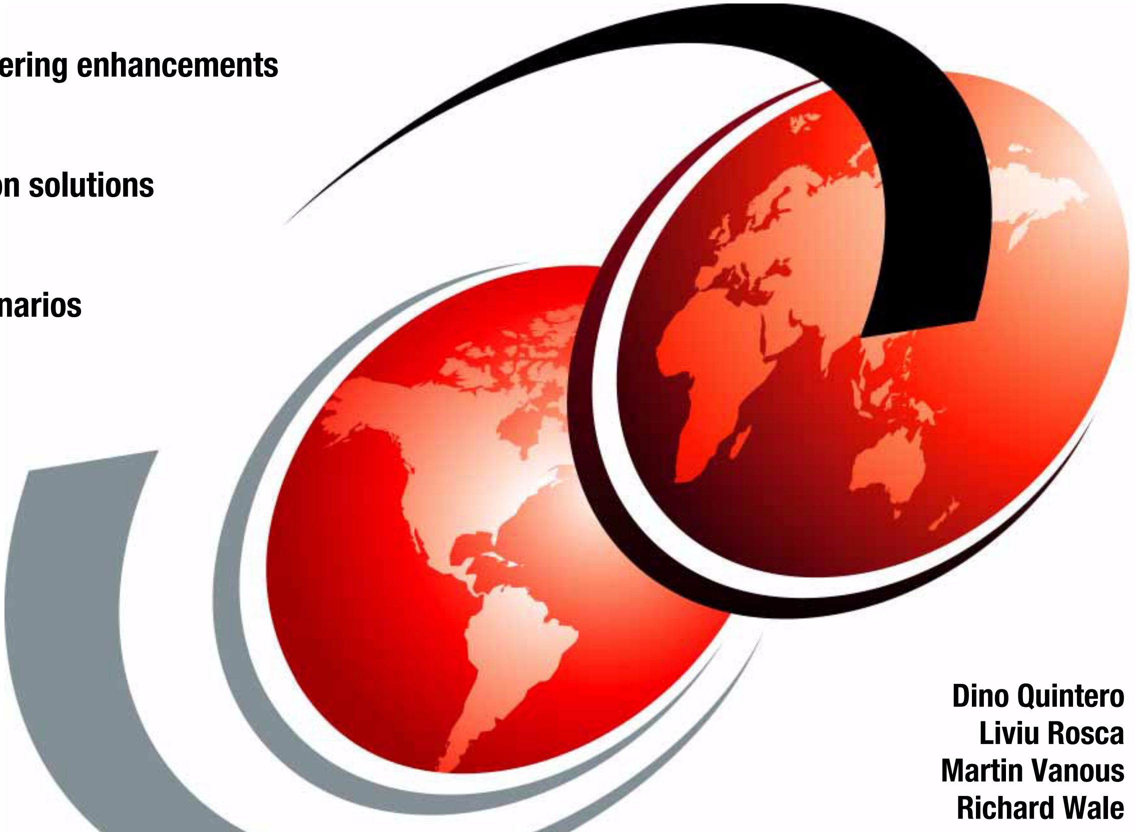
Virtualization and Clustering Best Practices Using IBM System p Servers

Latest clustering enhancements revealed

Virtualization solutions

Sample scenarios included



Dino Quintero
Liviu Rosca
Martin Vanous
Richard Wale
Jun Yoshino
Octavian Lascu

Redbooks

ibm.com/redbooks

International Technical Support Organization

Virtualization and Clustering Best Practices Using IBM System p Servers

May 2007

SG24-7349-00

Note: Before using this information and the product it supports, read the information in "Notices" on page vii.

First Edition (May 2007)

This edition applies to IBM System p4, IBM System p5, AIX 5L V5.3, Virtual I/O Server (VIOS) Version 1.2.0 and 1.3.0, Hardware Management Console (HMC) Version 3.3.7 and 5.2.1, Cluster Systems Management (CSM) Version 1.5.1 and 1.6.0, High Availability Cluster Multi-Processing (HACMP) Version 5.4.

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 illustrate 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.

