement fast disk takeover

- HACMP V5.1 or higher
- Cluster.es.clvm.rte (HACMP CRM component)
- AIX 5.2 or higher
- Bos.clvm.enh 5.2.0.11 or higher
- Enhanced concurrent vgs in non-concurrent resource group(s)

13.4.2 How fast disk takeover works

Historically, disk takeover has involved breaking a scsi reserve on each disk device in a serial fashion. The amount of time it takes to break the reserve varies by disk type. In a large environment with hundreds of disks, this can add significant amount of time to the fallover.

Fast disk takeover reduces total fallover time by providing faster acquisition of the disks without having to break scsi reserves. It utilizes enhanced concurrent volume groups, and additional LVM enhancements provided by AIX 5.2.

AIX 5.2 introduced the ability to varyon an enhanced concurrent volume group in two different modes:

- Active Mode
- Passive Mode

Active mode is similar to a non-concurrent volume group being varied online with a simple **varyonvg**. It provides full read/write access to all logical volumes, fileystems and supports all LVM operations.

Passive mode is the LVM equivalent of disk fencing. Passive mode only allows read ability of the VGDA and the first 4K of each logical volume. It does *not* allow read/write access to filesystems or logical volumes. It also does not support LVM operations.

When a resource group, containing the volume group, is brought online it is first varied on in passive mode and then it is varied on in active mode. The active mode state only applies to the current resource group owning node. As any other resource group member node comes online, the volume group is varied on in passive mode.

When the owning/home node fails, the failover node simply changes the volume group state from passive mode to active mode through the LVM. This change takes ~10 seconds and is at the volume group level. It can take longer with multiple volume groups with multiple disks per volume group. However, the time impact is minimal compared to the previous method of breaking scsi reserves.

The active and passive mode flags to the varyonvg are not documented as they should *not* be used outside an HACMP environment. It can, however, be easily found in the hacmp.out log.

Active mode varyon command:

```
varyonvg -n -c -A app2vg
```

Passive mode varyon command:

```
varyonvg -n -c -P app2vg
```

Important: Do not run these commands without cluster services running

To determine if the volume group is online in active or passive mode verify the "VG PERMISSION" field from the **lsvg** output as shown in Figure 13-2 on page 581.

There are other distinguishing LVM status features you will notice for volume groups that are being utilized in a fast disk takeover configuration. For example, the volume group will show online in concurrent mode on each active cluster member node via the **lspv** command. However, **lsvg -o** will only report the volume group online to the node that has it varied on in active mode. An example of how passive mode status is reported is shown in Figure 13-2 on page 581.

```
Melany / > lspv |grep vpath
vpath0          0022be2a8607249f            app2vg
vpath1          0022be2a8617133e            None
vpath2          0022be2a86607918            app1vg          concurrent
vpath3          0022be2a8662c1a4            None
vpath4          0022be2a8662ce0e            app2vg
vpath5          0022be2a8662dfa8            None
vpath6          0022be2a8662f794            None
vpath7          0022be2a86630978            app1vg          concurrent

Melany / > lsvg -o
rootvg

Melany / > lsvg app1vg
VOLUME GROUP:   app1vg                  VG IDENTIFIER:  0022be2a00004c48
VG STATE:       active                  PP SIZE:        16 megabyte(s)
VG PERMISSION:  passive-only            TOTAL PPs:      1190
MAX LVs:        256                     FREE PPs:       1180
LVs:            0                       USED PPs:       10
OPEN LVs:       0                       QUORUM:         2
TOTAL PVs:      2                       VG DESCRIPTORS: 3
STALE PVs:      0                       STALE PPs:      0
ACTIVE PVs:     2                       AUTO ON:        no
Concurrent:     Enhanced-Capable        Auto-Concurrent: Disabled
VG Mode:        Concurrent
Node ID:        6                       Active Nodes:
MAX PPs per PV: 1016                    MAX PVs:        32
LTG size:       128 kilobyte(s)         AUTO SYNC:      no
HOT SPARE:      no                      BB POLICY:      relocatable
```

Figure 13-2 Passive mode volume group status

13.4.3 How to enable fast disk takeover

There is not an actual option or flag within the HACMP cluster configuration specifically related to fast disk takeover. It is a logical relationship on how the cluster is configured.

The shared volume groups must be enhanced concurrent volume groups. These volume groups are then added as resources to a non-concurrent mode style resource group. The combination of these two things is how HACMP determines to use the fast disk takeover method of volume group acquisition.

When a non-concurrent style resource group is brought online, HACMP checks one of the volume group member disks to see if it is an enhanced concurrent

volume group or not. HACMP determines this by running **lqueryvg -p devicename -X**. If the return output is 0, then it is a regular non-concurrent volume group. If the return output is 32, then it is an enhanced concurrent volume group.

In Figure 13-3, hdisk0 is a rootvg member disk which is non-concurrent. Vpath0 is an enhanced concurrent volume group member disk.

```
Maddi / > lqueryvg -p hdisk0 -X
0
Maddi / >lqueryvg -p vpath0 -X
32
```

Figure 13-3 Example of how HACMP determines volume group type

In AIX 5.1, there were three different values that could be returned. The value of 16 could be returned when using SSA concurrent volume groups. This volume group type is obsolete in AIX 5.2 and above.

13.4.4 Advantages

There are at least two pros of using fast disk takeover:

- Faster disk acquisition time
- LVM ODM synchronization

We have already explained the first benefit listed above in “How fast disk takeover works” on page 579. The other, LVM ODM synchronization, is directly related to using enhanced concurrent volume groups and having the gsclvmd daemon running with cluster services.

When all member nodes of an enhanced concurrent volume group are online in an active cluster, LVM related changes executed on the home node (via command line or via SMIT) are automatically synchronized across the other members. This action greatly reduces the possibility of mismatched volume group information on each node.

However, this advantage itself is not considered a best practice in an HACMP environment. C-SPOC is the recommended method of keeping LVM related ODM information in sync across cluster nodes. C-SPOC has the following advantages over the stand alone LVM ODM synchronization:

1. Independent of cluster node status (active/inactive)
2. Can be used for JFS related changes
3. Updates vgda time stamp files on cluster nodes

13.4.5 Known issues

One current disadvantage of using fast disk takeover is you cannot stop cluster services using the **forced** option. When stopping a cluster via **smit clstop** HACMP checks to see if the local node is currently hosting any fast disk takeover volume groups. If so, then the forced option will not appear in the menu.

The reason is, the volume groups would be left online and they are dependent on gsclvmd. Gsclvmd is dependent on groups services. If you stop cluster services, you stop the services needed to maintain volume group consistency. This leaves a possible exposure to the volume groups.

In the later versions of HACMP it is not as common to use the forced down option. However, there are certain maintenance periods it can be desirable to use to reduce the overall down time of application services.

This limitation is well known, and development is working on possibilities to hopefully have this removed in the future. However, there are no guarantees that this limitations will ever be removed.

13.5 Disk heartbeat

In this section we discuss the following concerning disk heartbeat:

1. Overview
2. Prerequisites
3. Performance considerations
4. Configuring (two node example)
5. Testing
6. Monitoring

13.5.1 Overview

Disk heartbeat is another form of non-IP heartbeat that utilizes the existing shared disks of any disk- type. This feature, which was introduced in HACMP V5.1, is quickly becoming the preferred method of non-IP heartbeat as it eliminates the need for serial cables and/or 8-port asynchronous adapters. It also can easily accommodate greater distances between nodes when using a SAN environment.

This feature requires using enhanced concurrent volume groups to allow access to the disk by each node as needed. It utilizes a special reserved area on the disks to read and write the heartbeat data. Since it uses a reserved area, it allows the use of existing data volume groups without losing any additional storage space. However, please be aware of “Performance considerations” on page 584

It is possible to use a dedicated disk/LUN for the purpose of disk heartbeat. However, since disk heartbeat uses the reserved space, the remaining data storage area is unused. The bigger the disk/LUN you use solely for this purpose, the more space that is wasted. However, you could use it later for additional storage space if needed.

Just like other non-IP networks, a disk heartbeat network is a point-to-point network. If more than two nodes exist in your cluster, you will need a minimum of ***N*** number of non-IP heartbeat networks. Where ***N*** represents the number of nodes in the cluster. For example, a 3 node cluster requires at least 3 non-IP heartbeat networks.

13.5.2 Prerequisites

The following software and hardware is required:

- HACMP V5.1 or higher
- Cluster.es.clvm.rte (HACMP CRM component)
- AIX 5.2 or higher (ML2 recommended)
- RSCT 2.3.1.1 or higher (2.3.3.1 recommended)
- Bos.clvm.enh (required for enhanced concurrent vg support)
- Shared disks (configured as enhanced concurrent volume groups)

13.5.3 Performance considerations

Most modern non-raid disks can perform ~100 seeks per second. The sectors used for disk heart beating are part of the VGDA. The VGDA is located at the outer edge of the disk, and may not be near the application data. This means t every time there is a disk heartbeat that a seek will be performed. Disk heartbeating will typically (with the default parameters) require four (4) seeks per second. That is, each of two nodes will write to the disk and read from the disk once/second, for a total of 4 IOPS. When choosing a disk to be used for disk heartbeat, it is recommended to use disk that has fewer than 60 seeks per second. The **`filemon`** tool can be used to monitor the seek activity on a disk.

If you choose to use a disk that has an I/O load that is above the recommended value, then it is recommended to change the failure detection rate of the disk heartbeat network to *slow*. More information can be found in section 14.2

The stated recommendation is based on non-raid (or JBOD) storage. The technology of the disk subsystem affects the overall recommendation. For example:

- If the disk is part of an enterprise class storage subsystem with large amounts of write cache, like ESS, then the seeks can be much higher.
- If the disk used for heart beating is part of a RAID set, or RAID subset, with little or no caching, the disk would support fewer seeks, due to the extra activity required by RAID operations. Check with the manufacturer to determine how many seeks the specific unit can support.

13.5.4 Configuring disk heartbeat

This example consists of a two-node cluster (nodes Justen and Christie) with a shared ESS vpath devices (vpath0 and vpath3 respectively) to be used as the disk heartbeat device. Both vpaths are already configured as member disks of an enhanced concurrent volume group.

There are two different methods to configure a disk heartbeat device:

- Use the discovery method
- Use the predefined devices method

For this example we will use the predefined devices method. When using this method it is necessary to create a diskhb network first, then assign the disk-node pair devices to the network.

The key information needed, before continuing with the predefined method, is knowing exactly what the devices names are on each node. It is not necessary that the names match as demonstrated in our example. The devices can be matched by the pvid on each node by running **`lspv`** on each system.

Note: When using the discovery method, HACMP matches the devices automatically and provides a pick list to choose from.

Create the diskhb network as follows:

In **`smit hacmp`** → **`Extended Configuration Extended Topology Configuration`** → **`Configure HACMP Networks`** → **`Add a Network to the HACMP cluster`** → **`choose diskhb.`** Enter the desired network name (defaults to net_diskhb_01) as shown in Figure 13-4 on page 586.

```
                    Add a Serial Network to the HACMP Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
* Network Name                                         [net_diskhb_01]
* Network Type                                          diskhb
```

Figure 13-4 Adding diskhb network

Now add two communication devices, one for each node, to the disk heartbeat network created in the previous step.

In **smit hacmp** → **Extended Configuration** → **Extended Topology Configuration** → **Configure HACMP Communication Interfaces/Devices** → **Add Communication Interfaces/Devices** → **Add Pre-Defined Communication Interfaces and Devices** → **Communication Devices** → Choose the diskhb created in the previous step (net_diskhb_01) -> Press Enter.

For Device Name, this is a unique name you can chose to describe the device. It will show up in your topology under this name, much like serial heartbeat and ttys have in the past.

For the *Device Path*, type in */dev/vpath0.* Then choose the corresponding node for this device.

```
                         Add a Communication Device

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
* Device Name                                          [jc_disk_hb]
* Network Type                                          diskhb
* Network Name                                          net_diskhb_01
* Device Path                                          [/dev/vpath0]
* Node Name                                            [Justen]           +
```

Figure 13-5 Adding individual diskhb communication devices

After creating the first device of any non-IP network, it is normal to get the warning message as shown in Example 13-1 on page 587

Example 13-1 Diskhb warning

```
WARNING: Serial network [net_name] has 1 communication device(s) configured.
Two devices are required for a serial network.
```

Once you repeat this process for the other node (Christie) and the other device (vpath3) the warning will no longer exist as the two device requirement is fulfilled.

13.5.5 Testing disk heartbeat connectivity

Once the device and network definitions have been created, it is recommended to test it to make sure communications are working properly. If the volume group is varied on in normal mode on any one of the nodes, the test will probably not succeed.

/usr/sbin/rsct/bin/dhb_read is used to test the validity of a diskhb connection. The usage of *dhb_read* is as follows:

Command	Action
dhb_read -p devicename	dumps diskhb sector contents
dhb_read -p devicename -r	receives data over diskhb network
dhb_read -p devicename -t	transmits data over diskhb network

To test the diskhb network connectivity we will set one node, Justen, to receive while the other node, Christie, will transmit.

On Justen we execute:

```
dhb_read -p rvpath0 -r
```

On Christie we execute:

```
dhb_read -p rvpath3 -t
```

Note: The devicename is the raw device as designated with the *"r"* proceeding the device name. For hdisks, the dhb_read utility automatically converts it to the proper raw device name. For all other devices, it is required to specify it explicitly.

If the link between the nodes are operational, both nodes will display "*Link operating normally"* as shown in Figure 13-6.

```
Justen /usr/sbin/rsct/bin >./dhb_read -p rvpath0 -r
Receive Mode:
Waiting for response . . .
Link operating normally
Justen /usr/sbin/rsct/bin >


Christie /usr/sbin/rsct/bin >./dhb_read -p rvpath0 -t
Transmit Mode:
Detected remote utility in receive mode.  Waiting for response . . .
Link operating normally
Christie /usr/sbin/rsct/bin >
```

Figure 13-6 Disk heartbeat communications test

In most cases the diskhb device is part off a shared data volume group. If the volume group is not currently a resource in a resource group, then add it to a resource group and synchronize the cluster.

13.5.6 Monitoring disk heartbeat

Once cluster services are running, you can monitor the activity of the disk (actually all) heartbeats via **`lssrc -ls topsvcs`**. The key field to monitor is the *Missed HBs.* If the total continues to grow, it is a good indication that the disk is not optimal for a diskhb network. Either move the diskhb to another disk, or change the failure detection rate of the diskhb network to slow.

An excerpt of the diskhb network information follows in Figure 13-7

```
Justen /usr/sbin/rsct/bin >lssrc -ls topsvcs

NIM's PID: 286930
diskhb_1       [ 3]    2    1  S 255.255.10.1    255.255.10.1
diskhb_1       [ 3] rvpath0      0x82b759bd      0x82b81f74
HB Interval = 2.000 secs. Sensitivity = 4 missed beats
Missed HBs: Total: 0 Current group: 0
Packets sent    : 25934 ICMP 0 Errors: 0 No mbuf: 0
Packets received: 25934 ICMP 0 Dropped: 0
NIM's PID: 282856
```

Figure 13-7 Monitoring diskhb

The default grace period before heartbeats start processing is 60 seconds. If executing this command quickly after starting the cluster, you will not see any disk heartbeat information until the grace period time has elapsed.

14

Networking

This chapter describes some new network options available within HACMP 5.3. Some of these new features include the service IP distribution policy and the auto creation of the clhosts file. This chapter also presents a discussion of the EtherChannel functionality and how to take advantage of it within an HACMP cluster. In addition, it explains the purpose of the netmon.cf, clhosts and clinfo.rc files. Recommendations are outlined about environments where these files should be implemented.

14.1 Etherchannel

EtherChannel (EC) is a port aggregation method whereby up to eight ethernet adapters are defined as one EtherChannel. Remote systems view the EtherChannel as one IP and MAC address so up to eight times network bandwidth is available in one network presence.

Traffic is distributed across the adapters in the standard way (address algorithm) or on a round robin basis. If an adapter fails, traffic is automatically sent to the next available adapter in the EtherChannel without disrupting user connections. When only one link in the main EtherChannel is active, a failure test triggers a rapid detection / failover (in 2-4 seconds) to optional backup adapter with no disruption to user connections. Two failure tests are offered – the physical adapter link to network and the optional TCP/IP path to the user-specified node. When failure is detected, the MAC and IP addresses are activated on the backup adapter. When at least one adapter in the main channel is restored, the addresses are reactivated on the main channel.

The AIX V5.1 Network Interface Backup (NIB) configuration mode was replaced and enhanced in AIX V5.2. The new method is a single adapter EtherChannel with backup adapter, providing a priority (failback upon link repair) between the primary and backup links which the previous implementation lacked. The dynamic adapter membership (DAM) enhancement in AIX V 5.2 allows the dynamic reconfiguration of adapters within the EtherChannel without disruption to the running connection.

Note: HACMP itself does not state support for DAM as this is below the level that HACMP monitors. So no support statement should be required.

All multi-adapter channels require special EtherChannel or IEEE 802.3ad port configuration in the network switch. In most cases, the switch will be configured for EtherChannel mode. However, if the switch doesn't support EC or if the corporation has standardized on IEEE 802.3ad, then configure 802.3ad at both the switch and in AIX. Single-adapter links, on the other hand, require no special configuration at the network switch. This includes a single-adapter EtherChannel and the backup adapter connection.

EtherChannel has the following benefits:

- Higher bandwidth and load balancing options
  - Multi-adapter channels utilize aggregate bandwidth
  - Several user configurable alternatives for directing traffic across the channel adapters

- Built in availability features
  - Automatically handles adapter, link and network failures
  - Optional backup adapter to avoid SPOF (single point of failure) at network switch
  - Design techniques to avoid SPOFs
- A simple, flexible solution and growth path
  - One Ethernet MAC and IP address for entire aggregation (including backup adapter)
  - Accommodates future bandwidth requirements easily
  - User can add, delete, and reconfigure adapters dynamically (no service disruption)
- Various options for interoperability with network switch
  - Multi-adapter channels for both EtherChannel and 802.3ad capable switches
  - Single adapter channels and backup adapter links are transparent to the network switch
  - Channel backup adapter option (connect to a different network switch to avoid SPOF)
  - Channel operates without switch when two systems cabled directly (back-to-back, though not applicable in HACMP environments)
- It's free (assuming EC capable switches are already in place)
  - included in AIX and regularly enhanced since AIX v4.3.3

14.1.1 Implementing EtherChannel in an HACMP environment

HACMP officially supports the use of EtherChannel. The statement of support can be found here:

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/FLASH10284

Integrating an EtherChannel solution into a cluster is relatively easy and can actually simplify network addressing and subnet requirements. In many cases, all addresses are configured on the same logical interface.

In our example, we will only show the relevant parts as it relates to the combination of HACMP and EtherChannel specifically. To avoid repetition in this book, basic HACMP configuration knowledge is assumed. This will not be step by step HACMP menu steps. Other best practices, like having non-IP heartbeat networks configured and using an EC capable switch as opposed to a crossover cables, are recommended. Basic HACMP configuration knowledge is assumed

The following details are based on a previous write up we did on this combination in May 2004, but is still valid today. The original document can be found at:

http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD101785

Test environment configuration

Our test environment was constructed using the following combination of components:

- Two pSeries p630 systems (named *neo* and *trinity*)
- AIX V5.2 ML3
- HACMP V5.1
- Ethernet network connections ent0 through ent6:
  - ent0 and ent5 (unused) are integrated 10/100 adapters
  - ent1, ent2, ent3, ent4 (unused) are all on a single 4-port 10/100 adapter
  - ent6 - EtherChannel (comprised of ent2, ent3 and ent0
  - Three UTP Ethernet crossover cables

Figure 14-1 is a diagram of the cluster configuration we used:

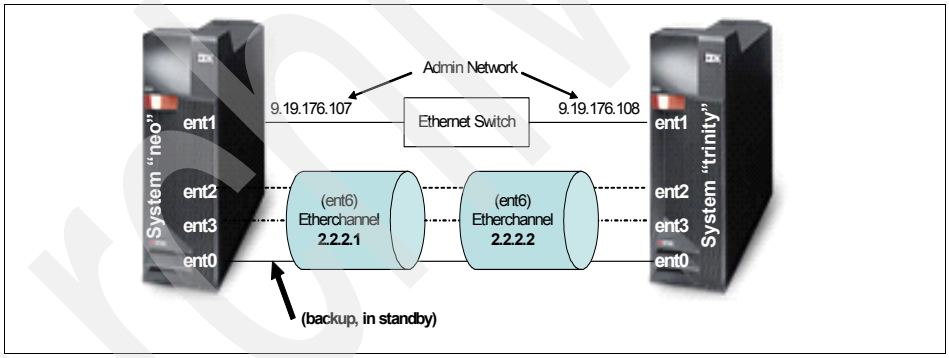

Figure 14-1 Etherchannel and HACMP test environment

In this test, we successfully implemented a "single adapter network" using HACMP IP Address Takeover (IPAT) with the EtherChannel function included in AIX V 5.2. The EtherChannel is responsible for providing local adapter swapping which is outside of HACMP. HACMP has no knowledge of EtherChannel and is completely independent. While a single adapter network is normally considered not ideal, EtherChannel makes this okay because there are multiple physical adapters within the single EtherChannel pseudo device. Thus, we can safely ignore the insufficient adapter warning messages posted during cluster synchronization.

Our configuration consisted of a rotating style resource group with a single adapter network using IP aliasing. Our testing proved to be beneficial in simplifying the HACMP setup. We implemented the EtherChannel connection without a network switch, by cabling the two test systems directly with crossover cables. This was only done for testing purposes. A typical HACMP environment would have these adapters cabled to an Etherchannel capable switch to fully exploit it.

Currently, switch manufacturers expect attachment of the individual links in the EtherChannel to be to the same network switch. For additional switch redundancy you can connect the backup adapter to a separate switch.

Choose the adapters for the EtherChannel carefully. The goal is to avoid a single point of failure. In the test environment, we had an integrated Ethernet adapter and a single 4-port Ethernet adapter on each system so we chose to configure the integrated adapter as the backup to eliminate our 4-port adapter as a single point of failure.

Configuration procedures

We set up our cluster via the following basic steps. details on each step, as completed for system *neo* follows.

1. Check the Ethernet adapter configurations and adapter cabling
2. Create EtherChannel interface
3. Configure IPs on new interface (en6) via TCP/IP
4. Add boot and service IPs to HACMP topology
5. Create a resource group and assign it the service IP
6. Synchronize cluster
7. Start cluster services
8. Test redundancy of NICs and make sure HACMP does not detect i

Start with unconfigured adapters, preferably cabled into an EC capable switch and the switch already set for an EC configuration. Our adapters had been configured previously so we removed the ODM interface definitions via **`smitty inet`**. We completed these basic steps on both systems, using the IP interfaces and IP addresses as shown in Figure 14-1 on page 594.

Step 1. Check Ethernet adapter configuration

The adapters that will become a part of the EtherChannel should be configured for the same speed and duplex mode. We configured ent0, ent2 and ent3 for 100 Mbps, full duplex via fastpath **`smitty eadap->Change / Show Characteristics of an Ethernet Adapter`** as shown in Figure 14-2 on page 596.

Note: If using Gigabit adapters, the media speed on most are set to auto-detect and are a non-tunable setting.

```
           Change / Show Characteristics of an Ethernet Adapter

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  Ethernet Adapter                                    ent2
  Description                                         IBM 4-Port 10/100 Bas>
  Status                                              Available
  Location                                            12-08
  TRANSMIT queue size                                 [8192]
  HARDWARE RECEIVE queue size                         [256]
  RECEIVE buffer pool size                            [384]
  Media Speed                                          100_Full_Duplex
  Inter-Packet Gap                                    [96]
  Enable ALTERNATE ETHERNET address                    no
  ALTERNATE ETHERNET address                          [0x000000000000]
  Enable Link Polling                                  no
  Time interval for Link Polling                      [500]
  Apply change to DATABASE only                        no

F1=Help           F2=Refresh         F3=Cancel          F4=List
Esc+5=Reset       F6=Command         F7=Edit            F8=Image
F9=Shell          F10=Exit           Enter=Do
```

Figure 14-2 Ethernet adapter settings

Tip: At this point, its a good idea to test these links by configuring IP addresses on each side. Just remember to remove the configuration prior to the next step.

Step 2. Configure Etherchannel

Configure the EtherChannel through the fastpath **`smitty etherchannel->Add an Etherchannel/Link Aggregation`** and select the appropriate adapters via F7. In our configuration, ent2 and ent3 comprise the main channel and ent0 is the backup adapter. Processing the following menu, as pictured in Figure 14-3 on page 597, creates the new EtherChannel interface (ent6).

```
                    Add An EtherChannel / Link Aggregation

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  EtherChannel / Link Aggregation Adapters            ent2,ent3           +
  Enable Alternate Address                            no                  +
  Alternate Address                                   []                  +
  Enable Gigabit Ethernet Jumbo Frames                no                  +
  Mode                                                round_robin +
  Hash Mode                                           default             +
  Backup Adapter                                      ent0                +
       Automatically Recover to Main Channel          yes                 +
  Internet Address to Ping                            []
  Number of Retries                                   []                  +
  Retry Timeout (sec)                                 []                  +



F1=Help             F2=Refresh          F3=Cancel           F4=List
Esc+5=Reset         F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 14-3 Add Etherchannel Menu

We selected round robin mode so both links will be utilized in this configuration. Please refer to the EtherChannel documentation to learn about the different modes and select the one that will best suit your configuration.

Important: When implementing a single interface with a single backup adapter (previously network interface backup or NIB) and specifying the values of “Number of Retries” and “Retry Timeout” make sure that they do not exceed HACMP’s NIM settings for failure detection rate. It is recommended to have these settings be at least half of the HACMP settings.

Step 3. Configure IP on Etherchannel device

Configure the IP interface (en6) on the EtherChannel using fastpath **`smitty chinet->choose en6`** as shown in Figure 14-4 on page 598.