Trademarks

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

i5/OS®
pSeries®
AIX 5L™
AIX®
BladeCenter®
Blue Gene®
Chipkill™
Domino®
DB2®
DS4000™
General Parallel File System™
GPFS™
HACMP™
IBM®
LoadLeveler®
Lotus®
Micro-Partitioning™
NetView®
OpenPower™
Passport Advantage®
POWER™
POWER Hypervisor™
POWERparallel®
POWER4™
POWER5™
POWER5+™
Redbooks®
Redbooks (logo) ®
RS/6000®
System p™
System p5™
System x™
System Storage™
Tivoli Enterprise™
Tivoli Enterprise Console®
Tivoli®
Viewpoint™
WebSphere®

The following terms are trademarks of other companies:

SAP, and SAP logos are trademarks or registered trademarks of SAP AG in Germany and in several other countries.

ITIL is a registered trademark, and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

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

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries 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, or service names may be trademarks or service marks of others.

Preface

This IBM® Redbooks publication highlights and demonstrates, through practical examples, clustering technologies, and principles that involve IBM System p™ servers and various IBM management software. Different viewpoints are exposed and analyzed, to help reveal areas of importance that may be exploited in reducing the total cost of ownership of your IT environment.

This book will help you install, tailor, and configure IBM System p5 and exploit its advanced features, such as Advanced POWER Virtualization, which provides new ways to get more out of your machine and therefore more from your investment.

This book will not tell you which server to buy, or which technology is best for you, or mandate the "right" way to deploy and administer your environment. This book shows you available methods, potential options, and different viewpoints. It also shows you how to get more from your system, and that to get more is not always as complicated as you might expect.

This book shows you that many "right" answers may exist for given problems, and also illustrates points to consider when evaluating potential solutions. We demonstrate what you need to understand, appreciate, and consider when making a choice or when to change a decision. Making the wrong choice may prove more expensive to manage, than the apparently expensive choices you make when purchasing elements for your IT environment.

The team that wrote this book

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

Dino Quintero is a Senior Certified Consulting IT Specialist and the worldwide Technical Marketing Manager for the IBM System Blue Gene® Solution. Before joining the Deep Computing Marketing Team, he was a Clustering Project Leader for the ITSO. He worked as a Clustering Performance Analyst for the Enterprise Systems Group focusing on industry benchmarks, such as TPC-C, TPC-H, and TPC-W. He also worked as a Disaster Recovery Architect for IBM Services and focused on backup and recovery solutions for large customers. He has been with IBM since 1996, and in the IT industry since 1992. His areas of expertise include enterprise backup and recovery, disaster recovery planning and implementation, and clustering architecture and solutions. He is an IBM

Certified Specialist on System p Administration and System p Clustering. Currently, he focuses on planning, influencing, leading, managing, and marketing IBM Blue Gene solutions. He also delivers technical lectures worldwide.

Liviu Rosca is an IT Specialist at IBM Services, Romania, for four years. His area of expertise includes IBM eServer pSeries® systems, AIX®, HACMP™ and WVR. He is IBM Certified AIX 5L™ and HACMP System Administrator and CCNP. He teaches AIX 5L and HACMP classes.

Martin Vanous is a Team Leader of UNIX® and Tivoli® Group in Czech Republic. He has nine years of experience in the AIX field. His areas of expertise include HACMP, APV, CSM, Tivoli, as well as developing System p clusters and solutions.

Richard Wale is an IT Specialist working for IBM UK. He holds a bachelor's (Hons) degree in Computer Science from Portsmouth University, England. He has eight years AIX and System p experience and is an IBM Certified Specialist in both areas.

Jun Yoshino is an IT Specialist working for IBM Japan Systems Engineering Co., Ltd. He has eight years of experience in AIX and System p field. His areas of expertise include APV and IVM.

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

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

Bruce Potter, William Wajda, Norman Nott, Joshua Horton, John Simpson, Linda Mellor, Lissa Valletta, Michael Schmidt, Joseph Chaky, Paul Swiatocha
IBM Poughkeepsie

Amarty Pears, Erin Burke
IBM Austin

Corneliu Holban
IBM New York

Steve Willoughby
IBM UK

Miloslav Cepelka, Vojta Šrejber
gedas CZ, Check Republic

Gabrielle Velez
International Technical Support Organization

Become a published author

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

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 IBM Redbooks form found at:

  ibm.com/redbooks

- Send your comments in an email to:

  redbooks@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. HYTD Mail Station P099
  2455 South Road
  Poughkeepsie, NY 12601-5400

1

Introduction

In this chapter we summarize the technologies and areas we will discuss in the subsequent chapters. This high level overview will highlight areas which maybe relevant to your environment. You may consider some of our observations obvious, however not everything is obvious to all. Some customers may be unaware of certain features or believe features too complex to implement within their environment. We will explain the use of some technologies to demonstrate how straightforward they can be. We will also provide overviews, considerations and links to more detailed documentation.

This chapter contains the following:

- “Management summary” on page 2
- “Potential business advantage” on page 12

1.1 Management summary

It is important to appreciate the range of technologies available and their benefits and implications. It is equally important to understand that a given feature may not bring equal benefit to every environment. Much depends on the scale and requirements of an environment and hosted applications. Similar considerations exist with hardware choice; a number of available options could be suitable. You need to factor many viewpoints and considerations during the decision making process.

Note: Throughout our discussions a number of common themes reoccur. Even though the information may seem redundant, discussing the topics or technologies from multiple viewpoints is important to emphasize their various aspects. Therefore, repeated conclusions or recommendations are beneficial for stressing the benefits of such implementations.

This initial section summarizes areas and technologies which will be discussed or referenced in subsequent chapters.

1.1.1 IBM System p5

In the current generation of IBM 64-bit POWER™-based UNIX servers, this family builds on the heritage of the previous pSeries and RS/6000® generations. The current product line is based on the POWER5 microprocessor; this being the successor to the POWER4 microprocessor found in the previous pSeries family. At the time of writing IBM System p5 spans a broad range of configurations scaling from 1 up to 64 CPUs (Entry, Mid-range and High-end). Raw performance is not the only beneficial quality; expansion, capability, flexibility, reliability, scalability, and technical innovation are equally prevalent.

1.1.2 Logical Partitioning (LPAR)

This feature was first introduced into the IBM System p4 product. It is the ability to carve logical separate servers from one physical server. Each logical server will have its own CPU, memory, and I/O resources and run its own instance of an operating system. It allows many individual servers, of varying configuration to be hosted on a single physical server - promoting server consolidation. Within the capacity of a physical server, the LPAR definitions can be of varying sizes and configurations. LPAR profiles can be modified and new ones added, as your requirements change over time. Additionally LPAR-capable servers can still be installed as a single operating system instance.

Note: For a complete description of LPAR technology and implementations with regard to IBM System p, refer to *Partitioning Implementations for IBM p5 Servers*, SG24-7039.

1.1.3 Dynamic Logical Partitioning (DLPAR)

DLPAR was another feature introduced with the IBM System p4 product family. It is the ability to dynamically add or remove resources (CPU, memory, or I/O) from an active LPAR - without requiring a partition (operating system) restart. You can also remove resources from an active LPAR, again without a restart. There are limitations and implications of the feature; for example the given LPAR needs to be running AIX 5L V5.2 or higher. IBM System p5 improves the granularity of the feature, by allowing smaller amounts of CPU/RAM resource to be moved.

Note: For examples of DLPAR, refer to 4.1.8, “DLPAR managed by IVM” on page 162.

1.1.4 Micro-Partitioning

Micro-Partitioning™ is one of the new functions provided by Advanced POWER Virtualization feature of IBM System p5. Whereas a dedicated LPAR is allocated physical CPUs, Micro-partitions share physical processors; that is a partition will be allocated fractions of a physical processor. When creating an LPAR, you select whether it will use shared or dedicated processors.

Note: Micro-Partitioning is only supported on LPARs running AIX 5L V5.3 (or higher) and certain supported Linux distributions. It is not supported on AIX 5L V5.1 or AIX 5L V5.2.

Partitions which share physical resources are called shared processor partitions, compared to a dedicated processor partition which exclusively uses whole processors. Using Micro-Partitioning, you can allocate 0.1 (10%) processor to achieve 10 LPARs per physical CPU; depending on your application requirements this could allow more LPARs to be hosted per physical server.