```
                Change / Show a Standard Ethernet Interface

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                  [Entry Fields]
  Network Interface Name                         en6
 INTERNET ADDRESS (dotted decimal)               [2.2.2.1]
  Network MASK (hexadecimal or dotted decimal)   [255.255.255.0]
  Current STATE                                   up                 +
  Use Address Resolution Protocol (ARP)?          yes                +
  BROADCAST ADDRESS (dotted decimal)             []
  Interface Specific Network Options
     ('NULL' will unset the option)
    rfc1323                                      []
    tcp_mssdflt                                  []
    tcp_nodelay                                  []
    tcp_recvspace                                []
    tcp_sendspace                                []
  Apply change to DATABASE only                   no                 +

F1=Help           F2=Refresh        F3=Cancel          F4=List
Esc+5=Reset       F6=Command        F7=Edit            F8=Image
F9=Shell          F10=Exit           Enter=Do
```

Figure 14-4 Configure IP to EtherChannel device

We repeated this procedure on node *trinity* using an IP address of 2.2.2.2.

Note: Remember, when running familiar TCP/IP commands to run them against the new psuedo interface (en6) and not the individual interfaces.

Step 4. Configure HACMP Topology

In our testing we chose to use IP aliasing when defining our HACMP network (channet) . We configured our boot IP addresses on each EtherChannel device (neo_boot 2.2.2.1, trinity_boot 2.2.2.2). We then defined our service IP address (bound to multiple nodes) 192.168.43.4 and our persistent IP addresses, 192.168.43.10 on neo and 192.168.43.20 on trinity. Our topology configuration can be seen in Figure 14-5 on page 599 in the ouput of the **`cllsif`** command.

```
Adapter Type Network Net Type Attribute Node IP Addr Hardware Addr Interface Name Global Name Netmask
neo_boot1     boot    channet ether public neo     2.2.2.1       en6   255.255.255.0
neoec_srv     service channet ether public neo     192.168.43.4        255.255.255.0
trinity_boot1 boot    channet ether public trinity 2.2.2.2       en6   255.255.255.0
neoec_srv     service channet ether public trinity 192.168.43.4        255.255.255.0
```

Figure 14-5 EtherChannel configuration cllsif output

Note: Although omitted from our example, at least one non-IP serial network should always be used in a production environment.

Step 5. Configure HACMP Resource Group

We configured one cascading resource group with the single service IP label defined to it. Since our focus was on the NIC redundancy testing, we simplified the configuration by omitting additional resource in the resource group. Our resource group definition can be seen in Figure 14-6.

```
                Change/Show Resources for a Rotating Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Resource Group Name                                   testec_rg
  Participating Node Names (Default Node Priority)      neo trinity
* Service IP Labels/Addresses                          [neoec_svc]        +
  Volume Groups                                         []                 +
  Filesystems (empty is ALL for VGs specified)          []                 +
  Application Servers                                   []                 +
```

Figure 14-6 Etherchannel Resource Group

Step 6. Synchronize the cluster

Though HACMP familiarity is assumed we wanted to show the warning message displayed when HACMP topology is configured as a single adapter network. When synchronizing the cluster we got the following the warning (see Figure 14-7 on page 600):

```
WARNING: There may be an insufficient number of communication interfaces
defined on node: neo, network: channet. Multiple communication interfaces
are recommended for networks that will use IP aliasing.

WARNING: There may be an insufficient number of communication interfaces
defined on node: trinity, network: channet. Multiple communication
interfaces are recommended for networks that will use IP aliasing.
```

Figure 14-7 Single adapter network warning

Since we truly have configured only one interface to HACMP topology, this warning message is expected. EtherChannel provides the local adapter redundancy and swapping as needed.

Important: When implementing a similar single adapter network configuration, for proper HACMP network detection configuring a netmon.cf file is needed. More information can be found in “Understanding the netmon.cf file” on page 605

Step 7. Start cluster services

Execute **`smitty clstart`** on each node and wait for node_up _complete.

Step 8. Testing

Our testing focused on physically pulling cables to see how the system responded and to make sure HACMP was unaware. While performing each test, we ran a ping from an outside client node, to both boot IPs and the service IP.

1. We pulled the cable from ent3. This resulted in continued service surviving on ent2. This was verified with netstat and entstat commands, along with the surviving ping running from the client. AIX makes note of this in the error report. HACMP however, is unaware that a failure occurred. The error report errors are shown in Figure 14-8.

```
F77ECAC2 0624145904 T H ent3 ETHERNET NETWORK RECOVERY MODE
8650BE3F 0624145904 I H ent6 ETHERCHANNEL RECOVERY
F77ECAC2 0624145904 T H ent2 ETHERNET NETWORK RECOVERY MODE
```

Figure 14-8 Etherchannel errors in AIX error report

2. We pulled the cable from ent2. This caused the standby adapter of ent0 to takeover the services. Much like the previous tests, AIX noted failure in the error report, but not HACMP. Since we used crossover cables, this had a dual effect of causing similar errors and swaps on both nodes.

3. We then pulled the lone surviving adapter of ent0. This, in turn, resulted in a full EtherChannel failure, which was noticed as a failed network by HACMP.

EtherChannel conclusions

Our overall thoughts about the implementation of EtherChannels in an HACMP environment were very positive. Although the configuration will require some additional initial planning, it was very quick and easy to setup. We were especially pleased with the recovery times of our testing; they were almost instantaneous and had no impact on our running cluster. We were also pleased at how the implementation of this model eliminates the removal of routes in HACMP events associated with local adapter swaps, making the failure time shorter and easier to troubleshoot.

In summary, the simplicity and overall benefits of the EtherChannel model make it a very promising choice when planning a new environment that needs HACMP's availability with scalable network bandwidth and redundancy. The dynamic scalability and possibilities for even greater redundancy are an even bigger incentive to consider migration to this type of configuration.

14.2 Distribution preference for service IP aliases

When using IP aliasing with multiple service IP addresses configured, HACMP will analyze the total number of aliases, whether defined to HACMP or not, and assign each service address to the least loaded interface. HACMP 5.1 and above, give you added control over their placement and allow you to define a distribution preference for your service IP label aliases.

This network-wide attribute can be used to customize the load balancing of HA service IP labels taking into consideration any persistent IP labels already configured. The distribution selected is maintained during cluster startup and subsequent cluster events. The distribution preference will be maintained as long as acceptable network interfaces are available in the cluster. However, HACMP will always keep service IP labels active, even if the preference cannot be satisfied.

There are four different distribution policies available:

- **Anti-collocation**: This is the default. HACMP distributes all service IP aliases across all base IP addresses using a "least loaded" selection process.
- **Collocation**: HACMP allocates all service IP aliases on the same network interface card (NIC).
- **Anti-collocation with persistent**: HACMP distributes all service IP aliases across all active physical interfaces that are NOT hosting the persistent IP

label. HACMP will place the service IP 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 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 addresses, HACMP lets you select the Collocation with Persistent distribution preference, but it issues a warning and uses the regular collocation preference by default.

14.2.1 Configuring service IP distribution policy

The distribution preference may be set or changed dynamically. The steps to configure this type of distribution policy are:

1. Enter `smit hacmp`
2. In SMIT, select `Extended Configuration > Extended Resource Configuration > HACMP Extended Resources Configuration > Configure Resource Distribution Preferences > Configure Service IP labels/addresses Distribution Preferences` and press Enter.

   HACMP will display only networks using IPAT via Aliasing.
3. Select the network for which you want to specify the policy and press Enter.
4. From the `Configure Service IP Labels/Address Distribution Preference` screen choose the Distribution Preference desired.
5. Press Enter to accept your selection and update the HACMP ODM on the local node.
6. In order for the change to take effect and to get propagated out to all nodes you will need to synchronize your cluster. Go to the `Initialization and Standard Configuration` or `Extended Configuration` menu and select `Verification and Synchronization`. This will trigger a dynamic reconfiguration event.

Note: Configuring the service IP distribution policy resulted in the following messages:

```
cldare: Detected changes to service IP label app1svc. Please note that
changing parameters of service IP label via a DARE may result in releasing
resource group <name>
```

Viewing the distribution preference for service IP label aliases

You are supposed to be able to display the current distribution preference for each network using the **cltopinfo** or the **cllsnw** commands.

The output of **cltopinfo -w** will display the following:

```
# /usr/es/sbin/cluster/utilities/cltopinfo -w
NODE cobra:
        Network net_diskhb_01
                cobra_vpath0    /dev/vpath0
        Network net_ether_01
                app1svc 192.168.100.83
                app2svc 192.168.100.82
                cobraa  10.10.31.33
                cobrab  10.10.32.33
NODE viper:
        Network net_diskhb_01
                viper_vpath0    /dev/vpath0
        Network net_ether_01
                app1svc 192.168.100.83
                app2svc 192.168.100.82
                viperb  10.10.32.32
                vipera  10.10.31.32
Network net_ether_01 is using the following distribution preference for service
labels:
Collocation with persistent - service label(s) will be mapped to the same
interface as the persistent label
```

The following is the sample output of **cllsnw -c** displaying the service label distribution preference (sldp) for a particular network:

```
#/usr/es/sbin/cluster/utilities/cllsnw -c
#netname:attr:alias:monitor_method:sldp:
net_ether_01:public:true:default:ppstest::sldp_collocation_with_persistent
```

Lab experiences with service distribution policy

In our testing we were able to change the service IP distribution policy with cluster services down on all of the nodes and on cluster startup see the IP labels and persistent IPs get distributed according the specified policy. This was visible in the output of netstat -i:

```
python-# more /etc/hosts
10.10.31.31   pythona      # base address 1
10.10.32.31   pythonb      # base address 2
192.168.100.31  p630n01 n1 # python persistent address
192.168.100.82  app1svc    # cobra service address
192.168.100.83  app2svc    # viper service address

python-# netstat -i
```

```
Name  Mtu   Network     Address            Ipkts Ierrs    Opkts Oerrs  Coll
en0   1500  link#2      0.2.55.4f.c4.ab    5044669     0  1828909     0     0
en0   1500  10.10.31    pythona            5044669     0  1828909     0     0
en0   1500  192.168.100 p630n01            5044669     0  1828909     0     0
en0   1500  192.168.100 app1svc            5044669     0  1828909     0     0
en0   1500  192.168.100 app2svc            5044669     0  1828909     0     0
en3   1500  link#3      0.20.35.e2.7f.8d   3191047     0  1410806     0     0
en3   1500  10.10.32    pythonb            3191047     0  1410806     0     0
lo0   16896 link#1                         1952676     0  1957548     0     0
lo0   16896 127         localhost          1952676     0  1957548     0     0
lo0   16896 localhost                      1952676     0  1957548     0     0
```

Note: In the output above node **python** had the resource groups for nodes **cobra** and **viper** and their corresponding service IPs. The distribution policy was set to `Collocation with persistent.`

Our testing of the dynamic change of this policy resulted in no move of any of the labels after a synchronization. The following message was logged during the synchronization of the cluster after making the service IP distribution policy change:

```
Verifying additional pre-requisites for Dynamic Reconfiguration...

cldare: Detected changes to service IP label app1svc.  Please note
that changing parameters of service IP label via a DARE may result in
releasing resource group APP1_RG .

cldare: Detected changes to service IP label app2svc.  Please note
that changing parameters of service IP label via a DARE may result in
releasing resource group APP2_RG .
```

Note: For this instance the message logged is generic and only gets reported because a change was detected. As long as that was the only change made no actual resources will be brought offline.

A change to the service IP distribution policy is only enforced whenever we manually invoke a swap event or stop and restart HACMP on a node. Note that this is the intended behavior of the feature in order to avoid any potential disruption of connectivity to those IP addresses. The remaining cluster nodes will not enforce the policy unless cluster services are also stopped and restarted on them.

14.3 Understanding the netmon.cf file

In HACMP you can create a netmon.cf configuration file with a list of additional network addresses. These addresses will only be used by topology services to send ICMP ECHO requests in order to help determine an adapter's status under certain circumstances.

The implementation of this file is therefore not required, but recommended in cluster configurations with only a single network card on each node or in a cluster where failures have left a single adapter network remaining. In these scenarios it can be difficult for HACMP to accurately determine an adapter failure since topology services cannot force traffic over the single adapter to confirm its proper operation.

An enhancement to netmon, the network portion of RSCT topology services, allows for a more accurate determination of a service adapter failure. This function can be used in a configuration that requires the use of a single service adapter per network.

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

Creating a netmon.cf file

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

Requirements for creating the file:

- The file must consist of one IP address or IP label per cable.
- The file should contain remote IP labels/addresses that are not in the cluster configuration and that can be accessed from HACMP interfaces. We recommend the use of the router's IP address.
- A maximum of 30 IP addresses/labels can be defined in netmon.cf
- Include each IP address and its corresponding label in the `/etc/hosts` file.

> **Note:** When the NIM process (from RSCT Topology Services) attempts to determine the state of local adapters it may try to use hostname resolution. If the IP and corresponding label are not in `/etc/hosts` and a problem or a delay is encountered while trying to resolve the address, the overall failure detection time may be prolonged and result in slow fallover operations.

The contents of your netmon.cf file may resemble the following:

```
/usr/es/sbin/cluster/netmon.cf
192.168.100.76
p690_1_lpar3
192.168.100.35
router_lan1

/etc/hosts  (corresponding entries)
192.168.100.76   node365       #client node running oracle
192.168.100.21   p690_1_lpar3  #client node hosting application 4
192.168.100.35   node367       #client node running db2
192.168.100.200  router_lan1   #router hosting production lan
```

Recommendations and additional notes

As a general rule of thumb, you should implement a netmon.cf file in a two node cluster configuration using a single IP network, regardless of the implementation of a non-IP, serial heartbeat network. This should be done in order to help topology services identify an adapter failure if it ever goes into a singleton state, where basically only node is left in the cluster.

Other scenarios can include environments using a single logical EtherChannel interface made up of multiple ethernet links. In that environment link failures are handled seamlessly by the EtherChannel logic, but a complete channel failure would result in problems for topology services without the netmon.cf file. Implementing EtherChannel in an HACMP environment is discussed in detail in section “Etherchannel” on page 592.

14.4 Understanding the clhosts file

The clhosts file contains IP address information which helps to enable communication among monitoring daemons on clients and within the HACMP cluster nodes. The tools that utilize this file include: clinfoES, HAView, and clstat. The file resides on all HACMP cluster servers and clients in the `/usr/es/sbin/cluster/etc/` directory.

When a monitor daemon starts up, it reads the `/usr/es/sbin/cluster/etc/clhosts` file to determine which nodes are available for communication. Therefore, it is important for these files to be in place whenever trying to use the monitoring tools from a client outside of the cluster. Whenever the server portion of HACMP is installed the clhosts file is updated on the cluster nodes with the loopback address (127.0.0.1). The contents of the file within each cluster node typically will only contain the following line:

```
127.0.0.1     # HACMP/ES for AIX
```

Creating the clhosts file

In prior releases, you were required to manually create and maintain the client-based clhosts file. In HACMP 5.3, you can automatically generate the clhosts file needed by clients when you perform a verification with the automatic corrective action feature enabled. The verification will create a `/usr/es/sbin/cluster/etc/clhosts.client` file on all cluster nodes.

The file will look similar to the following example:

```
# /usr/es/sbin/cluster/etc/clhosts.client Created by HACMP Verification /
Synchronization Corrective Actions
# Date Created: 07/01/2005 at 12:45:29
192.168.100.102 #dlpar_app2_svc
192.168.100.101 #dlpar_app1_svc
192.168.202.204 #alexis_base1
192.168.202.205 #alexis_base2
192.168.100.61  #alexis
192.168.201.203 #jessica_base2
192.168.201.202 #jessica_base1
192.168.100.72  #jessica
192.168.200.200 #jordan_base1
192.168.200.201 #jordan_base2
192.168.100.71  #jordan
```

Notice that all of the addresses are pulled in including the boot, service, and persistent IP labels. Before utilizing any of the monitor utilities from a client node the `clhosts.client` file must be copied over to all clients as /usr/es/sbin/cluster/etc/clhosts. Remember to remove the .client extension when you copy the file over to the client nodes.

Important: The clhosts file on a client should never contain 127.0.0.1, loopback, or localhost.

Clstat on a client and the clhosts file

When running the clstat utility from a client, the clinfoES daemon will obtain its cluster status information from the server side SNMP and populates the HACMP MIB on the client side. It will be unable to communicate with the deamon and report that it is unable to find any clusters if it has no available clhosts file.

In this type of environment it is critical to implement a clhosts file on the client. This file will give the clinfoES daemon the addresses to attempt communication with the SNMP process running on the HACMP cluster nodes.

Restriction: When using IPAT via Replacement do not include standby addresses in the clhosts file.

HAView and the clhosts file

HAView monitors a cluster's state within a network topology based on cluster specific information in the `/usr/es/sbin/cluster/etc/clhosts` file. It must be present on the Tivoli NetView management node. Make sure that the hostname and service label of your Tivoli NetView nodes are exactly the same. (If they are not the same, add an alias in the `/etc/hosts` file to resolve the name difference.)

If an invalid IP address exists in the clhosts file, HAView will fail to monitor the cluster. Make sure that the IP addresses are valid, and there are no extraneous characters in the file.

14.5 Understanding the clinfo.rc file

HACMP may be configured to change the MAC address of a network interface by the implementation of hardware address takeover (HWAT). In a switched ethernet network environment, the switch may not always get promptly informed of the new MAC. In turn, the switch will not route the appropriate packets to the network interface.

The clinfo.rc script is used by HACMP to flush the system's ARP cache in order to reflect changes to network IP addresses. It does not update the cache until another address responds to a ping request. Flushing the ARP cache typically is not necessary if the HACMP hardware address swapping facility is enabled. This is because hardware address swapping maintains the relationship between an IP address and a hardware address.

Note: HWAT is only supported in HACMP when using IPAT via replacement.

On clients not running clinfoES, you may have to update the local ARP cache indirectly by pinging the client from the cluster node. In order to avoid this, add the name or address of a client host you want to notify to the PING_CLIENT_LIST variable in the clinfo.rc script. The clinfoES program will calls the `/usr/es/sbin/cluster/etc/clinfo.rc` script whenever a network or node event occurs. Through use of PING_CLIENT_LIST entries in clinfo.rc can update the ARP caches for clients and network devices such as routers.

When a cluster event occurs, clinfo.rc runs the following command for each host specified in PING_CLIENT_LIST:

```
ping -c1 $host
```

Note: This assumes the client is connected directly to one of the cluster networks.

Configuring the clinfo.rc file

Do the following to ensure that the new MAC address is communicated to the switch:

1. Modify the line in `/usr/es/sbin/cluster/etc/clinfo.rc` that currently reads:

```
PING_CLIENT_LIST=" "
```

2. Include on this line the names or IP addresses of at least one client on each subnet on the switched Ethernet.

3. Run clinfoES on all nodes in the HACMP cluster that are attached to the switched Ethernet.

Remember to do the following:

- If you normally start HACMP cluster services using the /usr/es/sbin/cluster/etc/rc.cluster shell script, specify the -i option.
- If you normally start HACMP cluster services through SMIT, specify yes in the Start Cluster Information Daemon? field.
- A copy of the /usr/es/sbin/cluster/etc/clinfo.rc script must exist on each server node and client in the cluster in order for all ARP caches to be updated.

How clinfo and clinfo.rc work

The format of the clinfo call to clinfo.rc:

```
clinfo.rc {join,fail,swap} interface_name
```

Clinfo obtains information about the interfaces and their current state, and checks for changed states of interfaces:

- If a new state is UP, Clinfo calls clinfo.rc join interface_name.
- If a new state is DOWN, Clinfo calls clinfo.rc fail interface_name.
- If Clinfo receives a node_down_complete event, it calls clinfo.rc with the fail parameter for each interface currently UP.
- If Clinfo receives a fail_network_complete event, it calls clinfo.rc with the fail parameter for all associated interfaces.
- If Clinfo receives a swap_complete event, it calls clinfo.rc swap interface_name.

Part 5

Disaster recovery

Part 5 presents topics about HACMP Extended Distance (HACMP/XD). The subjects covered in this part include:

- HACMP Extended distance concepts and planning
- HACMP with cross-site LVM
- HAGEO disaster recovery scenario
- GLVM concepts and configuration

15

HACMP Extended distance concepts and planning

This chapter discusses the HACMP Extended Distance (HACMP/XD) features and capabilities, and describes how to install and configure some of the disaster recovery features of HACMP/XD.

We discuss:

- HACMP/XD components
- Disaster recovery considerations
- More information

15.1 HACMP/XD components

The High Availability Cluster Multi-Processing for AIX (HACMP) base software product addresses part of the continuos operation problem. It addresses recovery from the failure of a node, an adapter, or a local area network within a computing complex at a single site.

HACMP/XD extends the base capability of the HACMP, by providing automated fallover/fallback support for applications over geographically dispersed systems. Systems running in different locations are defined as HACMP nodes assigned to sites and they are managed by HACMP like usual nodes.

The key function of HACMP/XD is data replication across sites. To accomplish this function HACMP/XD can use several components:

- HAGEO
- PPRC
- GLVM

15.1.1 HACMP/XD HAGEO

The HAGEO/GeoRM software is the original solution for data replication over TCP/IP networks. Initially, it was built as a separate product which could be used as a standalone version (GeoRM), only for data replication or as an integrated version, working together with HACMP to provide automated site fallover/fallback functions for the applications using the replicated data. Beginning with HACMP 5.1, HAGEO is part of HACMP/XD software.

A typical HACMP/XD High Availability Geographic Cluster HAGEO) is presented in Figure 15-1 on page 615.

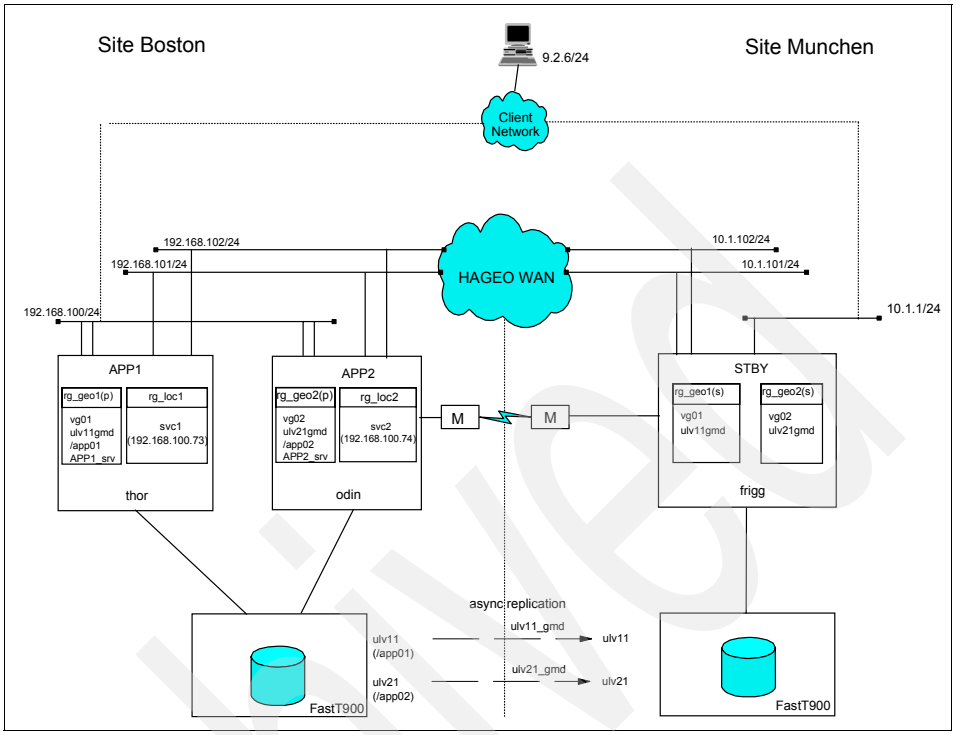

Figure 15-1 Example of HACMP/XD HAGEO configuration

HAGEO provides:

- Ability to configure a cluster with geographically separate sites.

  HAGEO extends HACMP to encompass two geographically distant data centers or sites. This extension prevents an individual site from being a single point of failure within the cluster.

  The geo-mirroring process supplies each site with an updated copy of essential data.

  Either site can run key applications, ensuring that mission-critical computing resources remain continuously available at a geographically separate site if a failure or disaster disables one site.

- Automatic failure detection and notification.

  HAGEO works with HACMP to provide automatic detection of a site or geographic network failure. It initiates the recovery process and notifies the system administrator about all failures it detects and actions it takes in response.

- Automatic fallover

HAGEO includes event scripts to handle recovery from a site or geographic network failure. These scripts are integrated with the standard HACMP event scripts.

You can customize the behavior for your configuration by adding pre- or post-event scripts, just as you can for HACMP.

- Fast recovery from a disaster.

  HAGEO also provides fast recovery of data and applications at the operable site. The geo-mirroring process ensures that the data is already available at the second site when a disaster strikes.

  Recovery time typically takes minutes, not including the application recovery time.

- Automatic resynchronization of data during site recovery.

  HAGEO handles the resynchronization of the mirrors on each site as an integral part of the site recovery process. The nodes at the rejoining site are automatically updated with the data received while the site was in failure.

- Reliable data integrity and consistency.

  HAGEO's geographic mirroring and geographic messaging components ensure that if a site fails, the surviving site's data is consistent with the failed site's data.

  When the failed site reintegrates into the cluster, HAGEO updates that site with the current data from the operable site, once again ensuring data consistency.

- Flexible, scalable configurations.

  HAGEO software supports a wide range of configurations, allowing you to configure the disaster recovery solution unique to your needs.

  You can have up to eight nodes in an HAGEO cluster, with varying numbers of nodes at each site.

  HAGEO is file system and database independent, since the geo-mirroring device behaves the same as the disk devices it supports. Because the mirroring is transparent, applications configured to use geo-mirroring do not have to be modified in any way.

HAGEO components

The software has three significant functions:

- GeoMirror:

  Consists of a logical device and a pseudo device driver that mirrors at a second site; the data is entered at one site. TCP/IP is used as a transport for mirrored data. GeoMirror can be used in synchronous and asynchronous

mode, depending on the communication bandwidth between sites, and the application transaction volume (which determines the amount of changed data).

- GeoMessage:

  Provides reliable delivery of data and messages between GeoMirror devices at the two sites. GeoMessage is a kernel-to-kernel process messaging system that the GeoMirror device driver uses to send and receive messages over IP-based networks. GeoMessage can use UDP or TCP as transport protocol over the IP network.

- Geographic topology:

  Provides the logic for integrating the geo-mirroring facilities with HACMP facilities to provide automatic failure detection and recovery from events that affect entire sites. This component includes:

  – Scripts and programs that integrate handling GeoMirror and GeoMessage in cluster events such as node and network joins and failures.

  – Scripts that integrate the starting and stopping of the GeoMirror and GeoMessage functions into the HACMP start and stop scripts.

  – Error-log messages to ensure GeoMirror and GeoMessage activities are logged.

HACMP/XD HAGEO basic configurations

You can configure an HAGEO cluster in any of the configurations supported by the HACMP base software. These include standby, one-sided takeover, mutual takeover, and concurrent access configurations.

- Standby configurations

  The standby configuration is a traditional redundant hardware configuration where one or more nodes in the cluster stand idle until a server node fails.

  In HAGEO, this translates to having an idle site. A site is not completely idle since it may also be involved in the geo-mirroring process. But nodes at this site do not perform application work.

- Takeover configurations

  In a takeover configuration, all nodes are processing; no idle nodes exist. Configurations include:

  – Intrasite (local) takeover

  – Remote one-sided takeover

  – Remote mutual takeover

- Concurrent configurations

In a concurrent access configuration, all nodes at one site have simultaneous access to the concurrent volume group and own the same disk resources. The other site is set up the same way.

If a node leaves the site, availability of the resources is not affected, since other nodes have the concurrent volume group varied on.

If a site fails, the other site offers concurrent access on nodes at that site. A concurrent application can be accessed by all nodes in the cluster.

The HACMP Cluster Lock Manager must be running on all nodes in the cluster. Not all databases can be used for concurrent access that involves nodes across the geography.

15.1.2 HACMP/XD PPRC integration feature

This feature was introduced simultaneously in HACMP V4.5 PTF5 and HACMP V5.1, and provides automated site fallover and activation of remote copies of application data in an environment where the IBM Enterprise Storage Server (ESS) is used in both sites and the Peer to Peer Remote Copy (PPRC) facility provides storage volumes mirroring.

A typical configuration for HACMP/XD PPRC is shown in Figure 15-2.

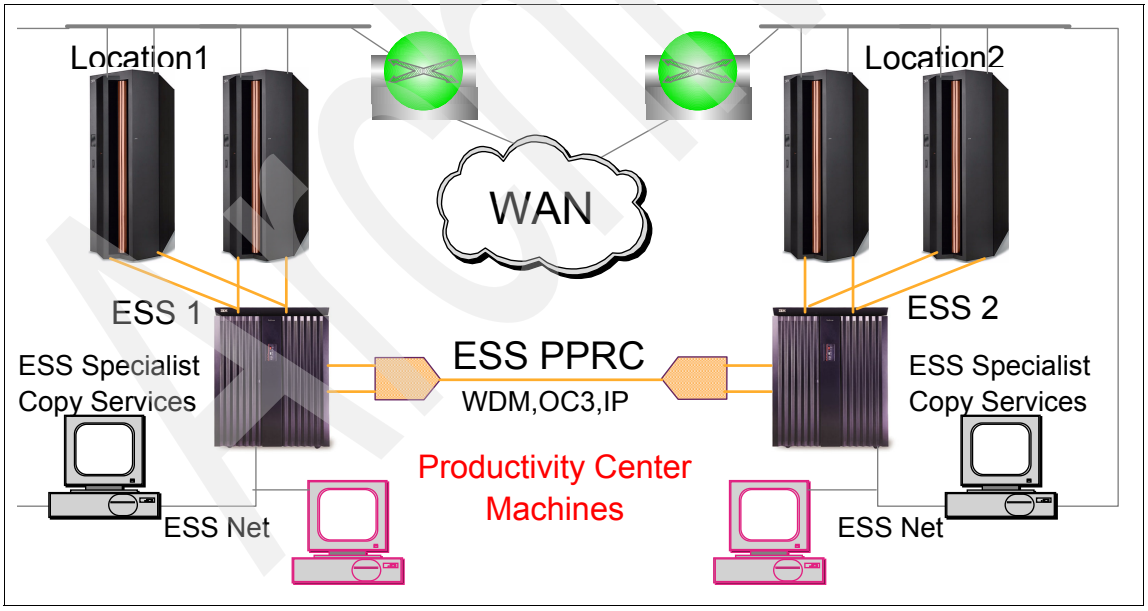

Figure 15-2 Example of an HACMP/XD PPRC configuration

In case of primary site failure, data should be available for use at the secondary site (replicated via PPRC). The data copy in the secondary site must be activated in order to be used for processing.

The HACMP/XD PPRC integration feature provides automated copy split in case of primary site failure and automated reintegration when the primary site becomes available.

For detailed information, see *HACMP/XD for ESS PPRC Version 5.3: Planning and Administration Guide*, SC23-4863.

15.2 Disaster recovery considerations

Disaster recovery strategies cover a wide range from no recovery readiness to automatic recovery with high data integrity. Data recovery strategies must address the following issues:

- Data readiness levels.
  - Level 0: None. No provision for disaster recovery.
  - Level 1: Periodic backup. Data required for recovery up to a given date is backed up and sent to another location.
  - Level 2: Ready to roll forward. In addition to periodic backups, data update logs are also sent to another location. Transport can be manual or electronic. Recovery is to the last log data set stored at the recovery site.
  - Level 3: Roll forward or forward recover. A shadow copy of the data is maintained on disks at the recovery site. Data update logs are received and periodically applied to the shadow copy using recovery utilities.
  - Level 4: Real time roll forward. Like roll forward, except updates are transmitted and applied at the same time as they are being logged in the original site. This real-time transmission and application of log data does not impact transaction response time at the original site.
  - Level 5: Real time remote update. Both the original and the recovery copies of data are updated before sending the transaction response or completing a task.
- Site interconnection options.
  - Level 0: None. There is no interconnection or transport of data between sites.
  - Level 1: Manual transport. There is no interconnection. For transport of data between sites, dispatch, tracking, and receipt of data is managed manually.

  - Level 2: Remote tape. Data is transported electronically to a remote tape. Dispatch and receipt are automatic. Tracking can be either automatic or manual.
  - Level 3: Remote disk. Data is transported electronically to a remote disk. Dispatch, receipt, and tracking are all automatic.

- Recovery site readiness.
  - Cold: A cold site typically is an environment with the proper infrastructure, but little or no data processing equipment. This equipment must be installed as the first step in the data recovery process.

    Both periodic backup and ready to roll forward data can be shipped from a storage location to this site when a disaster occurs.
  - Warm: A warm site has data processing equipment installed and operational. This equipment is used for other data processing tasks until a disaster occurs. Data processing resources can be used to store data, such as logs. Recovery begins after the regular work of the site is shut down and backed up.

    Both periodic backup and ready to roll forward data can be stored at this site to expedite disaster recovery.
  - Hot: A hot site has data processing equipment installed and operational and data can be restored either continually or regularly to reduce recovery time.

    All levels from roll forward to real-time remote update can be implemented.

HACMP/XD software provides the highest level of disaster recovery:

- Data readiness level 5: HACMP/XD provides real-time remote update data readiness by updating both the original and the recovery copies of data prior to sending a transaction response or completing a task.
- Site interconnection level 3: HACMP/XD also provides remote disk site inter connectivity by transmitting data electronically to a geographically distant site where the disks are updated and all bookkeeping is automatic.
- Hot site readiness. Since recovery site contains operational data processing equipment along with current data, this keeps recovery time to a minimum.

In case of using HACMP PPRC feature, data mirroring is managed by the PPRC function at the storage level.

With HACMP/XD, the recovery site can be actively processing data and performing useful work. In fact, each site can be a backup for the other, thereby minimizing the cost of setting up a recovery site for each original production site.

15.3 More information

Refer to these references for more information:

- *High Availability Cluster Multi-Processing XD (Extended Distance) for HAGEO Technology: Concepts and Facilities*, SA22-7955
- *HACMP/XD for ESS PPRC Version 5.3: Planning and Administration Guide*, SC23-4863
- *HACMP/XD for Geographic LVM: Administration and Planning Guide*, SA23-1338

16

HACMP with cross-site LVM

This chapter describes a disaster recovery solution, based on AIX and a basic HACMP cluster. It is built from the same components generally used for local cluster solutions with SAN-attached storage.

Cross-site LVM mirroring replicates data between the disk subsystems at each site.

16.1 Cross-site LVM mirroring introduction

A storage area network (SAN) is a high-speed network that allows the establishment of direct connections between storage devices and servers. The maximum distance for the connections is defined by Fibre Channel limitations. This allows for two or more servers, located in different sites to access the same physical disks.

These remote disks can be combined into a volume group via the AIX 5L Logical Volume Manager (LVM) and this volume group can be imported to the nodes located at different sites. You can create logical volumes and set up a LVM mirror with a copy at each site. The number of active sites in a cross-site LVM mirroring supported in HACMP is limited to two.

The main difference between general local clusters and cluster solutions with cross-site mirroring is as follows:

Within local clusters, all nodes and storage subsystems are located in the same location. With cross-site mirrored cluster nodes and storage subsystems reside on different site locations. Each site has at least one cluster node and one storage subsystem with all necessary IP network and SAN infrastructure.

This solution offers automation of AIX LVM mirroring within SAN disk subsystems between different sites. It also provides automatic LVM mirroring synchronization and disk device activation when, after a disk or site failure, a node or disk becomes available.

Each node in a cross-site LVM cluster accesses all storage subsystems. The data availability is ensured through the LVM mirroring between the volumes residing on different storage subsystems on different sites.

Figure 16-1 on page 627 explains the two-site LVM cross-mirroring environment that we used for our cross-site LVM mirroring test.

In case of site failure, HACMP performed a takeover of the resources to the secondary site according to the configured cluster policy. It activates all defined volume groups from the surviving mirrored copy. In case one storage subsystem fails, data access is not interrupted and applications can access data from the active mirroring copy on surviving disk subsystem.

HACMP drives automatic LVM mirroring synchronization, and after the failed site joins the cluster, it automatically fixes removed and missing volumes (PV states *removed* and *missing*) and synchronizes data. 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.

16.1.1 Requirements

The following requirements are necessary to assure data integrity and appropriate HACMP reaction in case of site or disk subsystem failure:

- The force varyon attribute for the resource group must be set to true.
- The logical volumes allocation policy must be set to superstrict (this ensures that LV copies are allocated on different volumes, and the primary and secondary copy of each LP is allocated on disks located in different sites).
- The LV mirroring copies must be allocated on separate volumes that reside on different disk subsystem (on different sites).

When increasing the size of mirrored file system, is necessary to assure that the new logical partition will be allocated on different volumes and different disk subsystems according to the requirements above. For this task is always necessary to increase the logical volume first with appropriate volume selection first, then increase file system, preferably using C-SPOC (in this case, HACMP will enforce this).

Before configuring cross-site LVM mirroring environment, check for the following prerequisites:

- Configure the sites and resource groups and run the HACMP cluster discovery process.
- Ensure that both sites have copies of the logical volumes and that "*forced varyon*" attribute for a volume group is set to "*Yes"* if a resource group already contains a volume group.

16.2 Infrastructure considerations

Here we describe some consideration regarding SAN setup, fibre channel connections and LAN environment. The considerations and limitations are based on the technologies and protocols, used for cross-site mirroring cluster implementation.

SAN or network can be expanded beyond the original site, by the way of advanced technology.SG-245250-04

Here is an example of what kind of technology could be used for expansion. This list is not exhaustive:

- FCIP router
- Wave division multiplexing (WDM) devices. This technology include

- CWDM - Coarse Wavelength Division Multiplexing, which is the less expensive component among the WDM technology.
- DWDM stands for Dense Wave length Division Multiplexing.

16.3 Configuring cross-site LVM mirroring

In this section we show an example how to set-up and configure the cross-site LVM mirroring environment. In general you can make cross-site LVM mirroring configuration as a new cluster implementation. You could also change the existing local cluster by adding site dependencies and cross-site LVM features in the cluster configuration and integrate it within the cross-site environment.

16.3.1 Configure the cross-site LVM cluster

For our cross-site LVM mirroring test we configure the new cluster environment. Following the **Extended Configuration** smit menu, we first define the cluster topology including cluster definition, nodes, networks, network interfaces and non-IP over disk heartbeat paths as for normal cluster environment. Our nodes are named "*koper*" and "*nantes*"; *koper* resides on the site named *Slovenia* while *nantes* resides on the site, named *France* (see Figure 16-1 on page 627).

An important part in the cross-site environments are communication paths between the nodes on the both sites. The communication between the nodes consists of IP and non-IP connections. A non-IP connection is very important for cross-site cluster solutions to prevent node or site isolation ("split brain"). We configured the non-IP heartbeat using the heartbeat over disk feature.

Following the general recommendation for non-IP disk heartbeat networks (3.8, "Planning cluster networks" on page 163), we define two non-IP heartbeat networks. First network uses disk devices on ESS storage, that resides on the *France* site. The second network uses disk devices on FAStT/DS4xxx storage, that resides on the *Slovenia* site. This redundancy is set in order to keep alive the cluster non-IP heartbeats in case of one disk subsystem failure.

The Figure 16-1 on page 627 shows our test environment for cross-site LVM mirroring cluster testing. the following sections describes detailed configuration steps for configuring the necessary cross-site LVM specific in cluster configuration.

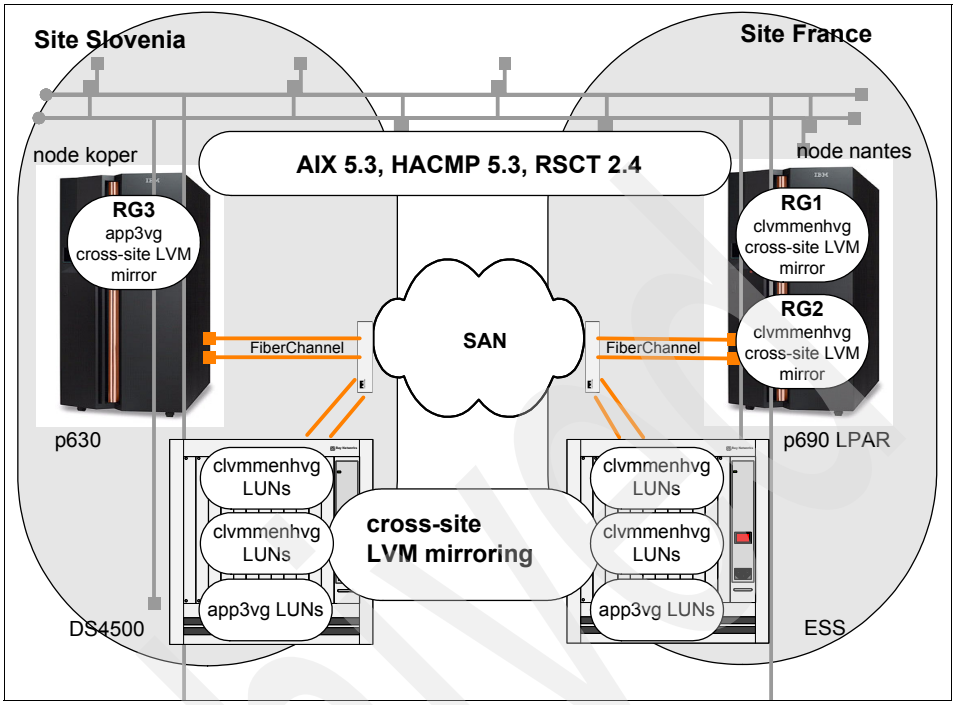

Figure 16-1 LVM cross-mirrored cluster testing environment

16.3.2 Configure cluster sites

You should use the HAMCP site definitions, when you configure cross-site LVM mirroring or any of the HACMP/XD components.

We configured the sites and add them to a cluster configuration, using smit menus. We run `smit hacmp` > **Extended Configuration > Extended Topology Configuration > Configure HACMP Sites > Add a Site.** We add the two sites France and Slovenia. Node koper is a part of Slovenia site while node nantes is a part of France site. The Figure 16-2 on page 628 shows the smit menu for site creation.

```
                                Add Site

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
                                                      [Entry Fields]
* Site Name                                          [France]             +
* Site Nodes                                          nantes              +
* Dominance                                          [Yes]                +
* Backup Communications                              [none]               +


F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Figure 16-2 Adding site in the topology configuration

The necessary field to fill for the definition of a site in the smit menu are as follows:

- Site name. Define a name of a site, used for different site dependencies assignment in further configuration
- Site nodes. For each site we define a list of nodes, residing on the site.
- Dominance. Selecting *Yes* we define this site as the dominant site for site isolation shutdown
- Backup Communications. Define backup communications type for site isolation detection (none, sgn, dbfs).

16.3.3 Configure cross-site LVM mirroring site dependencies

After we define the cluster topology with site dependencies, we assign the specific disk devices to the appropriate site. The storage on *France* site is ESS 2105-800, while the storage on *Slovenia* site is DS4500. The servers are using MPIO for ESS, and RDAC driver for DS4xxx storage. The SAN zoning configuration is created in such a way that each server can accesses the ESS disk subsystem through one fiber channel adapter, while accessing the DS4xxx storage through a different one. The Example 16-1 shows the disk list on *nantes* node. We plan to use the hdisk2 through hdisk6 and hdisk10 through hdisk14 for cross-site LVM mirroring configuration, while hdisks7 through hdisk9 are free for using in a non-mirrored volume group.

Example 16-1 lsdev -Cc disk output on nantes node

```
nantes /usr/es/sbin/cluster > lspv
hdisk0          0022be2ab1cd11ac                    rootvg          active
hdisk1          0022be2abc247c91                    altinst_rootvg
```

```
hdisk2          0022be2a8607249f                    clvmmenhvg      active
hdisk3          0022be2a0bfe1f60                    clvmmshvg       active
hdisk4          0022be2a86607918                    clvmmshvg       active
hdisk5          0022be2a08d4844e                    clvmmenhvg      active
hdisk6          0022be2a8662ce0e                    app3vg
hdisk7          0022be2a0bfe9e19                    None
hdisk8          00257400b4d32054                    None
hdisk9          0022be2a86630978                    None
hdisk10         0022be2a0bfe9eec                    clvmmenhvg      active
hdisk11         0022be2a0bfe9f8a                    clvmmenhvg      active
hdisk12         0022be2a0bfea026                    clvmmshvg       active
hdisk13         0022be2a0bfea0ca                    clvmmshvg       active
hdisk14         0022be2a11690be3                    app3vg
```

We use the smit for assigning the site / disk dependencies by running **smit hacmp > System Management (C-SPOC) > HACMP Physical Volume Management > Configure Disk/Site Locations for Cross-Site LVM Mirroring > Add Disk/Site Definition for Cross-Site LVM Mirroring.** You can also use the smit fast-path **smit cl_xslvmm** for accessing the **Configure Disk/Site Locations for Cross-Site LVM Mirroring** directly.

Note: The “Configure Disk/Site Locations for Cross-Site LVM Mirroring” menu selection functions correctly only, if the "Disk Discovery File" reflects the current disk configuration. We recommend to run HACMP discovery before configuring disk/site dependencies.

Attention: All disk should have PVIDs assigned before running the “Discover HACMP-related Information from Configured Nodes” in order to have complete disk information stored into the “Disk Discovery File”. You can do this with the command **chdev -l hdiskX -a pv=yes** on one node and then you remove and reconfigure disk devices on the other nodes (**rmdev..., cfgmgr**).

After we use the **Add Disk/Site Definition for Cross-Site LVM Mirroring menu** selection, we first select the site for our definition, as shown in Example 16-2. After that we select the disks that reside in the selected site, as shown in Example 16-3 on page 630.

Example 16-2 Site selection in the Add Site/Disk Definition smit menu

```
¦ Move cursor to desired item and press Enter.                          ¦

¦   France
```

```
¦   Slovenia
```

Example 16-3 Disk selection in the Add Site/Disk Definition smit menu

```
¦ [MORE...10]                                                        ¦
¦   0022be2a0bfea026        hdisk13 koper                            ¦
¦   0022be2a0bfea0ca        hdisk14 koper                            ¦
¦ > 0022be2a8607249f        hdisk2  nantes                           ¦
¦ > 0022be2a0bfe1f60        hdisk3  nantes                           ¦
¦ > 0022be2a86607918        hdisk4  nantes                           ¦
¦ > 0022be2a08d4844e        hdisk5  nantes                           ¦
¦   0022be2a8662ce0e        hdisk6  nantes                           ¦
¦   0022be2a0bfe9e19        hdisk7  nantes                           ¦
¦ [MORE...6]                                                         ¦
```

You can change the site / disk dependency later by running **smit cl_xslvmm** and selecting **Change/Show Disk/Site Definition for Cross-Site LVM Mirroring.** You can also remove the site and disk dependency later by running **smit cl_xslvmm** and selecting **Remove Disk/Site Definition for Cross-Site LVM Mirroring**.

16.3.4 Configure volume groups with cross-site LVM mirror

For our cross-site LVM mirroring test we created three volume groups. These are:

- clvmmshvg - shared non-concurrent
- clvmmenhvg - enhanced concurrent
- app3vg - enhanced concurrent.

We created our VGs using smit C-SPOC menus. We run **smit cl_admin** and then select **HACMP Logical Volume Management** for shared volume groups or HACMP **Concurrent Logical Volume Management** for both enhanced concurrent volume groups.

After selecting the participating nodes, we select the disks, as shown in Example 16-4. Then we fill all necessary fields in VG creation smit screen and we set the *Enable Cross-Site LVM Mirroring Verification* option to *true*. The Example 16-5 on page 631 shows the smit VG creation screen.

Example 16-4 Selecting the disk for VG creation in smit menu

```
¦ [MORE...1]                                                         ¦
¦ > 0022be2a86607918        France                                   ¦
¦ > 0022be2a0bfe1f60        France                                   ¦
¦ > 0022be2a0bfea026        Slovenia                                 ¦
¦ > 0022be2a0bfea0ca        Slovenia                                 ¦
```

```
¦   0022be2a86630978                                                    ¦
¦   0022be2a0bfe9f8a        Slovenia                                    ¦
¦   0022be2a0bfe9eec        Slovenia                                    ¦
¦   0022be2a0bfe9e19                                                    ¦
```

Example 16-5 shows how to create a shared volume group:

Example 16-5 VG creation smit screen

```
Create a Shared Volume Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

[TOP]                                                  [Entry Fields]
  Node Names                                          koper,nantes
  PVID                                                0022be2a86607918 0022>
  VOLUME GROUP name                                   [clvmmshvg]
  Physical partition SIZE in megabytes                 128                  +
  Volume group MAJOR NUMBER                           [44]                   #
  Enable Cross-Site LVM Mirroring Verification         true                 +

  Warning:
  Changing the volume group major number may result
  in the command being unable to execute
  successfully on a node that does not have the
  major number currently available.  Please check
  for a commonly available major number on all nodes
[MORE...2]
```

16.3.5 Configure an RG with cross-site LVM mirroring enabled VG

After we have all topology and LVM components defined in the cluster, we created resource groups. For two resource groups, *nantes* node is primary node. For the third resource group the primary cluster node is *koper*. Each resource group owns one volume group, which is cross-site mirrored.

The way of creating the resource group in a cross-site cluster environment is similar to a normal cluster configuration. The additional setting for site-dependent resource groups in the cluster environment, where sites are defined, is the *Inter-Site Management Policy*. The possible selections for this feature are:

- **Ignore**. This is default selection and ignores the site dependency settings for the resource group.
- **Prefer Primary Site**. The resource group may be assigned to be taken over by multiple sites in a prioritized manner. When a sites fails, the active site with

the highest priority acquires the resource. When the failed site rejoins, the site with the highest priority acquires the resource.

- **Online On Either Site**. The resources group may be acquired by any site in its resource chain. When a site fails, the resource group is acquired by the highest priority standby site. When the failed site rejoins, the resource group remains with its new owner.
- **Online On Both Sites**. The resource group is acquired by both sites. This selection defines the concurrent capable resource group.

After we defined the resource groups, we configure the resource group attributes like volume group, service IP, and the application server. The important parameter while adding the cross-site mirror enabled volume group in the resource group is the *"Use forced varyon of volume groups, if necessary"* field. You must set this field to *true* for any cross-site LVM mirroring configuration. This assures that in case of one storage or site failure, that specific volume group could be varied on the other node with only one (surviving) LV copy.

We add logical volumes and file system into the volume groups, after we defined resource groups and assign our volume groups in the resource groups. We configure this in the usual way of cluster configuring, by using C-SPOC. You can find more information about LVM component creation in the 8.4, "Shared storage management" on page 382.

For each resource group we define the application server. The application server runs a test application that provides an intensive writing on the file systems part of this resource group. With this load we are able to achieve the specific disk utilization between 70 and 100 percent.

After we set up the cluster environment, we activate the automatic error notification. We run **`smit hacmp`** > **Problem Determination Tools > HACMP Error Notification > Configure Automatic Error Notification > Add Error Notify Methods for Cluster Resources**. You can find more information about the error notification in the 12.3, "Error notification" on page 560.