Weights or priorities can be assigned to micro-partitions to allow dynamic balancing; if workload is low enough in one shared processor partition, other partitions can consume capacity beyond their entitlement. As such Micro-Partitioning provides opportunities to use system processor resources more efficiently and ability to react to sudden rise of workload. Micro-Partitioning can also provide increased resilience to failure for shared partitions; in the unlikely event of a CPU or memory DIMM failure, the failed device is removed from the shared pool; leaving the micro-partitions active with the remaining capacity.

1.1.5 Hardware Management Console (HMC)

The HMC (sometimes referred to as the “Hardware Service Console”) is a separate machine which provides hardware control of managed systems (physical IBM System p and System i servers). It is mandatory on the high-end servers (p/i570 and above); it provides LPAR management, dynamic resource allocation, monitoring and power control.

One valuable HMC feature is the ability to “Call Home” automatically and raise Hardware calls to IBM Service in the event of a detected problem. The HMC maintains a unified event log for all its managed servers, which can be used for diagnostic and alerting purposes.

> **Note:** At the time of writing, a single HMC will support up to 48 non p590/595, or 32 p590/595 servers and a maximum of 254 LPARs across the managed systems.

For critical deployments a redundant (or secondary) HMC is an available option. Should an HMC fail the managed systems remain available, but unless a redundant HMC is present administrators will loose access to the controlling features (including dynamic resource movement).

1.1.6 Virtualization and Advanced POWER Virtualization

Virtualization of hardware resources is the ability to provide finer granularity and sharing of system resources. It is a natural progression of the concept of LPARs introduced with IBM System p4. One feature of the Advanced POWER Virtualization option of IBM System p5 enables physical resources to be virtualized from an owning LPAR and allocated or shared to client LPARs. This allows transparent resource sharing of:

- CPU
- Network adapters (Ethernet, InfiniBand)
- Storage

Practically this means that more than one operating system can use a single CPU (refer to 1.1.4, “Micro-Partitioning” on page 3), multiple LPARs (AIX and Linux) can share physical Ethernet, Fibre Channel attached LUNs or internal disks.

One can choose to use virtualization for all mentioned components or only for selected resources (only CPU for example) or to use the whole machine.

Hardware virtualization provides the potential to increase the utilization of a physical machine. Improved machine utilization increases the return on investment.

Elements of Advanced POWER Virtualization are described and discussed in subsequent chapters.

1.1.7 Virtual Ethernet

This feature provides inter-LPAR communication without the requirement of dedicated physical network adapters for every LPAR. Virtual Ethernet allows administrators to define memory-based point-to-point communication between LPARs.

These connections are similar to regular high-speed Ethernet medium. The Virtual Ethernet feature was introduced on IBM System p5 and is therefore not available on IBM System p4. LPARs also need to be running AIX 5L V5.3 or certain variants of Linux. It is provided by default with the hardware and is not part of the Advanced POWER Virtualization feature.

Note: For more detailed discussions regarding Virtual Ethernet, refer to *Advanced POWER Virtualization on IBM System p5*, SG24-7940.

1.1.8 Virtual I/O Server (VIOS)

The Virtual I/O Server is part of the IBM System p5 Advanced POWER Virtualization feature. The software is installed into a dedicated partition created as Virtual I/O partition. Virtual Ethernet and Virtual SCSI are components of VIOS, allowing partitions to share physical I/O resources. VIOS provides virtual host adapters to client partitions; within VIOS, a virtual host adapter is mapped to a physical storage device configured via a Virtual Target Device. This allows client partitions to access physical resources and share it.

VIOS also provides network connectivity between internal client partitions and external networks, using the Shared Ethernet Adapter feature.

By leveraging virtual devices, partitions can be created without additional physical I/O resources and use virtual devices more efficiently. There are many ways to leverage VIOS, for various benefits; reduced or consolidated physical resources; more logical environment design; resource flexibility. For example, if your LPARs do not require the full bandwidth offered by a device, then sharing a physical card between LPARs will provide higher utilization of that card.