Note: DARE is not supported in an active cluster with sites defined. You can use C-SPOC for some cluster configuration changes. All the other configuration changes must be done when the cluster is not active.

16.4 Testing cross-site LVM mirroring

After we configure the cluster topology and resources, we synchronize and verify the cluster. We start the cluster and check if all resources and the communication paths are active. The **`cldump`** utility shows the information about the cluster nodes and network interfaces, as well as resource group status (including resource

group policies for each resource group). Example 16-6 shows the **cldump** output from our test cluster environment.

Example 16-6 cldump output on our cross-mirror cluster environment

```
koper /usr/es/sbin/cluster/utilities > cldump

Obtaining information via SNMP from Node: nantes...

_____________________________________________________________________________
Cluster Name: crossitelvm
Cluster State: UP
Cluster Substate: STABLE
_____________________________________________________________________________


Node Name: koper                State: UP

  Network Name: net_diskhb_01      State: UP

    Address:                Label: diskhb_sloveniatonantes State: UP

  Network Name: net_ether_01       State: UP

    Address: 10.10.1.9       Label: koper_base1        State: UP
    Address: 10.10.2.9       Label: koper_base2        State: UP
    Address: 192.168.100.117 Label: app3svc            State: UP


Node Name: nantes               State: UP

  Network Name: net_diskhb_01      State: UP

    Address:                Label: diskhb_francetoslovenia State: UP

  Network Name: net_ether_01       State: UP

    Address: 10.10.1.2       Label: nantes_base1       State: UP
    Address: 10.10.2.2       Label: nantes_base2       State: UP
    Address: 192.168.100.86  Label: app1svc            State: UP
    Address: 192.168.100.87  Label: app2svc            State: UP



Cluster Name: crossitelvm

Resource Group Name: RG1
Startup Policy: Online On Home Node Only
Fallover Policy: Fallover To Next Priority Node In The List
```

```
Fallback Policy: Fallback To Higher Priority Node In The List
Site Policy: ignore
Priority Override Information:
    Primary Instance POL:
Node                          Group State
----------------------------- ---------------
nantes                        ONLINE
koper                         OFFLINE

Resource Group Name: RG2
Startup Policy: Online On Home Node Only
Fallover Policy: Fallover To Next Priority Node In The List
Fallback Policy: Fallback To Higher Priority Node In The List
Site Policy: ignore
Priority Override Information:
    Primary Instance POL:
Node                          Group State
----------------------------- ---------------
nantes                        ONLINE
koper                         OFFLINE

Resource Group Name: RG3
Startup Policy: Online On Home Node Only
Fallover Policy: Fallover To Next Priority Node In The List
Fallback Policy: Fallback To Higher Priority Node In The List
Site Policy: ignore
Priority Override Information:
    Primary Instance POL:
Node                          Group State
----------------------------- ---------------
koper                         ONLINE
nantes                        OFFLINE
```

16.4.1 Tested scenarios

Graceful cluster shutdown with takeover

First we test the graceful shutdown with takeover on the node *nantes*. The cluster services on the *nantes* node stop and resource groups RG1, RG2 activate on *Slovenia* site node *koper* as expected. After test we start the *nantes* node and cluster initiate the move of the resource groups RG1 and RG2 back to their primary node (*nantes*), as defined in the resource group policy. All resources for RG1 and RG2 became available on *nantes* node and the applications became active.

Move one resource group to the other site.

For next test we select the *nantes* node, which owns two resource groups. With the C-SPOC *Move a Resource Group to Another Node / Site selection* we moved only RG1 resource group to the node *koper*. All resources for RG1 have been moved to *koper* node, while RG2 resource group remains active on the node *nantes* without interruption, as expected.

One storage subsystem failure

For the following test we simulate the ESS (primary) storage failure. We test the storage failure while all three test applications are active and extensive load on the disks exists. The utilization of the disks is near 100% for all test file systems. We simulate two different types of the storage failure.

For the first simulation we de-assigned all LUNs owned by our cross-site mirroring enabled volume groups. This simulates a logical internal failure in the storage. For the second failure simulation, we disconnect fibre channel cable on the *nantes* node to simulate the connection failure.

In both cases, the applications continue to work without interruption, and the volume groups and file systems remain available. After the failure we check the availability of disk and the status of logical volume copy synchronization. The disks from ESS storage are marked as *missing* and the logical volume status as *stale*. Example 16-7 shows the output of *clvmmenhvg* VG when ESS disk subsystem is not available.

Example 16-7 hdisk and logical volume status after one storage subsystem failure

```
nantes > lsvg -p clvmmenhvg
clvmmenhvg:
PV_NAME           PV STATE          TOTAL PPs   FREE PPs    FREE DISTRIBUTION
hdisk11           active            79          78          16..15..15..16..16
hdisk10           active            79          59          16..00..11..16..16
hdisk2            missing           74          73          15..14..14..15..15
hdisk5            missing           74          54          15..00..09..15..15
nantes > lsvg -l clvmmenhvg
clvmmenhvg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
cenglog2lv          jfs2log    1     2     2    open/stale    N/A
enhtest1lv          jfs2       20    40    2    open/stale    /app1fs
```

The applications continue to work with the remaining logical volume copy. After this test we make ESS storage available again. We used the CSPOC option *Synchronize Shared LVM Mirrors* and it automatically makes all hdisk devices available and synchronizes all logical volumes. We run **smit cl_admin->HACMP Logical Volume Management->Synchronize Shared LVM Mirrors->Synchronize by Volume Group,** then select the appropriate VG.

We verified the disk availability and the status of the logical volume copy synchronization. All disks in all volume groups are available and all logical volumes are in *synch* status. Example 16-8 shows the situation on the *clvmmenhvg* volume group after the storage reintegration.

Example 16-8 hdisk and logical volume status after the storage reintegration

```
nantes > lsvg -p clvmmenhvg
PV_NAME           PV STATE          TOTAL PPs   FREE PPs    FREE DISTRIBUTION
hdisk11           active            79          78          16..15..15..16..16
hdisk10           active            79          59          16..00..11..16..16
hdisk2            active            74          73          15..14..14..15..15
hdisk5            active            74          54          15..00..09..15..15
nantes > lsvg -l clvmmenhvg
clvmmenhvg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
cenglog2lv          jfs2log    1     2     2    open/syncd    N/A
enhtest1lv          jfs2       20    40    2    open/syncd    /app1fs
```

Failure of all disk connections on one site

For the following test we disconnect both fiber channel connections on site *France* and *nantes* node. After some time delay (couple of minutes) the cluster detected the storage failure for all file systems on shared VG and moved both resource groups RG1 and RG2 to another site *Slovenia*. Cluster activates both resource groups on the *koper* node, makes **varyonvg** of all volume groups, mounts file systems and starts applications. All resources are available on *koper* node, volume groups from RG1 and RG2 are activated.

Site failure

For the following test we simulate *France* site failure by simultaneously crashing *nantes* node and ESS disk subsystem fiber channel connections. The cluster detects the site failure and both RG1 and RG2 are moved to site *Slovenia*. Cluster activates both resource groups on the *koper* node, makes **varyonvg** of all volume groups, mounts file systems and starts applications. All resources are available on *koper* node, the volume groups belonging to RG1 and RG2 are activated with the surviving disk copy (DS4500 storage). The third resource group RG3 (on *koper*) works without interruption.

After this test we connected back the ESS disk subsystem, so that the ESS disk resources are available again and activate *nantes* node. Cluster initiates the movement of RG1 and RG2 back to their primary (*nantes*) node, as defined in the resource group policy. All resources for RG1 and RG2 became available on *nantes* node and the applications started.

17

HAGEO disaster recovery scenario

This chapter describes a disaster recovery scenario based on HACMP/XD HAGEO. We provide a detailed description of the steps needed to configure a HAGEO environment. We cover the following topics:

- Scenario description
- Planning the HAGEO configuration
- Installation and configuration of the HACMP/XD HAGEO
- Fallover and fallback considerations

17.1 Description of the scenario and planning

Our scenario uses three nodes in two sites: Boston and Munchen. Figure 17-1 details the node distribution and communications path between the two sites. An external client is able to access the public network at each site.

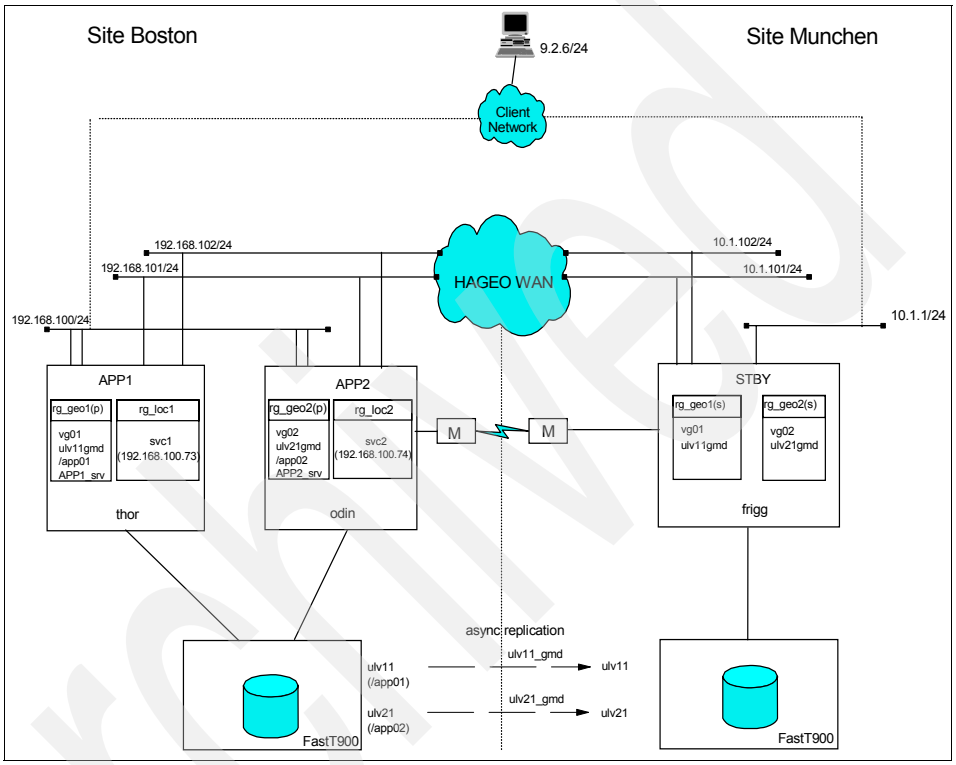

Figure 17-1 HAGEO scenario

We use node thor and odin in a mutual takeover configuration at site Boston, and node frigg as a standby node in site Munchen. Each node uses 2 ethernet interfaces for data replication between the sites.

The two geographical networks provide high availability and load balancing for the replication mechanism. A serial rs232 connection is used as a non-IP heartbeating paths between the sites. There are two resource groups built around the applications APP01 and APP02 running on top of the geo mirror devices.

17.1.1 Planning the network configuration

We define the following HACMP networks along with their subnets:

- a public network in site Boston, for the client access to the applications APP01 and APP02: 172.1.1.0/24. We use a single subnet for the local network of site Boston, because we use heartbeat over aliases for this network. Our heartbeat subnet is: 172.16.100.0/24.

  There is no HACMP network for the client network in site Munchen, since there is a single node with one ethernet interface attached to this network. The subnet used for this network is: 10.1.1.0/24.

- geo replication networks:
  - geo1: 192.168.101.0/24 (site Boston) and 10.1.101.0/24 (site Munchen);
  - geo2: 192.168.102.0/24 (site Boston) and 10.1.102.0/24 (site Munchen);
- disk heartbeat network in site Boston, as a non-IP heartbeat path between nodes odin and thor
- a secondary geographical network for serial heartbeat rs232 between nodes odin (Boston) and frigg (Munchen).

Table 17-1 shows the IP address configuration of the nodes in the cluster.

Table 17-1 Node interface configuration

Hostname	Interface name	IP address/ Netmask	AIX interface	Purpose
thor	thor_boot1	172.1.1.73/24	en0	boot
	thor_boot2	172.1.1.75/24	en1	boot
	thor	192.168.100.73/24	N/A	persistent
	thor_svc	192.168.100.75/24	N/A	service
	thor_geo1	192.168.101.73/24	en2	Geo_primary
	thor_geo2	192.168.102.73/24	en3	Geo_primary
odin	odin_boot1	172.1.1.74/24	en0	boot
	odin_boot2	172.1.1.77/24	en1	boot
	odin	192.168.100.74/24	N/A	persistent
	odin_svc	192.168.100.77/24	N/A	service
	odin_geo1	192.168.101.74/24	en2	Geo_primary
	odin_geo2	192.168.102.74/24	en3	Geo_primary

Hostname	Interface name	IP address/ Netmask	AIX interface	Purpose
frigg	frigg_geo1	10.1.101.192/24	en0	Geo_primary
	frigg_geot2	10.1.102.192/24	en2	Geo_primary
	frigg	10.1.1.192/24	en1	boot

We use a single subnet for the local network of site Boston, because we use the heartbeating over aliases.

17.1.2 Planning the logical volume configuration

The geo-mirror device definition requires creating the logical volumes with the same names at both sites. The volume group name containing the geo devices is used in the replicated resource group of HACMP, and it must also use the same in both sites.

Besides the logical volume it maps to, each geo-mirror device uses a state map logical volume to keep a record of the un-synchronized data of the local and remote hosts. When creating the state map logical volume you have to consider the size of the logical volume it is associated with, according to the following formula:

statemap size = max LV size / (region size x 2)

The max LV size is an estimation of the maximum capacity for the logical volume. The region size represents the size of the data block from the logical volume, which is mapped by a 4-bit data structure in the statemap logical volume. By default the region size is 32768 bytes(32 KB).

For example, we use ulv11 with a maximum size of 10 GB(160 PPs of 64MB PP size). The actual size of the statemap is:

10*1024*1024 KB/ (32KB x2) = 160 MB

The size for the statemap logical volume must be rounded to the PP size increments, so 192 MB (3 PPs) are actually allocated for the logical volume. Table 17-2 shows our configuration of the logical volumes we use in the HAGEO configuration.

Table 17-2 Logical volumes at primary and secondary site

Logical Volume	Volume group	Size (PP=128MB)	Sites
ulv11	vg01	160	Boston, Munchen

Logical Volume	Volume group	Size (PP=128MB)	Sites
ulv11_sm	vg01	3	Boston, Munchen
ulv11_log	vg01	1	Boston, Munchen
ulv11_log_sm	vg01	1	Boston, Munchen
ulv21	vg02	160	Boston, Munchen
ulv21_sm	vg02	3	Boston, Munchen
ulv21_log	vg02	1	Boston, Munchen
ulv21_log_sm	vg02	1	Boston, Munchen

Each logical volume has an statemap logical volume defined. The name and the size of the logical volume must be the same at both sites.

Note: For performance considerations we recommend allocating the statemaps on separate physical volumes than the data logical volumes.

17.1.3 GMD definition

For our configuration we defined four GMDs corresponding to two file systems. APP01 and APP02 are two generic application using data in file systems /app01 and /app02. Each logical volume has an associated statemap. Table 17-3 shows the geo device configuration.

Table 17-3 GMD definition

GMD name	Minor no.	Statemap LV	Logical Volume	Device mode	File system
ulv11_gmd	10	ulv11_sm	ulv11	async	/app01
ulv11_log_gmd	11	ulv11_log_sm	ulv11_log	async	N/A
ulv21_gmd	20	ulv21_sm	ulv21	async	/app02
ulv21_log_gmd	21	ulv21_log_sm	ulv21_log	async	N/A

17.2 HAGEO installation and configuration

We used in our scenario the following software components:

- AIX 5.3 ML02, RSCT version 2.4.2

- HACMP 5.3
- HAGEO 5.3

The installation of the HAGEO software requires that HACMP filesets are installed.

The following HACMP/XD HAGEO were installed:

- cluster.xd.license
- hageo.doc.en_US
- hageo.gmdsizing
- hageo.man.en_US
- hageo.manage
- hageo.message
- hageo.mirror

Example 17-1 shows a listing of the hageo filesets and versions, we used in our configuration.

Example 17-1 Listing of the hageo filesets installed

```
  Fileset                      Level  State  Type  Description (Uninstaller)
  ----------------------------------------------------------------------------
hageo.doc.en_US.data        5.3.0.0    C     F    HAGEO Product Manuals - U.S.
                                                    English
hageo.gmdsizing             5.3.0.0    C     F    GMD Sizing Demonstration Tool
hageo.man.en_US.message.data
                            5.3.0.0    C     F    HAGEO GeoMessage Man Pages -
                                                    U.S. English
hageo.man.en_US.mirror.data
                            5.3.0.0    C     F    HAGEO GeoMirror Man Pages -
                                                    U.S. English
hageo.manage.utils          5.3.0.0    C     F    HAGEO GeoManage Utilities
hageo.message.ext           5.3.0.0    C     F    HAGEO GeoMessage Device Driver
hageo.message.utils         5.3.0.0    C     F    HAGEO GeoMessage Utilities
hageo.mirror.ext            5.3.0.0    C     F    HAGEO GeoMirror Device Driver
hageo.mirror.utils          5.3.0.0    C     F    HAGEO GeoMirror Utilities
```

For further details about installing the HAGEO filesets, refer to *High Availability Cluster Multi-Processing XD (Extended Distance) for HAGEO Technology: Planning and Administration*, SA22-7956.

Configure the IP addresses of the adapters

We set up the boot IP addresses on the nodes, according to Table 17-1 on page 639. Besides the subnets described in this table, we use a dedicated subnet for heartbeating over aliases: 172.16.100.1/24.

The service and the heartbeat addresses are aliases over the boot IP labels, which are activated by the cluster services. The persistent IP address is an alias address bounded to a node, which stays active on that node even after stopping the cluster services or rebooting the system.

Define the logical volumes

On both sites we create similar logical volume configurations. Assuming that the logical volumes and file systems are already defined in the primary site, we define the logical volumes in the backup site Munchen. In Example 17-2 we create the replicated logical volumes on node frigg in site Munchen. Note that the type *statemap* we used is a descriptive attribute of the logical volume without any functional role.

Example 17-2 Creating the volumes and file systems at the remote site Munchen

```
frigg:/# mkvg -y vg01 -f -c hdisk1
frigg:/# varyonvg vg01
frigg:/# mklv -y ulv11_log -t jfs2log vg01 1
frigg:/# logform /dev/ulv11_log
logform: destroy /dev/rulv11_log (y)?y

frigg:/# mklv -y ulv11 -t jfs2 vg01 160

frigg:/# mkvg -y vg02 -f -c hdisk2
frigg:/# varyonvg vg02
frigg:/# mklv -y ulv21_log -t jfs2log vg02 1
frigg:/# logform /dev/ulv21_log
logform: destroy /dev/rulv21_log (y)?y

frigg:/# mklv -y ulv21 -t jfs2 vg02 160

frigg:/# mklv -y ulv11_sm -t statemap vg01 3
frigg:/# mklv -y ulv21_sm -t statemap vg02 3
frigg:/# mklv -y ulv11_log_sm -t statemap vg01 1
frigg:/# mklv -y ul211_log_sm -t statemap vg02 1
```

Note: Keep the logical volumes and the volume group definitions consistent at both sites, for integrating into replicated resource groups in HACMP. As example, if you create vg01, as an enhanced-concurrent volume group at site Boston, an enhanced-concurrent volume group with the same name must be created in site Munchen.

Defining the HACMP topology

Before configuring the mirror devices, you need to configure the cluster topology. In our scenario we describe the steps performed for defining the cluster topology:

1. Define the cluster name.

   Cluster name = itso

2. Configure the nodes, along with their communication paths.

   Node names: thor, odin, frigg

   Communication path: thor_geo1, odin_geo1, frigg_geo1

3. Run the discovery process, to acquire the IP and disk information from all nodes in the cluster.
4. Configure the HACMP sites.

We configure the sites: Boston and Munchen as in Example 17-3 and Example 17-4 on page 645.

Example 17-3 Defining the site Boston

```
Site Name                                       [Boston]
* Site Nodes                                     odin thor
* Dominance                                     [Yes]
* Backup Communications                         [sgn]
```

The Dominance field defines which site will be halted when site isolation occurs. Site isolation happens when all the geographic networks are down, but at least one node at each site is still up. To prevent data divergence, the nondominant site is halted.

The backup communication field defines an alternate way for the sites to stay in contact. The possible values are:

- sgn (Secondary Geographical Network)
- dbfs (Dial Back Fail Safe)
- none.

It is highly recommended to define a backup communication network in your cluster, since it helps preventing the site isolation in case of primary geo networks going down.

Example 17-4 Defining the site Munchen

```
* Site Name                                    [Munchen]
* Site Nodes                                   frigg
* Dominance                                    [No]
* Backup Communications                        [sgn]
```

5. Configure the HACMP networks

The following networks are defined in the HACMP cluster:

- the client LAN for site Boston: boston_ether_01

Example 17-5 details the definition of boston_ether_01 network in HACMP. Note that at this step, we specify the aliased heartbeat network.

Example 17-5 Defining the public network of site Boston

```
* Network Name                                 [boston_ether_01]
* Network Type                                 ether
* Netmask                                      [255.255.255.0]
* Enable IP Address Takeover via IP Aliases    [Yes]
  IP Address Offset for Heartbeating over IP Aliases [172.16.100.1]
```

- Local disk heartbeat network for site Boston: boston_diskhb_01.

  The disk heartbeat networks is used as a non-IP heartbeat network between node thor and odin in site Boston. Example 17-6 shows the disk heartbeat definition.

Example 17-6 Defining the disk heartbeat network

```
* Network Name                                 [boston_diskhb_01]
* Network Type                                 diskhb
```

- the geographical replication networks: net_Geo_Primary_01 and net_Geo_Primary_02.

  HACMP uses Geo_Primary network type for data replication with HAGEO. We define a Geo_Primary network in Example 17-7.

Example 17-7 Defining the Geo_Primary networks

```
* Network Name                                 [net_Geo_Primary_02]
* Network Type                                 Geo_Primary
* Netmask                                      [255.255.255.0]
```

```
* Enable IP Address Takeover via IP Aliases           No
```

By default the Geo_Primary network gets created with a public attribute. You have two options for configuring this network:

- Use a public network. At the time the communication interfaces are added there is no service IP label defined for the network. You have to define service node-bounded IP addresses, in the resource group definition menus.
- Use a private network, so that addresses you define in topology, by adding the communications interfaces to the network, are service IP labels.

In our scenario we use private HACMP networks, so we change the initial definition of the network from public to private, as in Example 17-8.

Example 17-8 Changing the network type attribute

```
* Network Name                                        itso_Geo_Primary_01
  New Network Name                                   []
* Network Type                                       [Geo_Primary]          +
* Netmask                                            [255.255.255.0]        +
* Enable IP Address Takeover via IP Aliases          No                     +
  IP Address Offset for Heartbeating over IP Aliases []
* Network attribute                                  private                +
```

Note: The Geo_Primary network must have service IP addresses signed for the geo mirror devices communication.

- the secondary geographical network, is an rs232 network which connects the sites, between nodes odin and frigg. Its definition is shown in Example 17-9.

Example 17-9 Defining the Geo_Secondary network

```
* Network Name                                       [net_Geo_Secondary_01]
* Network Type                                       Geo_Secondary
```

6. Add the communication interfaces and devices for the defined networks. At this step we populate the previously defined networks with their associated interfaces.

- For the client network in site Boston
- For primary geo networks, we add the IP address like in Example 17-10.

Example 17-10 Defining an interface on the Geo_Primary

```
* IP Label/Address                                   [thor_geo1]
* Network Type                                       Geo_Primary
```

```
* Network Name                                        net_Geo_Primary_01
* Node Name                                          [thor]
  Network Interface                                  []
```

- For the secondary geo network (see Example 17-11).

Example 17-11 Defining a Geo_Secondary network interface

```
* Device Name                                        [odin_tty0]
* Network Type                                        Geo_Secondary
* Network Name                                       itso_Geo_Secondary_01
* Device Path                                        [/dev/tty0]
* Node Name                                          [odin]                  +
```

- and for serial disk heartbeat (Example 17-12):

Example 17-12 Defining a serial disk heartbeat interface

```
* Device Name                                        [odin_hdisk2]
* Network Type                                        diskhb
* Network Name                                        boston_diskhb_01
* Device Path                                        [/dev/hdisk2]
* Node Name                                          [odin]
```

7. Configure persistent IP addresses (Example 17-13): odin, thor. We defined the hostname interfaces as persistent addresses in cluster. Node frigg has a single IP interface in the public net at site Munchen. Since it is the single node at site Munchen, we did not define the hostname interface in HACMP.

Example 17-13 Configuring the persistent IP addresses

```
* Node Name                                           thor
* Network Name                                       [boston_ether_01]
* Node IP Label/Address                              [thor]
```

8. Synchronize the cluster topology defined so far.

   Our configuration defined is detailed in the output of **cltopinfo** command from Example 17-14.