Note: For sample VIOS configurations, refer to 4.1.2, “IVM setup” on page 126.

1.1.9 Integrated Virtualization Manager (IVM)

Beginning with IBM System p4, there were two approaches to system usage: full system and LPAR. To utilize the LPAR feature or to manage larger server environments a Hardware Management Console (HMC) was required. However there are situations where an HMC may not be the most appropriate or cost-effective solution.

IVM is a new option available on entry to mid range IBM System p5 servers. It provides a subset of HMC’s management and Advanced POWER Virtualization feature, without the need for a separate HMC; IVM is an enhancement to the VIO Server, and therefore is hosted on a VIOS LPAR.

You should consider using IVM in one of the following scenarios:

- Your environment comprises only a few entry level systems.
- You do not need all functions provided by the HMC.
- Investing in an HMC increases the cost of deployment and ownership.

IVM provides a simple, easy-to-use Web-based interface that allows you to perform basic administrative functions. This interface is different to the WebSM interface and client used by HMCs. Supported Web-browsers are used to remotely managed IVMs.

Note: At the time of writing, IVM can only manage the physical machine on which it is hosted, whereas an HMC can manage multiple physical machines. For a comparison of HMC and IVM, refer to Table 2-1 on page 20.

1.1.10 Partition Load Manager (PLM)

Partition Load Manager is another component of the Advanced POWER Virtualization feature. It automates redistribution of CPU and RAM resources within one physical machine using DLPAR operations. The PLM behavior depends on administrator-defined policies, allocating and de-allocating resources as required. A policy can contain rules for resource redistribution based on current CPU and memory load.

PLM can be used to:

- Redistribute resources for LPARs within different time periods.
- Monitor the whole machine using monitoring operation mode.
- Redistribute and optimize resource allocation depending on actual load.

PLM is one of the few tools which can also manage IBM System p4 machines based on actual utilization. It is possible to manage multiple IBM System p4 and IBM System p5 resources from a single PLM instance. The PLM instance does not require large amounts of resources itself, as such can be hosted in a small micro-partition using virtual I/O resources.

Note: For subsequent discussions of PLM, refer to 3.4.5, “PLM implementation” on page 123.

1.1.11 Management clusters

A management cluster is the concept of managing your distributed servers from a single or reduced number of points; Management clusters are different from application clusters or high availability clusters. Reducing the points of administration simplifies your environment. Managing servers as a single grouped entity encourages standardization and automation. It is a key driver to reducing the total cost of ownership.

The original example of this was the management software for the RS/6000 SP - Parallel System Support Programs (PSSP). PSSP provided a single point of hardware control, installation, user administration, monitoring and file distribution. In addition to distributed administration, management clusters also provide the ability of remote administration.

Current examples of management cluster products are IBM Director and Cluster Systems Management (CSM). Compared to PSSP which was solely for the IBM System p/AIX family, both CSM and Director provide cross-platform management (both hardware and software).

Note:

- However, CSM and IBM Director’s scope and functionality differ. For further discussions about CSM, refer to 2.5, “Cluster Systems Management (CSM)” on page 56, and 3.3, “CSM implementation in an AIX environment” on page 81.
- For more information about using IBM Director, refer to the Redpaper *IBM Director on IBM System p5*, REDP-4219.

1.1.12 High availability (HA) clusters

It is common place for business-critical applications to offer a high degree of service availability to their clients. From the business point of view there are several variables which need to be clarified before starting to think about high availability solutions.

First, does the desired service level allow for any downtime at all? If not, then you should consider a fault tolerant solution. This approach ensures that failure of any of the system components will not render the service unavailable.

Of course, this level of availability comes with a cost. Factors that should be taken into account are not only the financial consequences, but also bad publicity or loss of customers' confidence and loyalty. Here are some example considerations when planning for highly available solutions:

- What services are required to be highly available? If more than one, what is the priority?
- What is the desired degree of service availability? (for example, is a 5 minute downtime acceptable?)
- What is the cost of implementing a high availability solution (evaluation of hardware, software, and other expenses)?
- Which are the factors that affect service availability?
- What are the factors that can adversely affect the quality of the service? What is the minimum acceptable level for the quality of the service you provide?
- How fast do you need to restore a failed service?
- What types of failures do you want to be automatically detected and acted upon?
- Do you want to customize the reaction of your system to some particular event and trigger some specific action accordingly?
- If your environment allows for planned maintenance, would you be interested in reducing (or even eliminating) the perceived downtime created by that activity?