Example 17-14 Output of cltopinfo command

```
Cluster Name: itso
Cluster Connection Authentication Mode: Standard
Cluster Message Authentication Mode: None
Cluster Message Encryption: None
Use Persistent Labels for Communication: No
```

```
There are 3 node(s) and 5 network(s) defined
NODE frigg:
        Network boston_diskhb_01
        Network boston_ether_01
        Network net_Geo_Primary_01
                frigg_geo1      10.1.101.192
        Network net_Geo_Primary_02
                frigg_geo2      10.1.102.192
        Network net_Geo_Secondary_01
                frigg_tty0      /dev/tty0
NODE odin:
        Network boston_diskhb_01
                odin_hdisk2     /dev/hdisk2
        Network boston_ether_01
                odin_boot2      172.1.1.77
                odin_boot1      172.1.1.74
        Network net_Geo_Primary_01
                odin_geo1       192.168.101.74
        Network net_Geo_Primary_02
                odin_geo2       192.168.102.74
        Network net_Geo_Secondary_01
                odin_tty0       /dev/tty0

NODE thor:
        Network boston_diskhb_01
                thor_hdisk2     /dev/hdisk2
        Network boston_ether_01
                thor_boot1      172.1.1.73
                thor_boot2      172.1.1.75
        Network net_Geo_Primary_01
                thor_geo1       192.168.101.73
        Network net_Geo_Primary_02
                thor_geo2       192.168.102.73
        Network net_Geo_Secondary_01
No resource groups defined
```

Define the Geo mirror devices

1. Create all geo-mirror devices, using smitty hageo → Configure GeoMirror Devices → Configure a GeoMirror Device → Add a GeoMirror Device.

   In Example 17-15 we define the ulv11gmd corresponding to ulv11 logical volume and associate ulv11_sm logical volume as the statemap device.

Example 17-15 Defining the GMD using smit

```
Add a GeoMirror Device

Type or select values in entry fields.
```

```
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Device Name                                          [ulv11gmd]
* Minor Device Number                                    [10]
* State Map Logical Volume                           [/dev/rulv11_sm]
* State Map Size (Number of Entries)                   [1024]
* State Map Region Size                               [32768]
* Local Logical Volume                              [/dev/rulv11]
* Device Mode                                          async
* Device Role                                        primary
  High Water Mark                                      []
  Sync Concurrency Rate                                []
* Remote Node, LV, and Statemap          [frigg@/dev/rulv11@/dev/rulv11_sm]
  Remote Node, LV, and Statemap                        []
  Remote Node, LV, and Statemap                        []
  Remote Node, LV, and Statemap                        []
  Remote Node, LV, and Statemap                        []
  Remote Node, LV, and Statemap                        []
  Remote Node, LV, and Statemap                        []
  Local Peer and State Map Device                   [thor@/dev/rulv11_sm]
  Local Peer and State Map Device                      []
  Local Peer and State Map Device                      []
  Local Peer and State Map Device                      []
  Local Peer and State Map Device                      []
  Local Peer and State Map Device                      []
```

2. Synchronize GMD definition across the nodes

   Use smitty hageo → Configure GeoMirror Devices → Synchronize GeoMirror Devices.

3. Configure the Global Mirroring Proprieties. Use smitty hageo → Configure GeoMirror Devices → Configure Global GeoMirror Properties.

We use the following settings in our scenario (Example 17-16):

Example 17-16 Global mirroring properties

```
GMD(s) for HACMP to start in parallel                [1]
Network Protocol                                     [TCP]
  Temporal Ordering Policy                           [SYSTEM]
  Autoset Network Parameters                         [Yes]
  TCP Send/Receive Space Size (KBytes)               [512]
```

4. Synchronize the Geomirror Properties:

   Use smitty hageo → Configure GeoMirror Devices → Synchronize Global GeoMirror Properties

5. Verify the GMD definition, by using the geo_verify utility, or via smitty hageo → Verify HAGEO configuration.
6. On each cluster node, link the file systems /app01 and /app02 with the geo devices. Edit the /etc/filesystems and replace the file system logical volume and the log logical volume with the GMD devices as in Example 17-17.

Example 17-17 Defining the application file systems on GMDs

```
/app01:
        dev             = /dev/ulv11_gmd
        vfs             = jfs2
        log             = /dev/ulv11_log_gmd
        mount           = false
        check           = false
        account         = false
/app02:
        dev             = /dev/ulv21_gmd
        vfs             = jfs2
        log             = /dev/ulv21_log_gmd
        mount           = false
        check           = false
        options         = rw
        account         = false
```

7. Test the newly created GMDs and the file systems
   a. Load the geo kernel extension on the nodes:

   /usr/sbin/hageo/krpc/cfgkrpc -ci

   b. Configure the GMD devices

   Activate the volume groups on the nodes and configure the geo devices using the cfggmd command:

   /usr/lib/methods/cfggmd -l <gmd_name>

   c. Start the gmd devices

   On the primary site, on each node, before starting the gmd mark the geo device as down on remote node, using **gmddown** command. In our scenario, we use nod thor in primary site, and mark ulv11_gmd as down on node frigg.

   On node thor:

   /usr/lib/methods/gmddown -l ulv11_gmd frigg

   Start the gmd devices on the local node:

   /usr/lib/methods/startgmd -l ulv11_gmd

   On the remote node, mark the local peer of node thor down and start the geo devices. In our example, ulv11_gmd is activated on node thor. On the node frigg, we mark the gmd as down for odin and start the device:

```
/usr/lib/methods/gmddown -l ulv11_gmd odin
/usr/lib/methods/startgmd -l ulv11_gmd
```

Note: Before the actual start of the geo device, we mark the geo device as down on nodes were the geo device is configured, but not started. This prevents startgmd command to time-out contacting the remote peer.

d. Mount the file systems on the primary node

e. For releasing the geo devices, you have to umount the file systems, then stop the geo devices on the primary and secondary location, using the following sequence of commands at each location:

Stop the geo mirror device:

```
stopgmd -l < gmd_name>
```

Unconfigure the gmd:

```
ucfggmd -l < gmd_name>
```

Unload the kernel extension:

```
/usr/sbin/hageo/krpc/cfgkrpc -u
```

Define the HACMP/XD resource groups

We have created four resource groups:

- Two replicated resource groups containing the volume groups, file systems and GMDs, corresponding to applications APP01 and APP02, normally running in Boston, on node thor and respectively odin. They are activated in both sites at the same time: a primary instance in Boston and a secondary instance in site Munchen.
- Two resource groups containing the service IP labels: odin_svc and thor_svc available only at site Boston.

Defining the resource groups

Considerations for configuring the resources groups:

- you cannot mix site-dependent with cross-site resources in a resource group. In our scenario, the service IP labels odin_svc and thor_svc are available only in Boston, so they cannot be included in the same resource group with the geo devices.
- if you define dependencies between two resource groups, ensure you have the same nodes for both resource groups. Resource group dependency using mixed replicated and non-replicated resource groups is not allowed.
- you can take into consideration serial acquisition/release order to prioritize processing of the resource groups.

1. Define the additional resources:
   - Service IP addresses for local network in Boston: odin_svc, thor_svc
   - Configure the server applications: app01_srv, app02_srv
2. Define the resource groups

Defining the site bounded resource groups. In Example 17-18 we show the configuration of the resource group thor_svc_rg, associated with the service IP address, thor_svc in site Boston.

Example 17-18 Adding the service IP addresses in the resource group

```
Change/Show All Resources and Attributes for a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Resource Group Name                                  thor_svc_rg
  Inter-site Management Policy                         ignore
  Participating Nodes from Primary Site                thor odin
  Participating Nodes from Secondary Site

  Startup Policy                          Online On Home Node Only
  Fallover Policy                         Fallover To Next Priority Node
  Fallback Policy                         Fallback To Higher Priority Node
  Fallback Timer Policy (empty is immediate)              []
  Service IP Labels/Addresses                        [thor_svc]
  Application Servers                                     []
  Volume Groups                                           []
  Use forced varyon of volume groups, if necessary    false
  Automatically Import Volume Groups                  false
  Filesystems (empty is ALL for VGs specified)            []
  Filesystems Consistency Check                       fsck
  Filesystems Recovery Method                         sequential
  Filesystems mounted before IP configured            false
  Filesystems/Directories to Export                       []
  Filesystems/Directories to NFS Mount                    []
  Network For NFS Mount                                   []
  Tape Resources                                          []
  Raw Disk PVIDs                                          []
  Fast Connect Services                                   []
  Communication Links                                     []
  Primary Workload Manager Class                          []
  Secondary Workload Manager Class                        []
  Miscellaneous Data                                      []
  GeoMirror Devices                                       []
```

```
F1=Help             F2=Refresh          F3=Cancel           F4=List
F5=Reset            F6=Command          F7=Edit             F8=Image
F9=Shell            F10=Exit            Enter=Do
```

Next, we define the geo resources. Example 17-19 shows the resource group definition for application APP01. Here we add the disk resources which include the geomirror devices defined across the sites.

Example 17-19 Defining the geo resource groups

```
        Change/Show All Resources and Attributes for a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                      [Entry Fields]
  Resource Group Name                                 app01_rg
  Inter-site Management Policy                        Prefer Primary Site
  Participating Nodes from Primary Site               odin thor
  Participating Nodes from Secondary Site             frigg

  Startup Policy                                      Online On Home Node Only
  Fallover Policy                  Fallover To Next Priority Node In The List
  Fallback Policy                Fallback To Higher Priority Node In The List
  Fallback Timer Policy (empty is immediate)          []
  Service IP Labels/Addresses                         [odin_svc]
  Application Servers                                 [app01_srv]
  Volume Groups                                       [vg01]
  Use forced varyon of volume groups, if necessary     false
  Automatically Import Volume Groups                   false
  Filesystems (empty is ALL for VGs specified)        [/app01]
  Filesystems Consistency Check                        fsck
  Filesystems Recovery Method                          sequential
  Filesystems mounted before IP configured             false
  Filesystems/Directories to Export                   []
  Filesystems/Directories to NFS Mount                []
  Network For NFS Mount                               []
  Tape Resources                                      []
  Raw Disk PVIDs                                      []
  Fast Connect Services                               []
  Communication Links                                 []
  Primary Workload Manager Class                      []
  Secondary Workload Manager Class                    []
  Miscellaneous Data                                  []
  GeoMirror Devices                                   [ulv11_loggmd ulv11gmd]


F1=Help                  F2=Refresh                  F3=Cancel    F4=List
F5=Reset                 F6=Command                  F7=Edit      F8=Image
```

```
F9=Shell                    F10=Exit                    Enter=Do
```

3. Synchronize the HACMP cluster

   Use smitty hacmp → Extended Configuration → Extended Verification and Synchronization.

4. Start the cluster services

   Use **smitty clstart**, to start the cluster services on a node.

After starting the cluster services, each node in the primary site Boston will acquire the resource groups according to their priority. The service IP addresses thor_svc and odin_svc will be activated on nodes thor and odin, respectively.

The resource groups app01_rg and app02_rg have a primary instance in site Boston, where the file systems are mounted, and a secondary instance in site Munchen. The geo mirror devices are activated in both sites, at the time cluster services are started, enabling the data written in the primary site Boston to be copied over the Geo_Primary networks to node frigg in site Munchen.

Example 17-20 shows the resource group status when the cluster services are running on all nodes.

Example 17-20 Normal status of the resource group

```
-----------------------------------------------------------------------------
Group Name     Type           State      Location
-----------------------------------------------------------------------------
thor_svc_rg      non-concurrent  ONLINE     thor
                                 OFFLINE    odin

app01_rg         non-concurrent  ONLINE     thor
                                 OFFLINE    odin
                                 ONLINE SEC frigg

odin_svc_rg      non-concurrent  OFFLINE    thor
                                 ONLINE     odin

app02_rg         non-concurrent OFFLINE     thor
                                ONLINE      odin
                                ONLINE SEC  frigg
```

Local failover in Boston

When one of the nodes in site Boston fails, the second node will acquire its resources, like in a normal fallover case. The primary instance of the replicated resource group will be reactivated on the surviving node (see Figure 17-2 on

page 655). The geo mirror devices are restarted during the recovery process, so the data replication between sites is resumed.

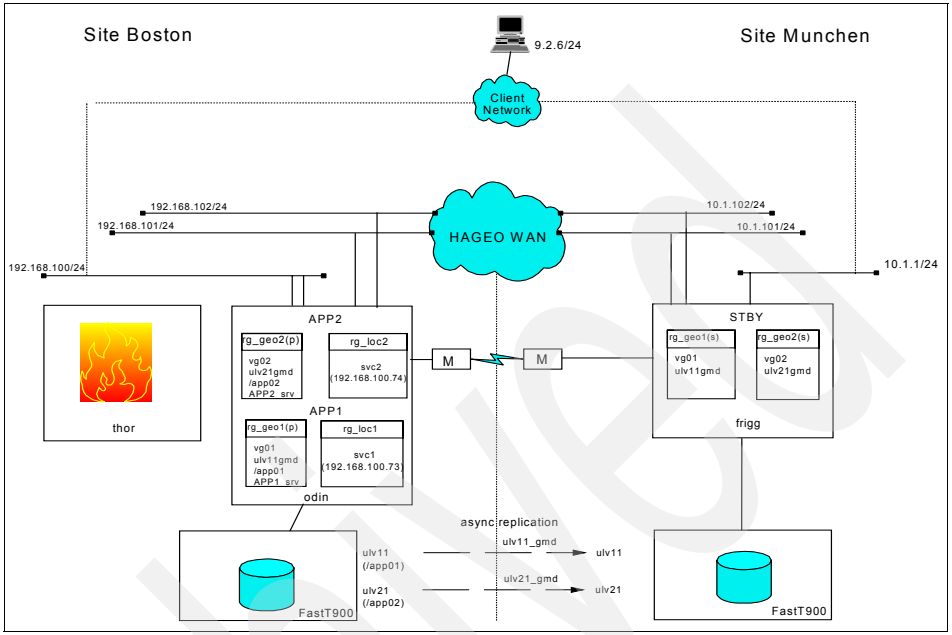

Figure 17-2 Resource relocation at node failure in the primary site

We test the local fallover in site Boston, by stopping the cluster services on node thor, using the graceful with takeover option. The resource groups: thor_svc_rg and app01_rg are activated to node odin.Example 17-21 shows the status of the resource groups after fallover.

Example 17-21 Resource Group status after fallover

```
-------------------------------------------------------------------------------
Group Name     Type           State      Location
-------------------------------------------------------------------------------
thor_svc_rg    non-concurrent  OFFLINE    thor
                               ONLINE     odin

app01_rg       non-concurrent  OFFLINE    thor
                               ONLINE     odin
                               ONLINE SEC frigg

odin_svc_rg    non-concurrent  OFFLINE    thor
                               ONLINE     odin

app02_rg       non-concurrent OFFLINE     thor
                              ONLINE      odin
```

Site failover/fallback

In case of a disaster at the primary site, both of the nodes from the primary site become unavailable. The cluster services on the remote node, checks the availability of the nodes in the primary site using all Geo_Primary and Geo_Secondary networks. In case of using the DBFS (Dial Back Fail Safe) system, the remote site calls the primary site to check if any node in the primary site is up. If there is no response from the primary site, the secondary site will acquire the resource groups. Figure 17-3 shows the resource groups relocation after a disaster at the primary site.

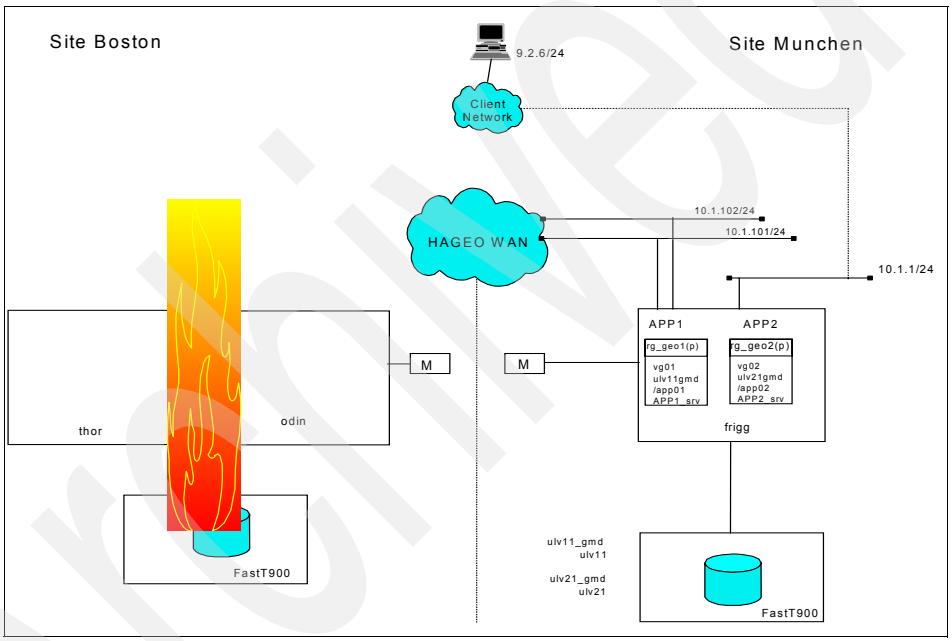

Figure 17-3 Site fallover

Note, that only the replicated resource groups are now active in site Munchen. The external client connects to the node frigg using the local network available in site Munchen. Since we use asynchronous geo mirror devices, at site fallover, the attributes of the geo devices changes to primary role, so they are able to process the write requests from the applications.

We simulate in our environment a site fallover by stopping the cluster services on both nodes odin and thor, one at a time. After site fallover to Munchen the status of the replicated resources is changed from “ONLINE SECONDARY” to “ONLINE” on node frigg (see Example 17-22 on page 657).

Example 17-22 Resource group status after site fallover

```
------------------------------------------------------------------------------
Group Name     Type           State      Location
------------------------------------------------------------------------------
thor_svc_rg      non-concurrent  OFFLINE    thor
                                 OFFLINE    odin

app01_rg         non-concurrent  OFFLINE    thor
                                 OFFLINE    odin
                                 ONLINE     frigg

odin_svc_rg      non-concurrent  OFFLINE    thor
                                 OFFLINE    odin

app02_rg         non-concurrent OFFLINE     thor
                                OFFLINE     odin
                                ONLINE      frigg
```

At the time the primary site reintegrates in the cluster, the Geo_Primary networks become up, and the data replication starts in reverse order, from the primary geo devices in site Munchen to the secondary geo devices in Boston (see Figure 17-4 on page 658).

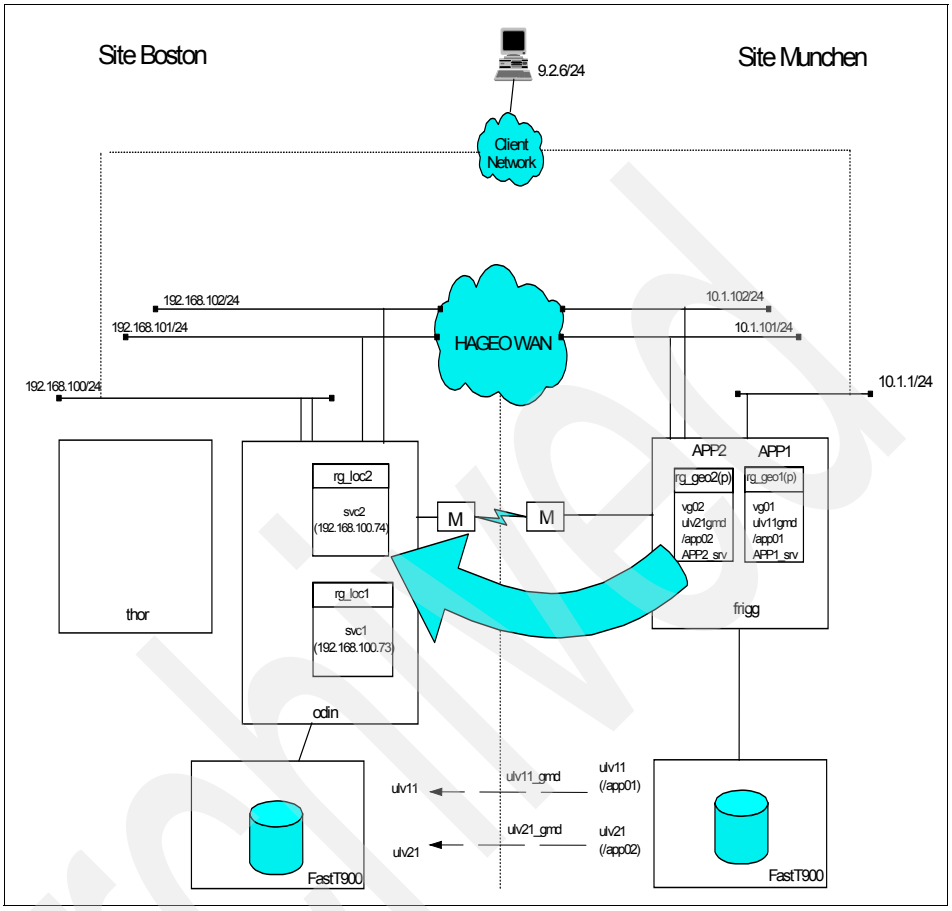

Figure 17-4 Site fallback

We use for our scenario the intra-site policy “Prefer Primary Site”. After the synchronization is completed, the resources are re-acquired at site Boston and the geo mirror devices’ attributes is reverted to the initial values: geo devices in site Boston have primary role and those from Munchen, secondary role. The file systems /app01 and /app02 are mounted, and both APP01 and APP02 are started in site Boston.

18

GLVM concepts and configuration

HACMP/XD Global Logical Volume Manager (GLVM) is a new high availability function that can mirror data across a standard IP network and provide automated fallover/fallback support for the applications using the geographically mirrored data.

GLVM performs the remote mirroring of AIX logical volumes using AIX's native Logical Volume Manager (LVM) functions for optimal performance and ease of configuration and maintenance.

This chapter examines geographic logical volume manager (GLVM): Concepts, installation and configuration, and migration.

18.1 HACMP/XD GLVM

GLVM provides similar functions as HAGEO, but using a simplified method to define and maintain the data replication between the sites. It is intended to be a long term replacement for HAGEO.

HACMP/XD GLVM provides two essential functions:

- Remote data mirroring
- Remote fallover and fallback

Together these functions provide high availability support for applications and data across a standard TCP/IP network to a remote site.

HACMP/XD for GLVM provides the following features for disaster recovery:

- Allows automatic detection and response to site and network failures in the geographic cluster without user intervention;
- Performs automatic site takeover and recovery and keeps mission-critical applications highly available through application fallover and monitoring.
- Allows for simplified configuration of volume groups, logical volumes and resource groups.
- Uses the TCP/IP network for remote mirroring over an unlimited distance
- Supports maximum sized logical volumes.

HACMP/XD:GLVM is HACMP extended distance using geographic logical volumes to mirror data to the remote site. HACMP/XD:GLVM:

- Supports clusters with multiple nodes at 2 sites
- Mirrors data by providing a local representation of the remote physical volume to the LVM.
- The local and remote storage systems don't have to be the same type of equipment.
- The aim of was to simplify the operation of HACMP/XD:GeoRM and let LVM control mirroring.
- Compared to HAGEO, there are shorter code paths, therefore GLVM is more efficient and robust.
- The code was simplified, therefore easier to support and use.
- The plan is that GLVM will eventually replace GMDs.

18.1.1 Definitions and concepts

- Remote physical volume (RPV)

  A pseudo device driver that provides access to the remote disks as though they were locally attached. The remote system must be connected via TCP/IP network and currently only runs with HACMP

  The distance between the sites is limited by the latency of the connecting network.

- The RPV consists of two parts:
  - RPV Client.

    This is a pseudo device driver that runs on the local machine and allows the AIX LVM to access remote physical volumes as though they were local. The RPC clients are seen as hdisk devices, which are logical representations of the remote physical volume.

    The RPV client device driver appears like an ordinary disk device driver, e.g., RPV client device, hdisk8, and will have all it's I/O directed to the remote RPV Server. It also has no knowledge at all about the nodes, networks etc.

    In HACMP/XD 5.3,concurrent access is not supported for GLVM, so when accessing the RPV clients, the local equivalent RPV servers and remote RPV clients must be in a defined state.

    When configuring the RPV client, the following is defined:

    - The address of the RPV server
    - The local address (defines the network to use)
    - The time-out. This field is primarily for the standalone GLVM option, as HACMP will overwrite this field with the cluster's config_too_long time. In an HACMP cluster, this will be the worst case scenario, as HACMP will detect problems with the remote node well before then.

    There is a smit menu to configure the RPV clients, smitty rpvclient
  - RPV Server

    The RPV Server runs on the remote machine, one for each physical volume that is being replicated. The RPV Server can listen to a number of remote RPV Clients on different hosts to handle fallover.

    The RPV Server is an instance of the kernel extension of the RPV device driver with names such as rpserver0 and is not an actual pseudo device.

    When configuring the RPV server, the following is defined:

    - The PVID of the local physical volume

- The addresses of the RPV clients (comma separated).

– Geographically mirrored volume group (GMVG)

A volume group that consists of local and remote physical volumes. Strict rules are applied to GMVGs to ensure that it is much less likely to find that you do not have a complete copy of the mirror at each site. For this reason the superstrict allocation policy is required for each logical volume in a GMVG. HACMP/XD will also expect each logical volume in a GMVG to be mirrored.

GMVGs are managed by HACMP/XD and recognized as a separate class of replicated resources, so have their own events. HACMP/XD verification will also issue a warning if there are resource groups that contain GMLV resources that do not have the forced varyon flag set and if quorum is not disabled.

There is a smit menu to configure the RPV servers, smitty rpvserver.

Important: HACMP/XD will insist that each physical volume that is part of a volume group with RPV Clients has the reverse relationship defined. This as a minimum every GMVG will consist of two physical volumes on each site - one local disk and the logical representation of the remote physical volume.

► GLVM Utilities

There are smit menus provided with GLVM to create the GMVGs and the logical volumes. Whilst not required, as they perform the same function as the equivalent smit menus under the covers, they do control the location of the logical volumes to ensure proper placement of mirror copies.

If you use the standard commands to configure your GMVGs, it is recommended to use the GLVM verification utility.

Important: The LVM commands and SMIT menus are not completely aware of RPV design, so it is possible to create geographic mirrored volume groups that do not have a complete copy of the data on either site

► New networks definitions are added to HACMP/XD for GLVM

XD_data Network that can be used for data replication only. This is the equivalent of the Geo_Primary network. This network will support adapter swap, but not fallover to another node. RSCT heartbeat packets will be sent on this network.