It should be noted, you need to consider more than just your application. Other parts of your environment also need to be reviewed. The benefits of a highly-available application are reduced if your monitoring solution is not as available or reliable. Any servers hosting your infrastructure need to be just as resilient to failure as your application-hosting servers.

Note: For more information about implementing high availability, refer to 2.1.8, "High availability level" on page 29.

1.1.13 High Availability Cluster Multi-Processing (HACMP)

HACMP is a clustering software solution for systems running IBM System p platforms running AIX, and recently Linux. It helps ensure application availability after a failure occurs at either software or hardware layer. Careful design eliminates or masks resources that can become single point of failure and could affect service availability, should a resource fail.

HACMP also provides resource monitoring, automated failure detection and application recovery. Application servers and their corresponding resources can be automatically moved between different systems that are members of the same HA cluster. HACMP can be leveraged for most type of applications; its suitability will depend on the specific application requirements and how it interacts with network and storage resources. Benefits of an HACMP solution include minimizing or even eliminating downtime required by maintenance, reconfiguration, or upgrade activities.

HACMP Extended Distance (HACMP/XD) provides HACMP functionality between geographic sites. Currently HACMP/XD can be used to provide disaster recovery between two sites. The key function of HACMP/XD is data replication between sites. HACMP/XD can be implemented using one of the following features:

- HAGEO - provides data replication over any type of TCP/IP network, automatic failure detection, site failover and recovery. It is independent of the application, disk technology, nature of data and distances between sites.
- HACMP/XD PPRC feature - provides automated site failover where IBM Enterprise Store Server (ESS) is used in both sites and Peer to Peer Remote Copy (PPRC) is used for mirroring storage volumes.
- Geographic Logical Volume Manager (GLVM) is the latest product which provides data mirroring across TCP/IP networks and automated failover/fallback support for applications which use the geographically mirrored data.

All three sub-components are suitable in certain environments. As with HACMP itself, detailed planning will be required to understand your requirements and how HACMP/XD could be leveraged.

Note: For further discussions regarding HACMP, refer to 4.1, “HACMP scenarios, one IVM-managed CEC” on page 126.

1.1.14 Cluster Systems Management (CSM)

CSM is the product developed as the successor to the legacy PSSP product. It provides a management cluster infrastructure for clusters consisting of both IBM System p and IBM System x™ hardware, running AIX and Linux.

Supporting both mixed and heterogeneous clusters, CSM provides a central management point (known as a CSM Management Server). From here all the client machines can be monitored and controlled. CSM leverages and builds upon existing features within the hardware platforms and operating systems to provide centralized configuration, installation, software maintenance, hardware control and monitoring. Centralized management can provide a more logical environment resulting in standardized support, increased efficiency and lower running costs.

CSM Version 1.6.0 brings improved integration and support, for example VIOS and IVM integration; support for new IBM System p5, IBM System x and IBM BladeCenter hardware; IBM System p5 Service Focal Point monitoring.

Note: For more information about Cluster Systems Management (CSM), refer to 3.3, “CSM implementation in an AIX environment” on page 81.

1.1.15 Network Installation Management (NIM)

NIM is tool for simplifying software installation on AIX 5L and Linux machines. Its main features include:

- AIX 5L and Linux base Operating System (OS) network installation (including network boot), which together with using hardware management points (HMC, IVM, and so on) enables complete remote installations and restores.
- Complete management for RPM (Red Hat Package Manager) and LPP (Licensed Program Product) software packages which includes software updates, installation and other software maintenance tasks. This allows software management of both OS-related and application-related packages.
- Concurrent software installation for individual nodes or multiple node groups.
- Software reports and comparison of the NIM managed nodes.
- Interoperability with CSM and IBM Tivoli Provisioning Manager.