Note: HACMP/XD 5.3 only supports 1 XD_data network. Etherchannel is supported.

XD_ip — Network that can be used for RSCT heartbeat via IP. Typically would be low bandwidth and just used to prevent cluster partitioning. Same as the ethernet type network, except the heartbeat parameters have been modified for the greater distance. Does not support IPAT and cannot be used for data mirroring.

XD_rs232 — Network that can be used for serial communications. Same as the RS232 network type, except that the heartbeat parameters have been modified for the greater distance. Similar to the Geo_Secondary network for HACMP/XD:HAGeo. For example could be a leased line or a serial line with line driver.

RSCT heartbeat packets will be sent over all the networks.

Note: HACMP/XD:GLVM requires one XD_data network for data replication and one of XD_rs232 or XD_ip to differentiate between a remote node failure or XD_date network failure.

Figure 18-1 on page 664 shows an example of two sites with one node at each. Viewing the replication from Node1, we see that the destination physical volume hdisk3 on Node2 are presented on Node1 as hdisk8.

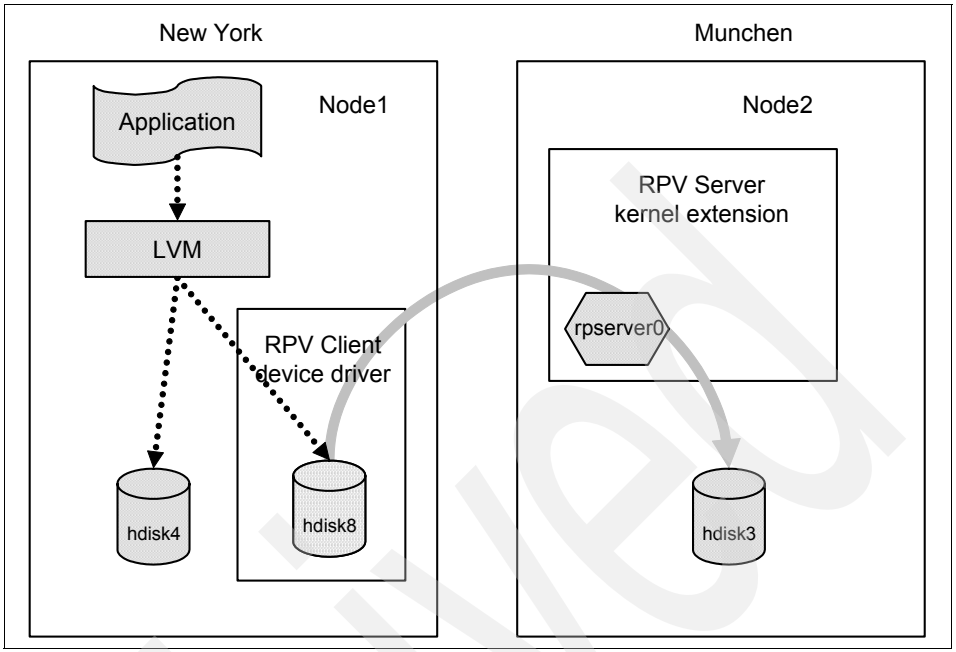

Figure 18-1 RPV client viewed from Node1

On Node2 there is an RPV Server for each physical volume. On Node1 there is a corresponding RPV Client for each RPV server, which presents to the LVM as a local physical volume. We can now construct a volume group glvm_vg on Node1, mirroring a local physical volume (hdisk4) to the local RPV Client (hdisk8). The RPV client and server will ensure that all I/O is transferred over the XD data network to the physical volumes on Node2.

When Node2 becomes the active node, the operation is reversed as can be seen in Figure 18-2 on page 665. The remote physical volumes on Node1 are presented through the RPV client as local physical volumes on Node2.

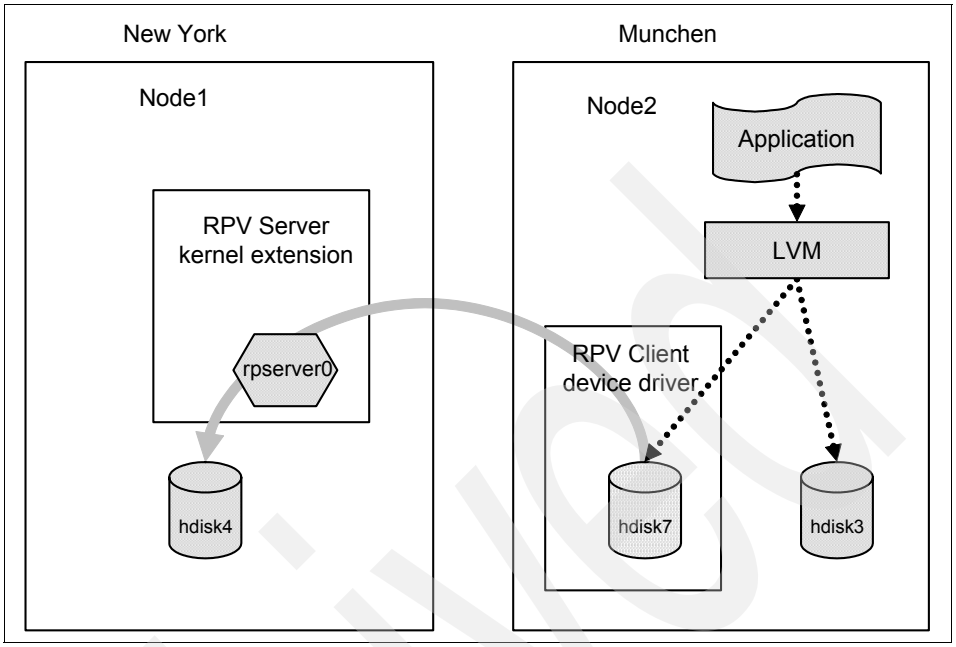

Figure 18-2 Reverse configuration on fallover to remote site

Note: As with physical volumes the shared between multiple systems, the hdisk numbering may not be consistent, the PVID however will be.

Figure 18-3 on page 666 shows the configuration of both nodes with the RPV servers and clients defined. A volume group glvm_vg is defined on both nodes, consisting of two local physical volumes, and the two local representations of the remote node's physical volumes.

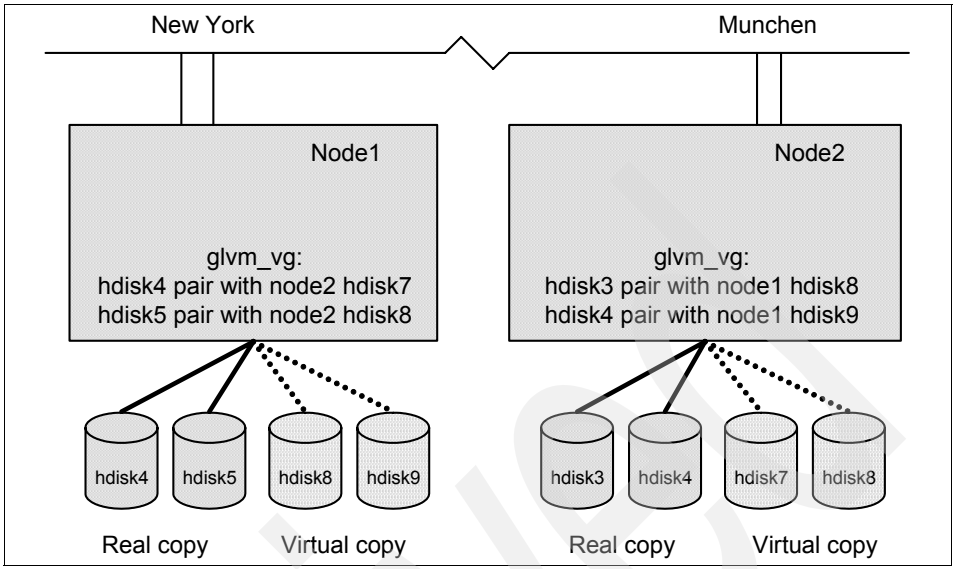

Figure 18-3 RPV client and server configuration on both sites

Configuring GLVM under HACMP control

The RPV clients, RPV servers and GLVG must be configured on all nodes before can be defined as part of HACMP resource group. There must be a RPV Server for every local disk and local RPV client for every remote disks that belongs to a GMVG.

Table 18-1 shows the labels for the configuration in Figure 18-3.

Table 18-1 RPV names used in our environment

	Local disks	RPV Servers	RPV Clients
NYC / Node1	hdisk4	rpvserver0	hdisk7 on Munchen
	hdisk5	rpvserver1	hdisk8 on Munchen
Munchen/ Node2	hdisk3	rpvserver0	hdisk8 on NYC
	hdisk4	rpvserver1	hdisk9 on NYC

It is possible to access the GMVG from either side as long as the appropriate clients and servers are available. That is, for the GMVG to be configured on Node1, then the RPV clients must be available on Node1 and the RPV servers available on the remote client.

GMVGs cannot be configured through CSPOC - so create on one node, then varyoff and import on the next node. Repeat till defined on all nodes in the cluster.

Once the GMVGs have been configured, they just need to be added to the resource group, as HACMP/XD will recognize them correctly and call the correct events for their processing.

General recommendations

It is recommended that the quorum be turned off for GMVGs. Although leaving the quorum on will allow for further checking, it will mean that HACMP/XD will attempt to fallover the resource group if half the disks are lost - if the remote site is down. This is often not the preferred behavior.

It is recommended that the forced varyon flag be set in the resource group attributes. This is so either site can bring the resource group online if the remote site is unavailable. This would have the potential of using stale data if HACMP/XD didn't protect against this scenario by using a "may be stale" flag.

Using stale data is possible in the following scenario

If production is active on primary site, but the XD_data network is down. Thus the data in the physical volumes on the primary site is more up to date then the backup site's data. The primary site is aware that the backup site is operational and that the data is not being replicated, so the GMVGs on the remote site are marked "may be stale".

If the application were to fall over to the backup site without the "may be stale" flag, the GMVG would be activated (forced on) and the stale data would be used when the application starts. The "may be stale" flag halts the activation of the volume group at this point to allow the administrator to make as decision as what to do.

Starting the cluster

After the cluster has been configured and synchronized, start HACMP cluster services on the primary node. Assuming that the remote server is not be available, the local RPV clients will not be accessible, so HACMP will have to force the varyon of the volume group and all I/O to the local physical volumes, will result in stale partitions on the RPV client volumes as the local physical partitions are modified. See Figure 18-4 on page 668.

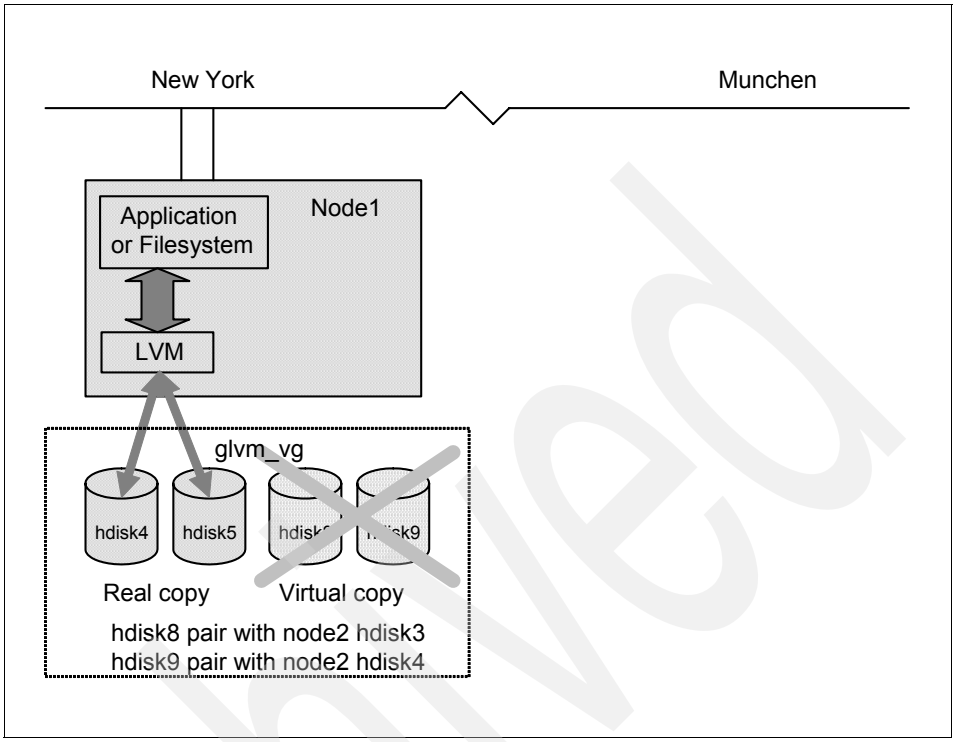

Figure 18-4 Node active at primary site with backup down.

When the node at the remote site becomes active, HACMP will activate the RPV Servers on that node and start the RPV clients on the primary site. HACMP/XD will run an event which will inform the LVM that the RPV clients are available, so that the stale partitions on the RPV clients are refreshed by copying data from the local physical volumes to the remote physical volumes. See Figure 18-5 on page 669.

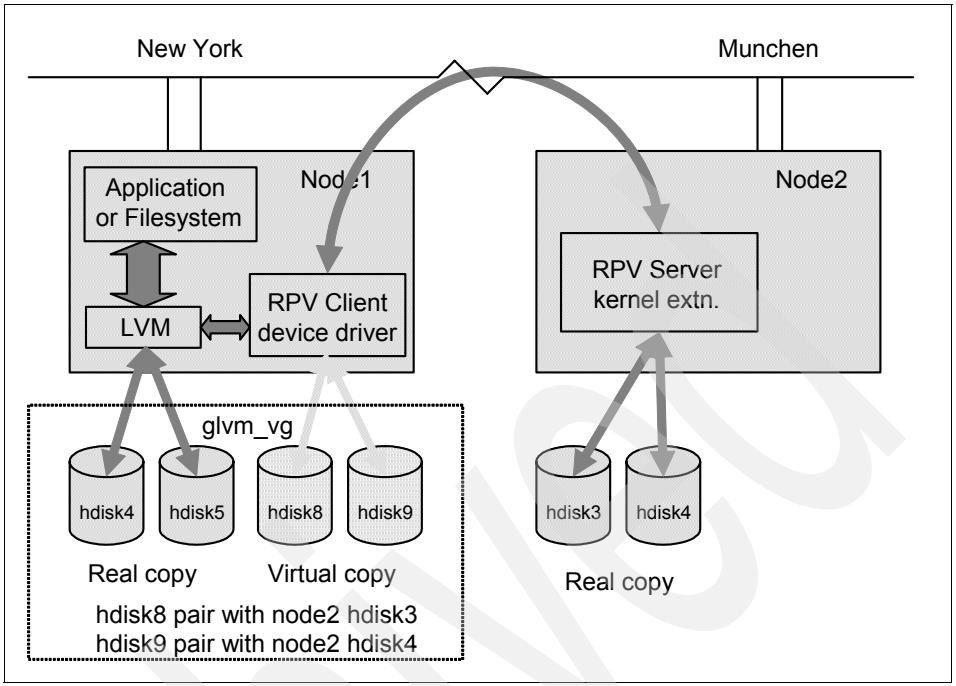

Figure 18-5 Backup site active and data replicating

If the application falls over to the backup site, or the primary site is taken off line, the reverse process will occur. Those physical volumes on the remote node, that were controlled by the RPV servers, are now the local physical volumes for the volume group, while the RPV clients (which point to the physical volumes on the primary site) will be offline until the primary node becomes available. The volume group will again have to be activated in forced mode. See Figure 18-6 on page 670.

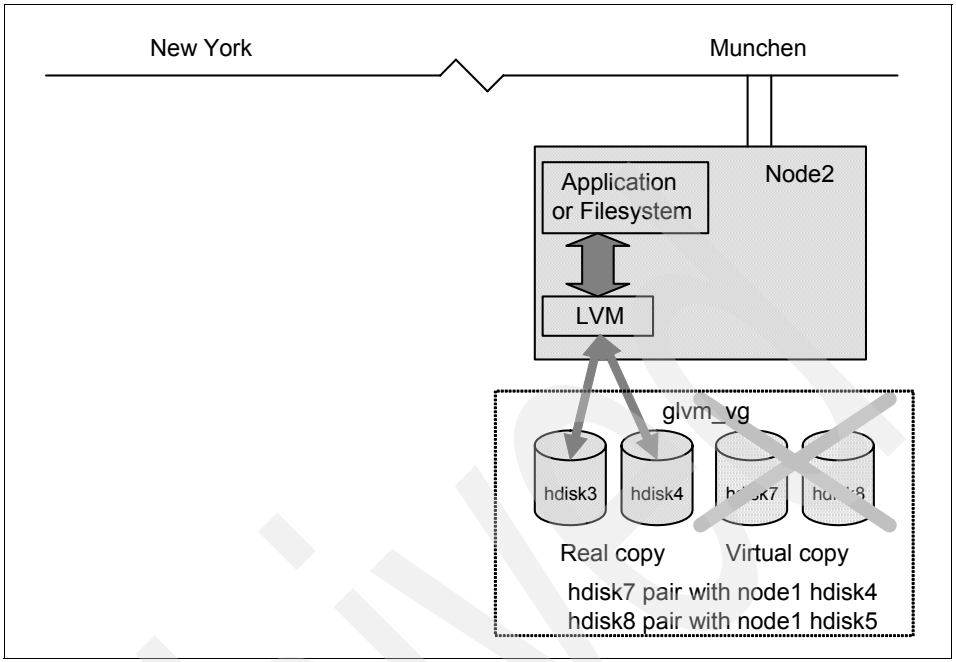

Figure 18-6 Backup node active with primary site down

When the primary site becomes active, HACMP/XD will either leave the resource group on the backup site, and the RPV servers there will start, bringing the backup servers RPV clients online. The LVM again will process the replication of the data on the backup's physical volumes that are marked stale, synchronizing through the RPVs to the primary node's physical volumes. See Figure 18-7 on page 671.

However if the resource group is configured to prefer the primary site, processing will stop on the backup site and the resources brought on line on production. This means that the application will be running on the primary site while the stale data is being synchronized back from the backup site. Any attempt by the application to read a stale partition will result in a read from the RPV server. See Figure 18-5 on page 669.

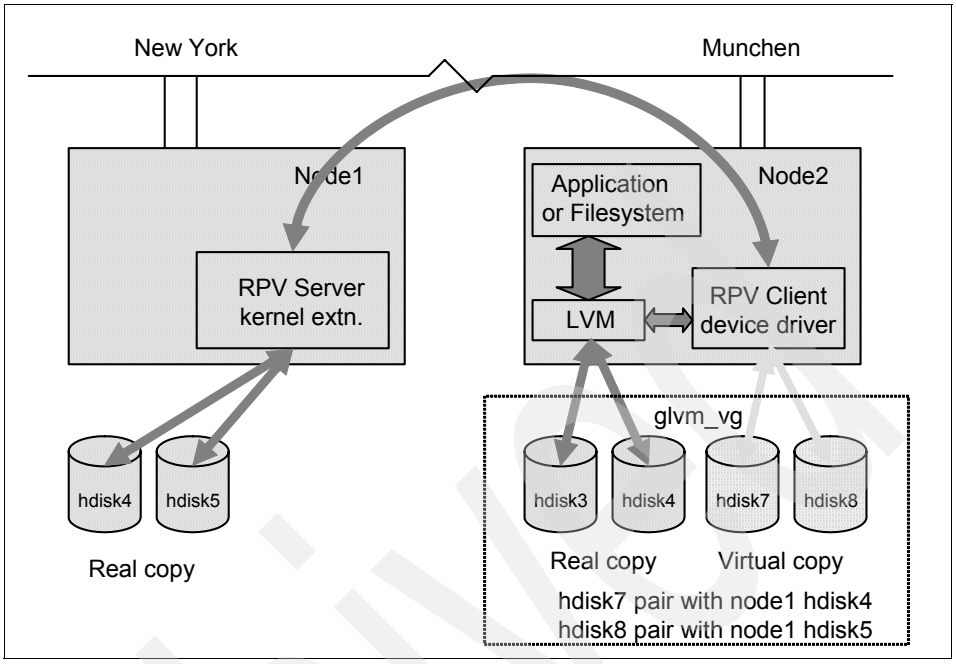

Figure 18-7 Primary node integrated into the cluster and replication started.

This question of site preference for the resource group can have important performance repercussions, as the I/O is handled by the LVM and it is largely unaware that the underlying device is a remote physical volume.

For example if we configure the local GMVG to be mirrored across two physical volumes and one RPV client (See Figure 18-8 on page 672) we increase the changes of the data being available locally.

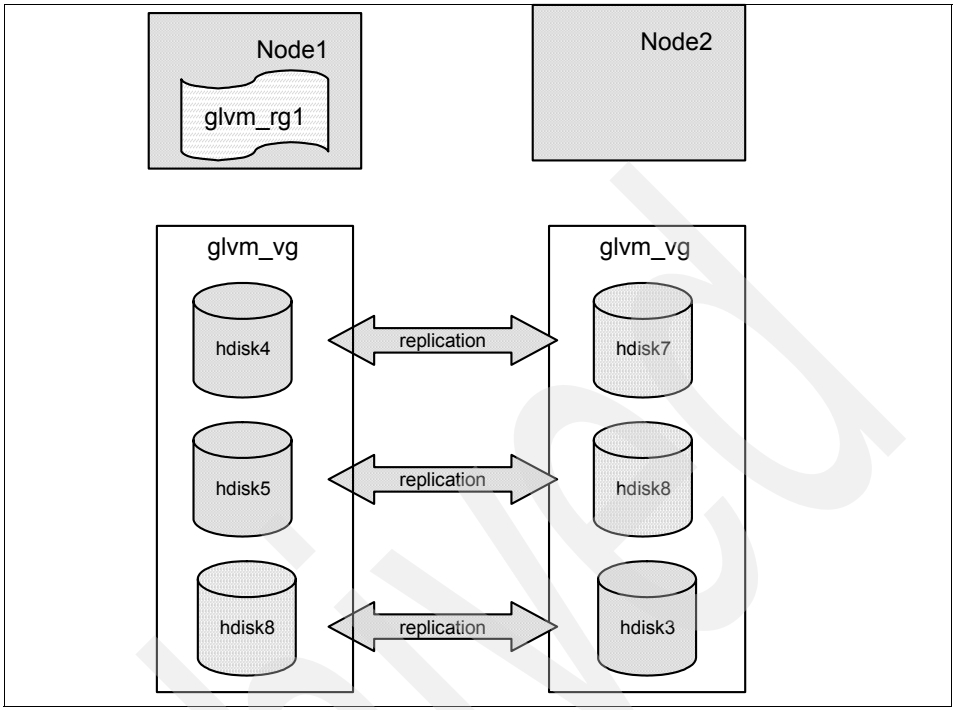

Figure 18-8 Application active on primary with 2 PV and one RPV client

However, if the primary site goes offline, it means that the backup site will be running on a volume group consisting of one local physical volume, and two RPV clients. See Figure 18-9 on page 673.

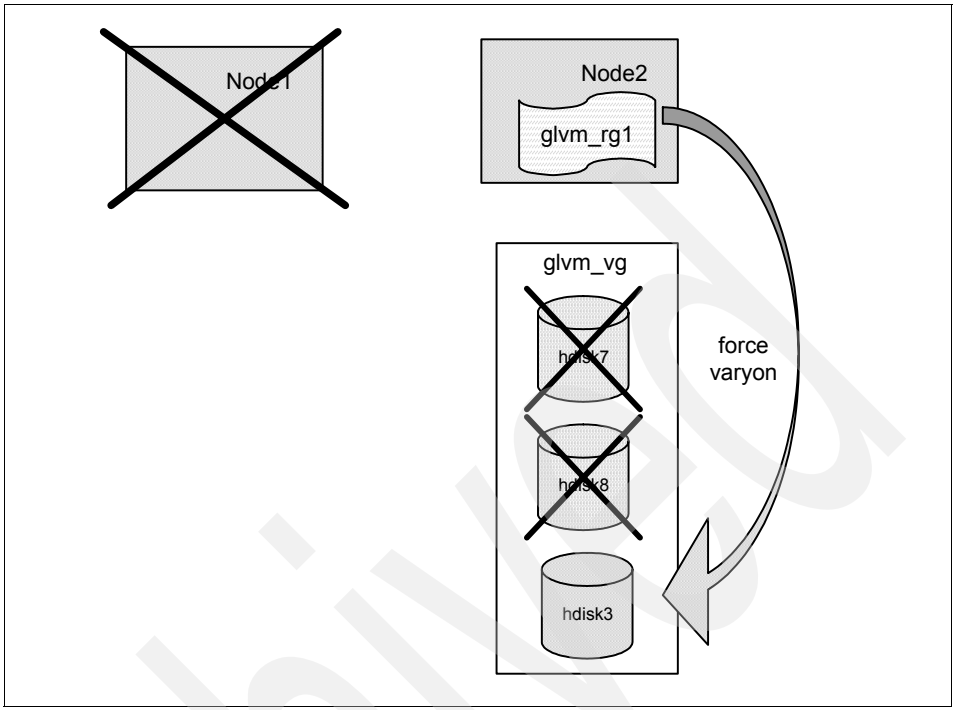

Figure 18-9 Primary site down and application using 1 local copy of VG

In this scenario, the site preference of the resource group has a major influence on the performance of the re-synchronization of the data.

If the resource group has no site preference, it will stay online on the secondary site and two copies of the stale data will be sent over the XD_data network - one for each of the RPV servers. See Figure 18-10 on page 674.

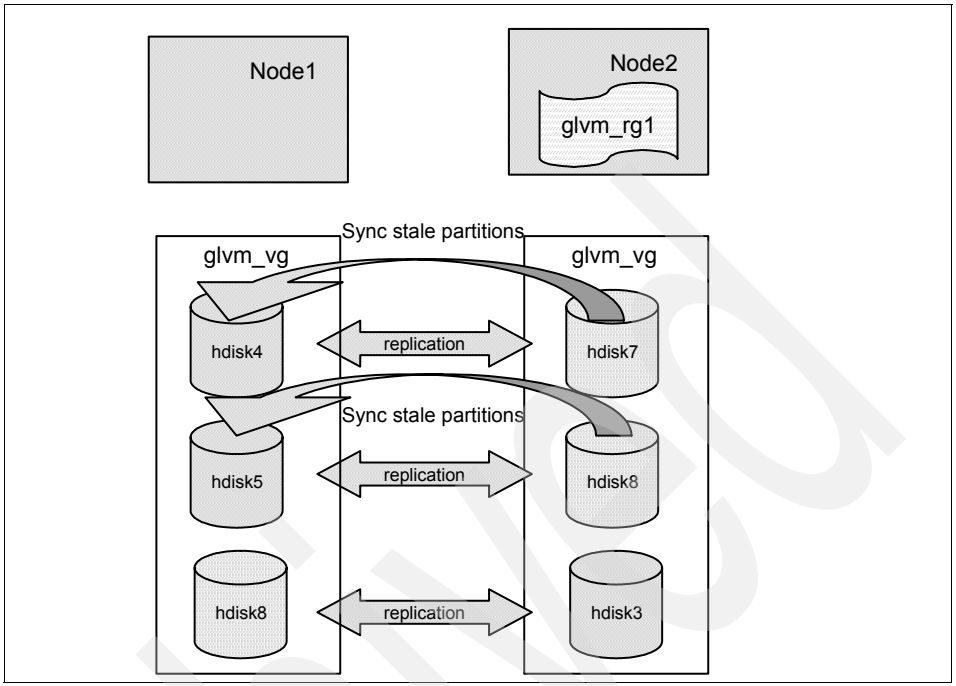

Figure 18-10 Application didn't fallover on integration of primary site

However if the site preference for the resource group is to fallback to the primary site, then the volume group will be activated on the primary site, with 1 physical volume (the RPV client) up to date, and the two local physical volumes with stale partitions. Thus there will only be one copy of each partition sent over the network to bring the physical volumes into a synchronized state. Reads from the stale partitions will take longer as they will include the network latency, as they must be made from the remote physical volumes. See Figure 18-11 on page 675.

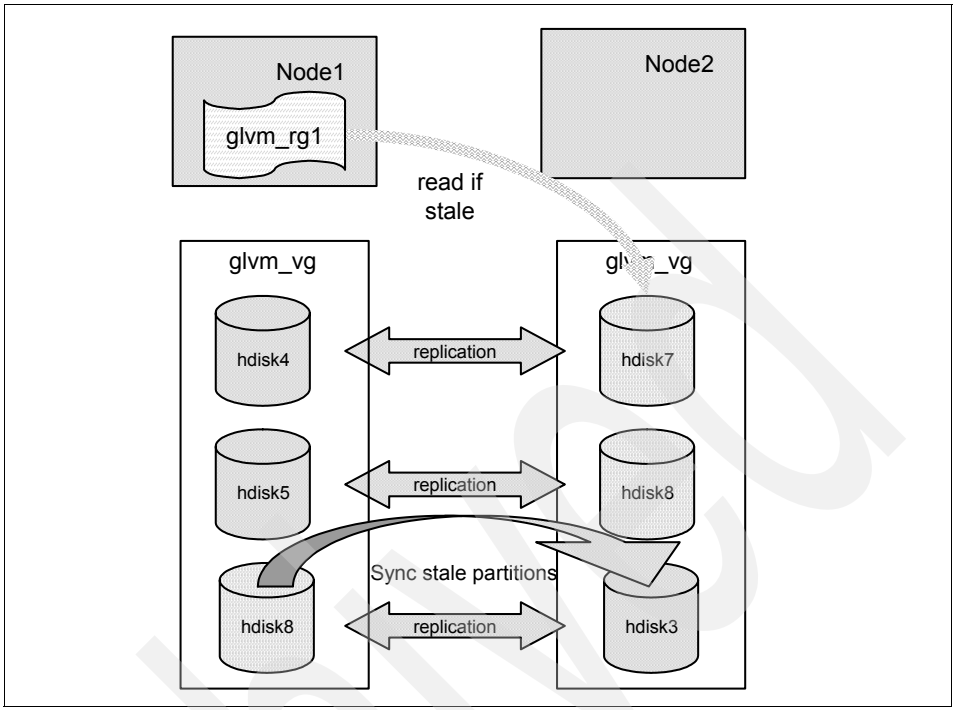

Figure 18-11 Application falls over on primary site integration

The I/O path

For normal I/O, the application will pass an I/O request to the LVM, which will pass the request through the disk device driver, through the adapter device driver and then to the physical volume.

With GLVM, the application will pass the I/O request to the LVM, which will pass the request to the RPV client device driver, which will send the request over the TCP/IP network to the remote RPV server. The remote RPV server will pass the request through the disk device driver on the remote node, through the adapter device driver to the remote physical volume. The response will then return the same way.

Any delay in the network will lead to slower I/O performance, while any error will be returned to the LVM as a physical volume device driver would.

The LVM will see the RPV client as a slower and less reliable physical volume - slower because of the longer I/O path (particularly network latency) and less reliable as long distance networks through multiple devices have a greater failure rate than local writes.

18.2 Migration, the logic for going HAGeo to GLVM

There is no automatic migration from HACMP/XD:HAGeo to HACMP/XD:GLVM, but both the HAGeo and GLVM versions can coexist on the same cluster. So a step by step migration of the Geomirrored resources can be performed with some downtime.

As HACMP/XD:HAGeo doesn't support dynamic reconfiguration, the whole cluster will have to be stopped for the topology and resource change. However the migration of the data from a GMD to a GLVM can be done with the application live. This migration does require the re-mirroring of all the data to the remote site.

Important: Careful planning is required as the full replication of each data logical volume will both effect the primary and secondary nodes performance as well as use a large proportion of the network bandwidth. The size of the logical volumes and network bandwidth will determine the time required, and this should be scheduled as far as possible during a quite period.

For our example migration, we will look at a cluster with two nodes and two applications at the primary site, and a single node at the backup. The steps are:

- Install GLVM filesets
- Stop remote site and create RPV servers on the remote site
- Create local rpv client and mirror
- Mirror the local data logical volumes
- Create local rpvservers and remote rpvclients
- Modify /etc/filesystems to point to LVs not GMDs (if using file systems)
- Stop cluster and modify topology and resource group definitions
- Verify and synchronize the cluster
- Start cluster

Figure 18-12 on page 677 shows the cluster running HACMP/XD:HAGeo. Node thor and odin are at site Boston, running one application each, with each application using one Geomirrored fleshiest (both the underlying logical volume and jfslog are mirrored), which is replicated to frigg at site Munchen.

Note: The steps described are assuming limited hardware, so the backup copy of the data will not be available for the entire migration period.

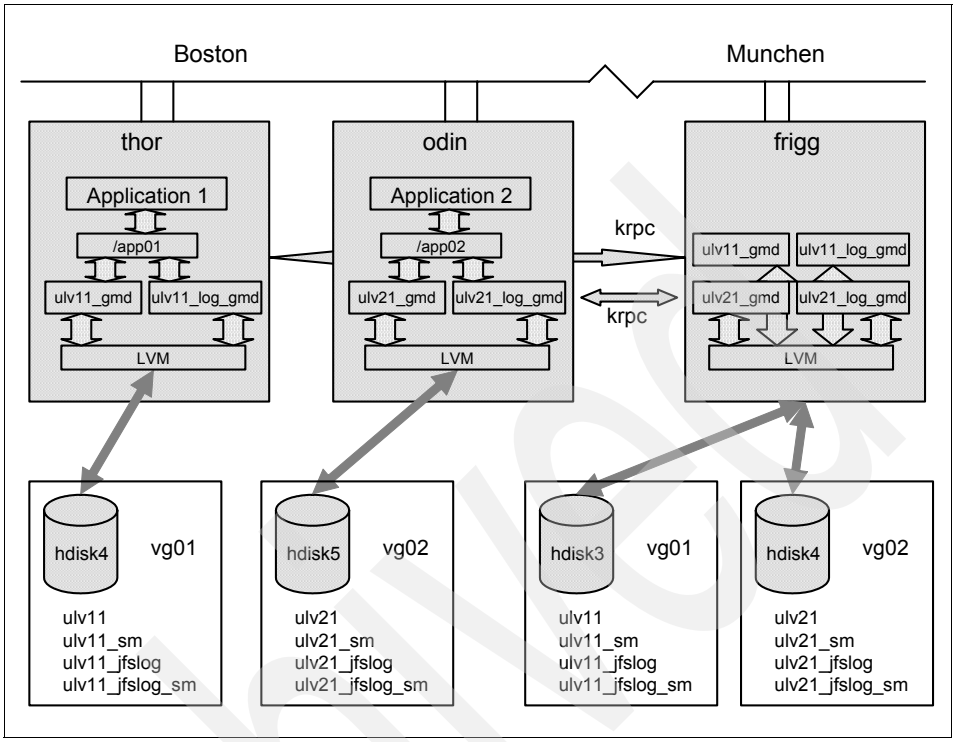

Figure 18-12 Example HACMP/XD cluster for migration to GLVM

For this exercise we will only look in detail at the migration of the resource group on node thor as the logic is the same for all nodes. Figure 18-13 on page 678 shows the details of the single replicated resource.

18.2.1 Install GLVM filesets and configure GLVM

Install the GLVM filesets on each node in the cluster. Take a HACMP and HAGeo snapshot.

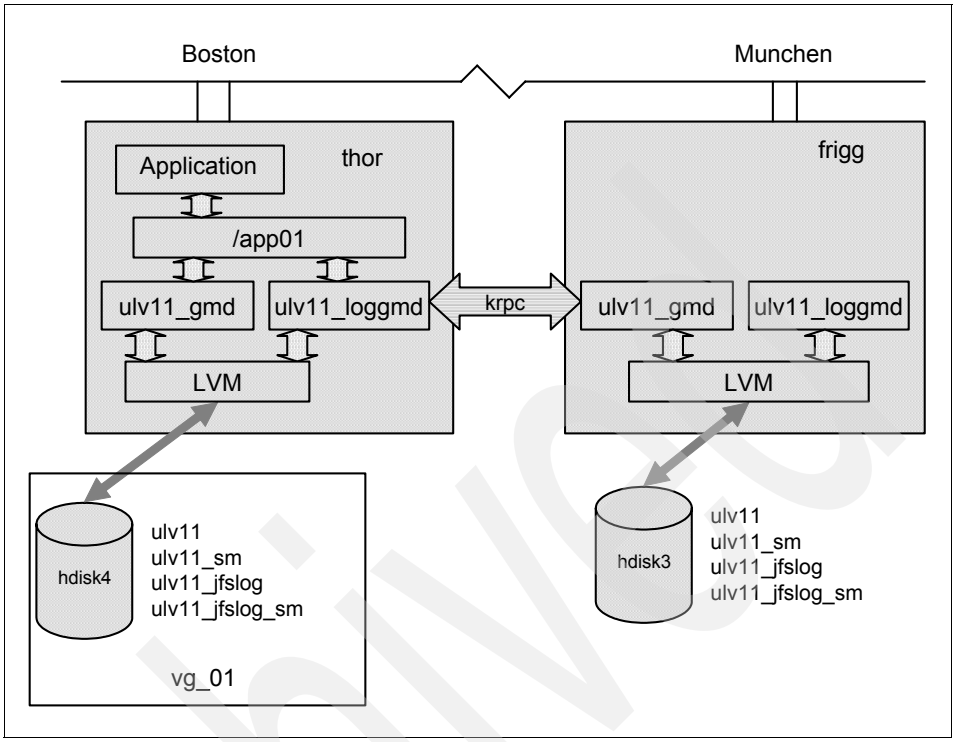

Figure 18-13 Example with one GMD resource

1. Stop remote site and create RPV servers on the remote site

   The aim of this migration is to move the data replication to use GLVM, without requiring extra hardware. The limitation of this process is that the remote copy of the GMDs will have to be turned off and then be overwritten as a GLVM device - leaving the customer without a backup copy of the data. If this is an issue, then extra hardware will be required.

   In our example, the replicated copy of the resource group on the backup site is stopped (the primary GMD no longer synchronizing data) and an RPV server created pointing to the physical volume. See Figure 18-14 on page 679.

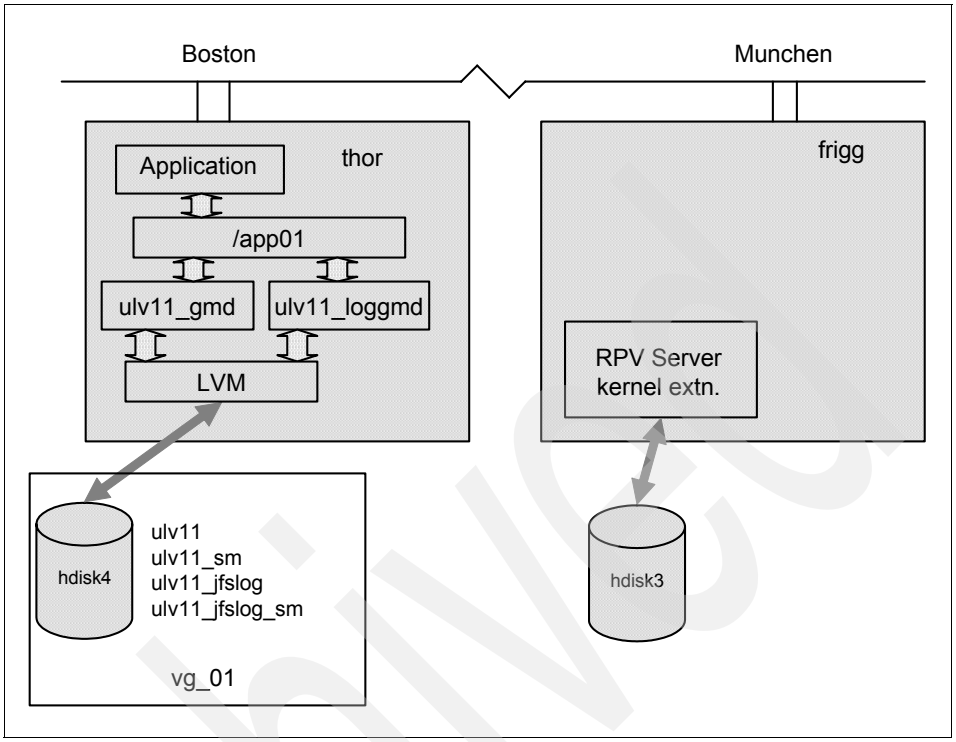

Figure 18-14 RPV Server created

2. Create local rpv client and mirror

   The next step is to create the local RPV client and add the physical volume to the GMD volume group. Each of the data and jfslog logical volumes needs to be changed to superstrict allocation policy and then extended by adding a copy on the RPV client.

   We recommend that the statemap logical volumes don't need to be mirrored, as they will no longer be required after the GMDs are turned off.

   Once the logical volumes have had the mirror copy defined, the data can then be synchronized. This will replicate all the data from the primary copy of the data to the backup site - effecting the performance of both nodes and placing a large load on the network. See Figure 18-15 on page 680.

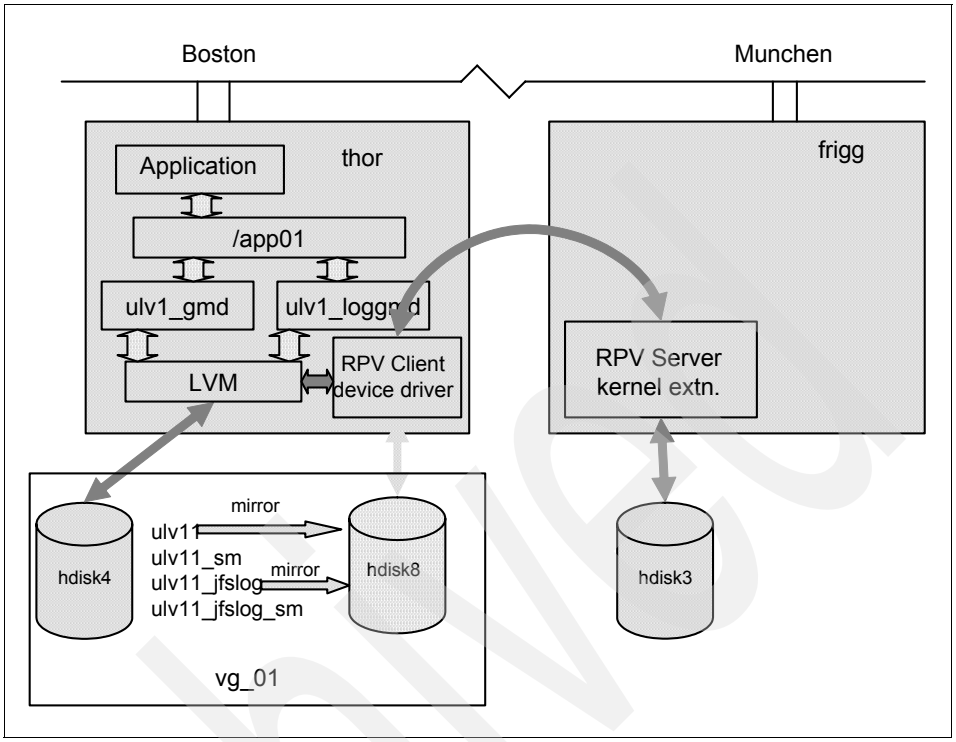

Figure 18-15 Mirror data LVs to RPV client

3. Create local rpvservers and remote rpvclients

   Although not required until the application falls over, the RPV servers and clients must be created on all nodes or HACMP will fail verification.

4. Modify /etc/filesystems to point to LVs not GMDs

   File systems will now point to the logical volume, not the GMD, so changes need to be made to /etc/filesystems on each node. Any statemap or mirrored logical volume should be removed from the GMVG, as HACMP/XD will return an error if there are unreplicated logical volumes on any physical volume in a GMVG.

5. Stop cluster and modify topology and resource group definitions

   As HACMP/XD:HAGeo doesn't support dynamic reconfiguration, the cluster must be stopped on all nodes, so that:

   – The XD_Data network can be configured
   – The GMD definitions removed from the resource group
   – The GLVM volume groups' force varyon set true

6. Verify and synchronize the cluster

   The GLVM changes need to be verified and synchronized to each node in the cluster.

7. Start cluster

   The cluster can now be started, with the modified resource group using the GLVM devices (See Figure 18-16), and the remaining resource groups using GMDs. The unused GMD definitions can now be removed.

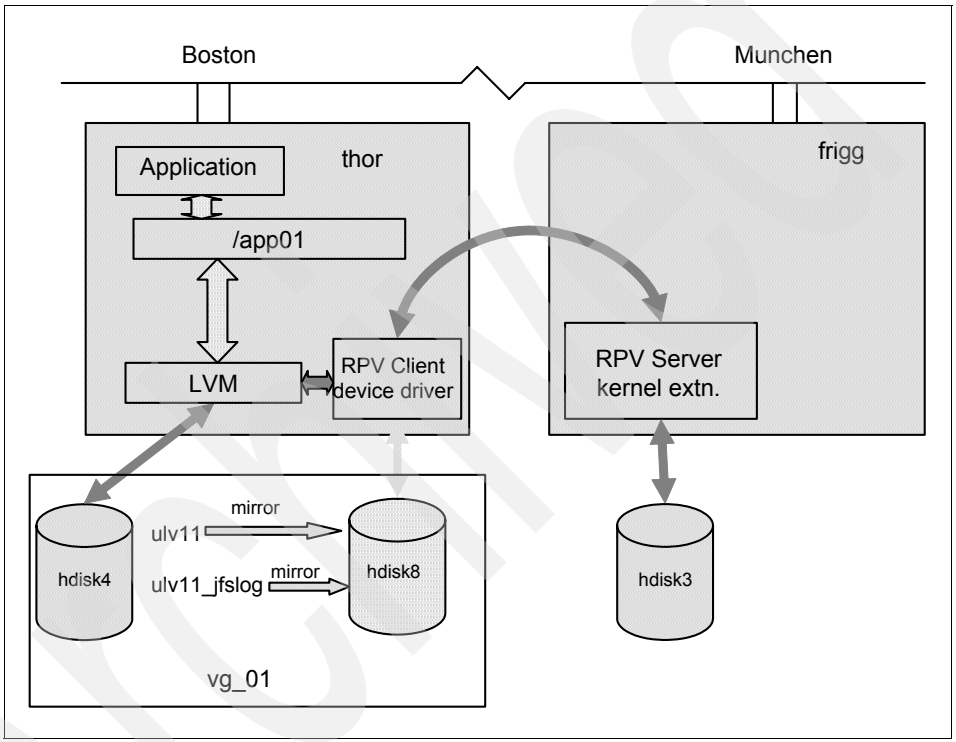

Figure 18-16 Application now using GLVM devices.

One option to consider is stopping only the resource group that contains the GMDs to be replaced, do a forced down of the cluster on all nodes, make the changes to the topology and resource group configurations, then restart the cluster. This will mean that the only outage experienced will be for the resource group that we are actually modifying.

18.2.2 Performance considerations

While this is not the scope of this redbook, the following should be considered in planning your GMVG configuration.

- Mirror write consistency

  The mirror read write consistency can be turned off to improve write performance, but on rebooting after a crash, a **`syncvg -f`** must be run before the logical volume can be accessed. Or the LV can be set :

  - Active

    Is the default for a mirrored logical volume

    Ensures a fast recovery after a system crash (no need to do the syncvg -f on reboot). This could lead to a performance problem on writes

  - Passive

    No performance penalty on writes, and will not require a syncvg -f after reboot. This will do a background resynchronization of all partitions if it is detected that the system was not shutdown gracefully

- LVM scheduling policies

  There are four read / write scheduling policies defined for mirrored logical volumes:

  - Parallel

    Reads will be balanced across the physical volumes (sent to the device with the shortest queue), writes will be sent to each physical volume in parallel (i.e., at the same time).

  - Sequential

    Reads will be from the primary copy and writes will be done in sequence (i.e., one copy after another).

  - Parallel write, sequential read

    Reads will be done from the primary copy and writes will be sent to all physical volumes in parallel.

  - Parallel write, round-robin read

    Reads will be from each copy in turn and writes will be sent to all physical volumes in parallel.

- Write verify

  There are two options:

  - Yes

    All writes to the logical volume will be followed by a read

  - No

    Writes not verified.

For GMVG's

- Mirror write consistency

  We recommend that the mirror write consistency be left active as a crash of the node will result in the synchronization of the whole logical volume. However if the network bandwidth and logical volume sizes can handle this then the passive mode could be considered.

- LVM scheduling policies

  The default parallel policy is recommended as the LVM developers have made a small change for GMVGs. The change is that the LVM will attempt to read from a local copy if the physical volumes are available, in preference to reading from the RPV.

- Write verify

  Would strongly recommend leaving off, which is the default.

18.2.3 Troubleshooting

- Unlike HAGEO, there is very little data in syslog, - one trace hook (4A6).
- HACMP snapshot contains the **`lsrpvserver -H and lsrpvclient -H`** output in the .info file
- snap -g Contains the RPV server and client configurations
- general.snap - filesets; attributes for rpvserver and rpvclients
- CuAt - contains information about the remote site name

Example 18-1 shows RPV server properties:

Example 18-1 Check RPV server characteristics

```
frigg:/# lsattr -El rpvserver0
auto_online n                                 Configure at System Boot   True
client_addr 192.168.101.74                    Client IP Address          True
client_addr 192.168.101.73                    Client IP Address          True
rpvs_pvid   0022be2aa13f292e0000000000000000 Physical Volume Identifier True
frigg:/# lsattr -El hdisk7
io_timeout  180                               I/O Timeout Interval       True
local_addr  10.1.101.192                      Local IP Address           True
pvid        0022be2aa13dc0720000000000000000 Physical Volume Identifier True
server_addr none                              Server IP Address          True
```

Also, to check the RPV error information, use (see Example 18-2 on page 684):

Example 18-2 RPV error sample

```
odin:/# lsrpvserver -H
# RPV Server     Physical Volume Identifier       Physical Volume
# -------------------------------------------------------------
  rpvserver0     0022be2aa13dc072                 hdisk2
odin:/# lsrpvclient -H
# RPV Client     Physical Volume Identifier       Remote Site
# -------------------------------------------------------------
  hdisk6         0022be2aa13f292e                 Munchen

LABEL:          RPVC_IO_TIMEOUT
IDENTIFIER:     D034B795

Date/Time:       Thu Jul 14 15:48:03 2005
Sequence Number: 16314
Machine Id:      002574004C00
Node Id:         frigg
Class:           U
Type:            PERM
Resource Name:   hdisk7
Resource Class:  disk
Resource Type:   rpvclient
Location:
VPD:

Description
No response from RPV server within I/O timeout interval.

Probable Causes
RPV server is down or not reachable.

Failure Causes
There is a problem with the data mirroring network.
The node or site which hosts the RPV server is down.
RPV server is not configured in the Available state.

        Recommended Actions
        Correct the problem which has caused the RPV server to be down or
        not reachable.  Then, tell the RPV client to resume communication
        with the RPV server by running the command:
                chdev -l <device> -a resume=yes
        where <device> is the name of this RPV client device.

---------------------------------------------------------------------------
```

18.3 Steps for migrating from HAGEO to GLVM

Installing the package for GLVM. Select the following packages from the installation media:

- cluster.doc.en_US.glvm.html
- cluster.doc.en_US.glvm.pdf
- cluster.xd.glvm
- glvm.rpv.client
- glvm.rpv.server
- glvm.rpv.util

1. We begin with a graceful stop of the cluster services on frigg. This will stop the geo mirror devices at the remote site Munchen:

   ```
   smitty clstop
   ```

   Wait for the cluster services to be stopped on the remote node. The geo devices will be in the “Defined” state.