NIM server and client software is shipped as part of AIX 5L itself, there is no additional license or usage costs.

Note: For some practical uses of NIM, refer to 3.4, “AIX Installation strategy” on page 118. For a broader discussion, refer to *NIM From A to Z in AIX 5L*, SG24-7296.

1.1.16 Planning

Planning is the process of defining needs which given solution has to fulfill and finding the best suited solution. The criteria which influences the choice of the overall system design are driven by many factors, for example:

- Solution function requirements.
- Application High Availability (HA) requirements.
- Application Capacity and Performance requirements.
- Time needed for the project implementation.
- Availability or Service Management requirements, such as a Service Level Agreement (SLA).
- Security rules.
- Compliance to the local organization rules and already used technologies.
- Price of the implementation, maintenance and total cost of ownership (TCO).

The planning of the *technology* used affects all of the mentioned factors together with planning of the implementation and support approaches and procedures.

1.1.17 Capacity planning

Capacity planning is prediction and monitoring of the system usage and growth. You need to be aware if your usage is increasing and understand how long the free capacity will remain available.

If existing capacity is exhausted, you need to appreciate the options available and the timing to implement the changes required to bring the system back to the agreed service level. Such a practice is equally important during design phases as well as in post-deployment (normal operating) phase.

Server hardware is not the only variable, infrastructure and environment are just as critical; for example available floor space, power, network bandwidth, cable capacity, secondary and backup storage, all need to be understood and tracked.

Note: For more detailed discussions about planning, refer to Chapter 2, “Planning and concepts” on page 17.

1.1.18 Server consolidation

This is the concept of migrating an existing environment from a given number of physical servers, to a smaller number. Common scenarios where this could be considered would be technology refreshes (either hardware or software) or application migrations.

Server consolidation can provide opportunities for reduction in total cost of ownership (TCO) and also increase the utilization of your investment. For example, reducing operating system images or physical servers (and therefore floor space, cabling, electricity, cooling, and so on) by leveraging innovative features to simplify or restructure your environment will result in decreased maintenance and administration costs, and maybe even in software licensing costs.

Note: For an example of server consolidation, refer to 2.2.1, “Migration paths from RS/6000 Scalable POWERparallel (SP)” on page 42.

1.2 Potential business advantage

This section reverses the viewpoint of the previous sections and discusses a number of business-related goals, illustrating how these might be achieved through technology and process improvements.

1.2.1 Saving money

By this we mean in principal reducing the total cost of ownership (TCO) of your environment through implementing processes or leveraging technology resulting in reduced support costs:

- Server consolidation
  Reducing the number of physical machines will result in reduced cabling, floor space, and power requirements, thus lowering the total cost of ownership. With the increased computing power of IBM System p5, there is the opportunity to reduce the number of hosted images, if applicable for your application. When considering server consolidation, we recommend you revisit your application architecture to validate it against the current technology.
- Replacement of legacy infrastructure
  Dependency on old hardware and software is a great risk to an environment, as hardware and software may become unsupported, thus obsolete.
  Even though in certain cases you might be able to purchase extended support contracts beyond the natural product life, should a failure occur on

such hardware or software, the resolution could be very expensive and/or time consuming. It is common for organizations to plan hardware replacement within 5 years to mitigate such risks.

- Centralized and remote administration
  Leveraging management clusters and points of hardware control capabilities simplifies your environment from the administrative viewpoint. Depending on your environment and application this may lead to multiple administration teams to be consolidated to a single location/team, thus reducing costs even further.

1.2.2 Saving time

By implementing automation and standard procedures, system administrators could significantly reduce the amount of time required time for (manual) repetitive tasks.

- Centralized administration
  Standardizing and streamlining your environment will bring opportunities for automation and customized scripting. For example using NIM for standard installation and patch management will streamline install times, and reduce the manual overhead of software updates (for example, APAR management). In addition to the operating system (AIX), NIM can also be used to install middleware and other applications.
- Standard tools and procedures
  These enable redirection and hand over of the second and third level support. Standardization also reduces expensive work of the senior experts.