   Export the GMD volume group definition on node frigg:

   ```
   exportvg vg01
   ```

   This operation removes the volume group definition from the ODM and deletes the file systems’ stanzas from /etc/filesystems.

Configure the RPV Server environment. Perform the following steps from the RPV server:

2. Setup the remote mirroring site name. On the node frigg, run **smitty rpvserver->Remote Physical Volume Server Site Name Configuration-> Define / Change / Show Remote Physical Volume Server Site Name.** Define the name of the site as in HACMP definition of site.

   You can use the **rpvsitename** command to define the site:

   ```
   /usr/sbin/rpvsitename -a 'Munchen'
   ```

3. From the “Remote Physical Volume Servers” menu, choose Add Remote Physical Volume Servers to define the RPV servers which are associated with the target disks for mirroring. After selecting the target disks, specify the IP address of the RPV client, as in Example 18-3:

Example 18-3 Adding a RPV server

```
                    Add Remote Physical Volume Servers

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Physical Volume Identifiers                        0022be2aa13f292e
```

```
* Remote PV Client Internet Address     [192.168.101.73,192.168.101.74]
+
  Configure Automatically at System Restart?         [no] +
  Start New Devices Immediately?                     [yes] +



F1=Help              F2=Refresh           F3=Cancel            F4=List
F5=Reset             F6=Command           F7=Edit              F8=Image
F9=Shell             F10=Exit             Enter=Do
```

If using the command line, use **mkdev** command as in Example 18-4:

Example 18-4 Adding a RPV server - using CLI

```
frigg:/# /usr/sbin/mkdev -c rpvserver -s rpvserver -t rpvstype \
>-a rpvs_pvid='0022be2aa13f292e' -a client_addr='192.168.101.73,\
192.168.101.74' -a auto_online='n'
rpvserver0 Available
```

Repeat the steps 1 and 2 for the second RPV.

Use **lsrpvserver** to list the RPV servers defined, as shown in Example 18-5:

Example 18-5 Listing the RPV servers

```
frigg:/# lsrpvserver -H
# RPV Server     Physical Volume Identifier       Physical Volume
# -------------------------------------------------------------
  rpvserver0     0022be2aa13f292e                 hdisk1
frigg:/# lsattr -El rpvserver0
auto_online n                                   Configure at System Boot   True
client_addr 192.168.101.73                      Client IP Address          True
client_addr 192.168.101.74                      Client IP Address          True
rpvs_pvid   0022be2aa13f292e0000000000000000 Physical Volume Identifier True
```

Configure the RPV clients. Perform these steps on each client:

4. Run **smitty rpvclient->Add Remote Physical Volume Clients**.

   Provide the IP address of the RPV server, and the local IP address used for data replication. Then select the remote disk from the list, as in Example 18-6

Example 18-6 Adding the RPV client

```
                    Add Remote Physical Volume Clients

Type or select values in entry fields.
```

```
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
  Remote Physical Volume Server Internet Address      10.1.101.192
  Remote Physical Volume Local Internet Address       192.168.101.74
  PV Identifiers                          0022be2aa13f292e0000000000000000
  I/O Timeout Interval (Seconds)                      [180] #
  Start New Devices Immediately?                      [yes] +

F1=Help              F2=Refresh           F3=Cancel            F4=List
F5=Reset             F6=Command           F7=Edit              F8=Image
F9=Shell             F10=Exit             Enter=Do
```

From the command line, see Example 18-7:

Example 18-7 Adding RPV client, using CLI

```
thor:/# /usr/sbin/mkdev -c disk -s remote_disk -t rpvclient \
>-a pvid='0022be2aa13f292e' -a server_addr='10.1.101.192' \
>-a local_addr='192.168.101.73' -a io_timeout='180'
hdisk6 Available

thor:/# lsattr -El hdisk6
io_timeout  180                                 I/O Timeout Interval       True
local_addr  192.168.101.73                      Local IP Address           True
pvid        0022be2aa13f292e0000000000000000 Physical Volume Identifier True
server_addr 10.1.1.192                          Server IP Address          True
```

At this time the disk devices are created on the client and can be used for integrating in a volume group and defining the logical volume mirrors. Use lsrpvclient to list the defined client RPVs.

At the operating system level, they are defined as normal hdisks. The LVM commands used for local volumes applies to the RPVs, too. <Example> shows the output of the **lsdev** command (Example 18-8):

Example 18-8 Listing of the physical volumes defined on thor

```
thor:/# lspv
hdisk0          0022be2a80b97feb                    rootvg          active
```

```
hdisk1          none                            None
hdisk2          0022be2aa13dc072                vg01            concurrent
hdisk3          0022be2aa13ea83e                vg02            concurrent
hdisk4          none                            None
hdisk5          none                            None
hdisk6          0022be2aa13f292e                None
```

Note: The PVID of the RPV client is the same as the PVID of the remote disk.

Repeat the steps 1-3 to create the reverse RPV pair, associating an RPV server for the local disk in node thor and a RPV client on node frigg.

Repeat the same step for the node odin, using as the local communication address, odin_geo1.

Define the LVM mirroring

5. Extend the volume group, containing primary data with the defined RPVs. Use the GLVM menus in smit to extend the volume group. Run **smitty glvm_vg->Add Remote Physical Volumes to a Volume Group**, or use the **extendvg** command:

   ```
   extendvg vg01 hdisk6
   ```

6. Mirror the volume group containing the RPVs

   Note: Before mirroring a logical volume you must change the allocation policy to superstrict. Use chlv -s s < lv_name> -u <upper_bound> to change the allocation policy to superstrict. Refer the man page for chlv for further details.

   In Example 18-9 we present how we changed the logical volumes ulv11_log and ulv11:

Example 18-9 Changing the logical volumes

```
thor:/# chlv -s s -u 2 ulv11_log
thor:/# lslv ulv11_log
LOGICAL VOLUME:     ulv11_log              VOLUME GROUP:   vg01
LV IDENTIFIER:      0022be2a00004c0000000104d52d0c6d.1 PERMISSION:
read/write
VG STATE:           active/complete        LV STATE:       opened/syncd
TYPE:               jfs2log                WRITE VERIFY:   off
MAX LPs:            512                    PP SIZE:        16 megabyte(s)
COPIES:             1                      SCHED POLICY:   parallel
LPs:                1                      PPs:            1
STALE PPs:          0                      BB POLICY:      relocatable
```

```
INTER-POLICY:       minimum                 RELOCATABLE:    yes
INTRA-POLICY:       middle                  UPPER BOUND:    2
MOUNT POINT:        N/A                     LABEL:          None
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LP COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?:     NO
thor:/# chlv -s s -u 2 ulv11
thor:/# lslv ulv11
LOGICAL VOLUME:     ulv11                   VOLUME GROUP:   vg01
LV IDENTIFIER:      0022be2a00004c0000000104d52d0c6d.2 PERMISSION:
read/write
VG STATE:           active/complete         LV STATE:       opened/syncd
TYPE:               jfs2                    WRITE VERIFY:   off
MAX LPs:            512                     PP SIZE:        16 megabyte(s)
COPIES:             1                       SCHED POLICY:   parallel
LPs:                10                      PPs:            10
STALE PPs:          0                       BB POLICY:      relocatable
INTER-POLICY:       minimum                 RELOCATABLE:    yes
INTRA-POLICY:       middle                  UPPER BOUND:    2
MOUNT POINT:        N/A                     LABEL:          /app01
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LP COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?:     NO
thor:/#
```

Mirror the volume group by running smitty glvm_vg → Add a Remote Site Mirror Copy to a Logical Volume. You can use **mirrorvg** command to mirror the volume group or **mklvcopy** to mirror the logical volumes, as example:

```
/usr/sbin/mklvcopy  -s's' ulv11_log 2 hdisk6
```

Check the status of the volume group and logical volumes using **lsvg**, as in Example 18-10:

Example 18-10 Using lsvg to query the status of the logical volume mirrors

```
thor:/# lsvg -p vg01
vg01:
PV_NAME           PV STATE      TOTAL PPs   FREE PPs    FREE DISTRIBUTION
hdisk2            active        639         476         128..00..92..128..128
hdisk6            active        639         478         128..02..92..128..128
thor:/# lsvg -l vg01
vg01:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
ulv11_log           jfs2log    1     2     2    open/syncd    N/A
ulv11               jfs2       160   320   2    open/stale    N/A
ulv11_sm            statemap   1     1     1    open/syncd    N/A
```

```
ulv11_log_sm        statemap   1     1     1    open/syncd    N/A
```

7. Stop the cluster services gracefully on the local node, using smitty clstop menu. Check the proper termination of the cluster resource processing. Use **lsgmd** to verify that the GMDs in “Defined” state.
8. On each node in the cluster, change the file system definition in /etc/filesystems file, to use the regular logical volumes, instead of the GMDs. See Example 18-11.

Example 18-11 Changing the file systems for working with the logical volumes

```
/app01:
        dev             = /dev/ulv11
        vfs             = jfs2
        log             = /dev/ulv11_log
        mount           = false
        check           = false
        account         = false
```

Important: If initially you have created the file systems using the **crfs** command the LVCB (logical volume control block) will get updated with the file system information, so that each **importvg** command will update the /etc/filesystems. You can check for LVCB data using **getlvcb -AT <lv_name>**. If you have created the fleshiest over GMD, using **mkfs**, the **importvg** command will not update the fleshiest information in the /etc/filesystem file.

9. Change the HACMP topology and resource definitions to use GLVM.

Note: HACMP/XD HAGEO does not support dynamic reconfiguration. You must stop the cluster services for changing the cluster configuration. HACMP/XD GLVM supports dynamic reconfiguration as long as you don’t have HAGEO installed.

For integrating the GLVM volume groups in HACMP you need to ensure that each logical volume is replicated. HACMP issues an error message if the geographically mirrored volume groups contains unreplicated logical volumes.

10. Reconfigure the cluster topology.
11. Change the network type from Geo_Primary to XD_data. At the time the redbook was published, two XD_data networks were not supported. You can have GMDs and RPVs configured in the same time in the cluster. However, the GMD and RPV resources cannot be part of the same resource group. If you have two Geo_Primary networks, you can leave the second network for the unconverted GMDs.

Example 18-12 shows how we changed the first Geo_Primary network in an XD_data type.

Example 18-12 Converting the HAGEO network into an XD_data network

```
                Change/Show an IP-Based Network in the HACMP Cluster

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                        [Entry Fields]
* Network Name                                       net_Geo_Primary_01
  New Network Name                                    [XD_data_net_01]
* Network Type                                              [XD_data]+
* Netmask                                             [255.255.255.0]+
* Enable IP Address Takeover via IP Aliases                        No+
  IP Address Offset for Heartbeating over IP Aliases               []
* Network attribute                                           public+



F1=Help              F2=Refresh           F3=Cancel            F4=List
F5=Reset             F6=Command           F7=Edit              F8=Image
F9=Shell             F10=Exit             Enter=Do
```

Note: If your are changing your Geo_Primary network attribute from private to public, you have to remove the network and recreate it.

12. Synchronize the cluster topology.
13. Changing the resource groups to integrate the RPVs. You don't have to configure special resources for using the RPVs in the cluster. At this time you should remove the GMD definitions from the resource groups (see Example 18-13).

Example 18-13 Defining the resource group in HACMP

```
Change/Show All Resources and Attributes for a Resource Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.

                                                  [Entry Fields]
  Resource Group Name                            app01_rg
  Inter-site Management Policy                   Prefer Primary Site
  Participating Nodes from Primary Site          thor odin
  Participating Nodes from Secondary Site        frigg
```

```
  Startup Policy                                   Online On Home Node Only
  Fallover Policy                                 Fallover To Next Priority
Node In The List
  Fallback Policy                                  Fallback To Higher
Priority Node In The Li>
  Fallback Timer Policy (empty is immediate)      []                      +
  Service IP Labels/Addresses                     []                      +
  Application Servers                             [app01_srv]             +
  Volume Groups                                   [vg01 ]                 +
  Use forced varyon of volume groups, if necessary  true                  +
  Automatically Import Volume Groups               false                  +
  Filesystems (empty is ALL for VGs specified)    [/app01 ]               +
  Filesystems Consistency Check                    fsck                   +
  Filesystems Recovery Method                      sequential             +
  Filesystems mounted before IP configured         false                  +
  Filesystems/Directories to Export               []                      +
  Filesystems/Directories to NFS Mount            []
  Network For NFS Mount                           []                    +
  Tape Resources                                  []                    +
  Raw Disk PVIDs                                  []                    +
  Fast Connect Services                           []                    +
  Communication Links                             []                    +
  Primary Workload Manager Class                  []                    +
  Secondary Workload Manager Class                []                    +
  Miscellaneous Data                              []
  GeoMirror Devices                               []                    +

F1=Help        F2=Refresh        F3=Cancel              F4=List
F5=Reset       F6=Command        F7=Edit                F8=Image
F9=Shell       F10=Exit          Enter=Do
```

14. Synchronize the cluster definition across the nodes.

15. Start the cluster on the nodes.

Part 6

Appendixes

A

Paper planning worksheets

Detailed Paper Planning Worksheets are found in Appendix A of the **HACMP 5.3 Planning and Installation Guide**.

We have found that it is useful to tailor these worksheets into a format that fits your environment. To that end, we have included in this appendix, a set of tailored worksheets to help with the design of a simple cluster.

Two-node cluster configuration assistant

Use this table if you plan to use the two-node cluster configuration assistant to configure your initial cluster. The two-node cluster configuration assistant simply requires the following information in order to setup a simple 2 node cluster with a single resource group.

Table A-1 Two-node Cluster Configuration Assistant

HACMP CLUSTER WORKSHEET Two NODE CONFIGURATION ASSISTANT	**DATE:**
Communication Path to Takeover Node	
Application Server	
Application Start Script	
Application Stop Script	
Service IP Label	

Node planning worksheets

The following Planning Worksheets can be used to guide you through the planning and implementation of an HACMP cluster. You will notice that the worksheets cover all the important aspects of the cluster configuration and follow a logical planning flow.

Table A-2 Cluster overview

HACMP CLUSTER WORKSHEET - PART 1 of 11 CLUSTER OVERVIEW	**DATE:**
CLUSTER NAME	
ORGANIZATION	
NODE 1 HOSTNAME	
NODE 2 HOSTNAME	
NODE 1 HACMP NAME	
NODE 2 HACMP NAME	
COMMENTS	

Table A-3 Cluster Hardware

HACMP CLUSTER WORKSHEET - PART 2 of 11 CLUSTER HARDWARE		**DATE:**
HARDWARE COMPONENT	**SPECIFICATIONS**	**COMMENTS**
COMMENTS		

Table A-4 Cluster Software

HACMP CLUSTER WORKSHEET - PART 3 of 11 CLUSTER SOFTWARE		**DATE:**
SOFTWARE COMPONENT	**VERSION**	**COMMENTS**
AIX		
RSCT		
HACMP		
COMMENTS		

Table A-5 Cluster Ethernet Networks

HACMP CLUSTER WORKSHEET - PART 4 of 11 CLUSTER ETHERNET NETWORKS					**DATE:**
NETWORK NAME	**NETWORK TYPE**	**NETMASK**	**NODE NAMES**	**IPAT VIA IP ALIASES**	**IP Address Offset for Heartbeating over IP Aliases**
COMMENTS					

Table A-6 Cluster Serial Networks

HACMP CLUSTER WORKSHEET - PART 5 of 11 CLUSTER POINT TO POINT AND SERIAL NETWORKS					**DATE:**
NETWORK NAME	**NETWORK TYPE**	**NODE NAMES**	**Device**	**INTERFACE NAME**	**ADAPTER LABEL**
COMMENTS					

Table A-7 Interfaces

HACMP CLUSTER WORKSHEET - PART 6 of 11 INTERFACES					**DATE:**
node01					
IP Label	**IP Alias Dist. Preference**	**NETWORK INTERFACE**	**NETWORK NAME**	**INTERFACE FUNCTION**	**IP ADDRESS /MASK**
node02					
IP Label	**IP AliasDist. Preference**	**NETWORK INTERFACE**	**NETWORK NAME**	**INTERFACE FUNCTION**	**IP ADDRESS /MASK**
COMMENTS					

Table A-8 Shared Disks

HACMP CLUSTER WORKSHEET - PART 7 of 11 SHARED DISKS					**DATE:**
node01			**node01**		
VGNAME	**VPATHS**	**HDISK**	**HDISK**	**VPATHS**	**VGNAME**
COMMENTS					

Table A-9 Shared Volume Groups

HACMP CLUSTER WORKSHEET - PART 8 of 11 SHARED VOLUME GROUPS (NON-CONCURRENT)		**DATE:**
RESOURCE GROUP	**VOLUME GROUP 1**	**VOLUME GROUP 1**
COMMENTS	Create the shared Volume Group on the first node and then import on the second node. #importvg -y app1vg -V 90 vpath0 (may have to make the pv available with chdev -l vpath0 -a pv=yes) #chvg -an app1vg (set vg to not auto vary on) #mount /app1 (ensure the filesystem mounts) #umount /app1 #varyoffvg app1vg (leave VG offline in order for HACMP to manage)	

Table A-10 Application Worksheet

HACMP CLUSTER WORKSHEET - PART 9 of 11 APPLICATION WORKSHEET				**DATE:**
APP1				
ITEM	**DIRECTORY**	**FILESYSTEM**	**LOCATION**	**SHARING**
EXECUTABLE FILES				
CONFIGURATION FILES				
DATA FILES				
LOG FILES				
START SCRIPT				
STOP SCRIPT				
FAILOVER STRATEGY				
NORMAL START COMMANDS AND PROCEDURES				
VERIFICATION COMMANDS AND PROCEDURES				
NORMAL START COMMANDS AND PROCEDURES				
NODE REINTEGRATION				

HACMP CLUSTER WORKSHEET - PART 9 of 11 APPLICATION WORKSHEET				DATE:
APP2				
ITEM	DIRECTORY	FILESYSTEM	LOCATION	SHARING
EXECUTABLE FILES				
CONFIGURATION FILES				
DATA FILES				
LOG FILES				
START SCRIPT				
STOP SCRIPT				
FAILOVER STRATEGY				
NORMAL START COMMANDS AND PROCEDURES				
VERIFICATION COMMANDS AND PROCEDURES				
NORMAL START COMMANDS AND PROCEDURES				
NODE REINTEGRATION				
COMMENTS				

Table A-11 Application Monitoring

HACMP CLUSTER WORKSHEET - PART 10 of 11 APPLICATION MONITORING	**DATE:**
APP1	
Can this Application Be Monitored with Process Monitor?	
Processes to Monitor	
Process Owner	
Instance Count	
Stabilization Interval	
Restart Count	
Restart Interval	
Action on Application Failure	
Notify Method	
Cleanup Method	
Restart Method	
APP2	
Can this Application Be Monitored with Process Monitor?	
Processes to Monitor	
Process Owner	
Instance Count	
Stabilization Interval	
Restart Count	
Restart Interval	
Action on Application Failure	
Notify Method	
Cleanup Method	
Restart Method	

Table A-12 Resource Groups

HACMP CLUSTER WORKSHEET - PART 11 of 11 RESOURCE GROUPS)		**DATE:**
RESOURCE NAME		
Inter-Site Management Policy		
Participating Node Names		
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		
Application Servers		
Volume Groups		
Filesystems		
Filesystem Consistency Check		
Filesystems Recovery Method		
Filesystems/Directories to Export		
Filesystems/Directories to NFS mount		
Network for NFS mount		
Primary WLM Class		
Auto Import Volume Groups		
Filesystems Mounted before IP Configured.		
COMMENTS		

Abbreviations and acronyms

ACL	Access Control List
AIX	Advanced Interactive Executive
API	Application Programming Interface
ARP	Address Resolution Protocol
ATM	Asynchronous Transfer Mode
BOS	Base Operating System
CA	Certificate Authority
C-CPOC	Cluster Single Point Of Control
CEC	Central Electronic Complex
CGI	Common Gateway Interface
CLI	Command Line Interface
CLVM	Concurrent Logical Volume Manager
CPU	Central Processing Unit
CSM	Cluster Systems Management
CWDM	Coarse Wave Division Multiplexing
CWOF	Cascading Without Fallback
DAC	Disk Array Controller
DARE	Dynamic Reconfiguration
DBFS	Dial Back Fail Safe
DES	Data Encryption System
DLPAR	Dynamic LPAR
DNP	Dynamic Node Priority
DNS	Domain Name Service
DWDM	Dense Wave Division Multiplexing
ECM	Enhanced Concurrent Mode
ESS	Enterprise Storage Server
FC	Fibre Channel
FCIP	Fibre Channel IP
FDDI	Fiber Distributed Data Interface
GLVM	Geographical LVM
GMD	Geographic Mirror Device
GPFS	General Parallel File System
HACMP	High Availability Cluster Multi-Processing
HACMP/ES	HACMP Enhanced Scalability
HACMP/XD	HACMP Extended Distance
HA-NFS	High Availability NFS
HBA	Host Bus Adapter
HPS	High Performance Switch
HSC	Hardware Service Console
HWAT	Hardware Address Takeover
IBM	International Business Machines Corporation
IHS	IBM Http Server
IPAT	IP Address Takeover
ITSO	International Technical Support Organization
JBOD	Just a Bunch Of Disks
JFS	Journal File System
JRE	Java Runtime Environment
LAA	Locally Administered Address
LAN	Local Area Network
LDAP	Lightweight Directory Application Protocol
LPAR	Logical Partition
LUN	Logical Unit Number
LV	Logical Volume
LVCB	Logical Volume Control Block
LVDD	Logical Volume Device Driver

LVM	Logical Volume Manager
MAC	Media Access Control
MAL	Mechanism Abstraction Layer
MIB	management Information Base
MPIO	Multi-Path I/O
MPM	Mechanism Pluggable Module
MTU	Media Transmission Unit
NFS	Network File System
NIM	Network Interface Module
POL	Priority Override Location
PP	Physical Partition
PPK	Private-Public Key
PPRC	Peer-to-Peer Remote Copy
PV	Physical Volume
PVID	Physical Volume ID
RAC	Real Application Cluster
RAID	Redundant Array of Independent Disks
RDAC	Redundant Disk Array Controller
RG	Resource Group
RM	Resource Monitor
RMC	Resource Monitoring and Control
RPV	Remote Physical Volume
RSCT	Reliable Scalable Clustering Technology
SAN	Storage Area Network
SCSI	Small Computer System Interface
SDD	Subsystem Device Driver
SPOF	Single Point Of Failure
THL	Trusted Host List
VG	Volume Group
VGDA	Volume Group Descriptor Area
VIO	Virtual I/O
VIOS	Virtual I/O Server
VLAN	Virtual LAN
WLM	Workload manager

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 710. Note that some of the documents referenced here may be available in softcopy only.

- *HACMP/ES Customization Examples,* SG24-4498
- *Understanding LDAP - Design and Implementation,* SG24-4986
- *Advanced POWER Virtualization on IBM System p5*, SG24-7940

Other publications

These publications are also relevant as further information sources:

- *High Availability Cluster Multi-Processing Administration Guide*, SC23-4862
- *High Availability Cluster Multi-Processing for AIX 5L Planning and Installation Guide*, SC23-4861
- *High Availability Cluster Multi-Processing XD (Extended Distance) for HAGEO Technology: Concepts and Facilities*, SA22-7955
- *HACMP/XD for ESS PPRC Version 5.3: Planning and Administration Guide*, SC23-4863
- *HACMP/XD for Geographic LVM: Administration and Planning Guide*, SA23-1338

Online resources

These Web sites and URLs are also relevant as further information sources:

- OpenSSL project Web site

  `http://www.openssl.org`

- OpenSSH project page

http://www.openssh.org

- OpenSSH on AIX

  http://www.sourceforge.net/projects/openssh-aix

- HACMP recommended maintenance levels

  http://www-912.ibm.com/eserver/support/fixes/fcgui.jsp

- Availant home page; testers for non-IBM hardware

  http://www.availant.com

How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

C

D

E

F

G

H

I

J

K

L

M

N

O

P

Q

R

S

T

U

V

W

X

Z

Implementing High Availability Cluster Multi-Processing (HACMP) Cookbook

(1.0” spine)
0.875”<->1.498”
460 <-> 788 pages

Implementing High Availability Cluster Multi-Processing (HACMP) Cookbook

Redbooks

Extended case studies with practical disaster recovery examples

Explore the latest HACMP and HACMP/XD V5.3 features

Advanced POWER virtualization explained

This IBM Redbook will help you install, tailor and configure the new HACMP V5.3, and understand the new and improved features like Dynamic LPAR integration, Virtual I/O, and Disaster Recovery (DR) configurations.

This redbook gives a broad understanding of the HACMP and HACMP Extended Distance (HACMP/XD) architecture. If you plan to install, migrate, or merely administer a high availability cluster, this book is right for you. Disaster recovery elements and how HACMP fulfills these necessities are also presented in detail.

This cookbook helps AIX professionals that are seeking a comprehensive and task-oriented guide for developing the knowledge and skills required for HACMP cluster design and implementation as well as for daily system administration. It is designed to provide a combination of theory and practical experience.

This book will be especially useful for system administrators currently running both HACMP/ES and HACMP Extended Distance (XD) clusters who may want to consolidate their environment and move to a new HACMP V5.3. There is a detailed description of a node-by-node migration to HACMP/ES 5.3 and a comprehensive discussion about how to prepare for an upgrade or migration.

INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION

BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

For more information:
ibm.com/redbooks

SG24-6769-00 ISBN 0738494143