1.2.3 Improving availability

Availability is usually defined as a percentage of the total time a system is available for its end users. Expected availability is usually included as part of the Service Level Agreement (SLA) for an environment. In addition to the outages generated by software or hardware errors, servers may also appear as unavailable under heavy load. Among the ways to mitigate the risk of a system becoming unavailable and to improve system resilience to various factors, we find:

- Reducing the amount of single points of failure within your environment by deployment of redundant elements (where applicable), for example, redundant network and Fibre Channel cards, dual HMCs, and so on.
- HACMP can be used to provide automated failover between LPARs. These LPARs can be located within the same physical server, or in different physical servers; this allows for dual or multi-site implementations.

- PLM dynamically reacts to load changes and adds or removes resources as needed. The load changes can be triggered by application demands, or by changes in HACMP topology, such as failure of one cluster node followed by application redistribution on the remaining nodes. As a consequence of the cluster change, PLM redistributes resources and keeps the response time of the critical applications within accepted (and acceptable) limits.
- Monitoring; if hardware, system or application availability is critical to your business, you need to ensure proper and reliable monitoring to protect the availability. An HMC provides the Service Focal Point application which delivers centralized hardware monitoring for all managed servers. There is an optional “Call Home” function which can automatically raise Hardware Problem Tickets to IBM Support. AIX provides RSCT-based monitoring for free, however you will need to implement and configure such monitoring based on your requirements.
- Regular scheduled maintenance; ensure your environment is current with regular scheduled maintenance. For example, a rolling schedule can be used to apply security APARs, PTFs, AIX Technology Levels and microcode updates as required. Keeping AIX current with fixes and updates mitigates against known exposures and issues. It is also necessary to keep IBM System p5 System firmware current when using either HMC or IVM. As new HMC/IVM versions are released they have a tendency to require the latest firmware levels as a prerequisite.

1.2.4 Improve machine utilization

With traditional standalone servers you had the limitation of static resources. If your application did not require or use all the CPU and memory resources, the server would be under-utilized. While it is arguably better to have free resource (which my be need for peak loads) under-utilization means you do not exploit your investment to its full potential (thus increasing TCO).

By allowing multiple Operating instances to be hosted on a single machine, LPARs provide the potential to address resource under-utilization, as described in 1.1.2, “Logical Partitioning (LPAR)” on page 2. LPARs allow more efficient use of the overall physical resources - improving the return on investment (ROI[1]).

Micro-Partitioning, took utilization improvements a step further by increasing the granularity of resource allocation (see 1.1.4, “Micro-Partitioning” on page 3). This technology, combined with DLPAR allows intelligent resource allocation. However this only provides a solution for CPU and RAM utilization.

[1] ROI measures how effectively a company uses its capital to generate profit; IT infrastructure is also part of a company’s capital investment, thus impacting the overall ROI.

If your environment uses SAN storage and network connectivity, then it involves physical I/O cards. However, not all applications hosted in your environment require the full bandwidth provided by the storage or network cards. Moreover, each application has its own usage profile; some will be constantly busy, others will have peaks of activity. To leverage the I/O bandwidth, Virtual I/O has been introduced.

Virtual I/O Server (VIOS) can share physical devices between LPARs. thus allowing you to make better utilization of the physical hardware, when combined with Micro-Partitioning and DLPAR.

Using technology to increase the overall utilization of your server will improve your return on investment.

1.2.5 Business processes compliance

Many companies are using forms of IT process methodology for the overall IT Management as ITIL® or CobiT[2]. These management and process architecture guidelines are usually general, and there is a strong need for particular tools enabling the use of these methodologies in working environments.

Technologies and tools, such as CSM, HACMP, Advanced Power Virtualization (APV) can help implement the methodology. Table 1-1 shows how these management products map to the IT process methodologies.

Table 1-1 Mapping IT Process Methodologies and corresponding tools

Process/Methodology	Tool
Incident Management	CSM monitoring via RMC
Change and Deployment Management	CSM, HMC management tools
Release Management	NIM software distribution and management
Configuration Management	CSM hardware inventory, NIM software inventory
Service Execution	APV, PLM virtualization and load optimization capabilities
Availability Management, SLA Management	HACMP Clustering solution and availability measurement

[2] http://www.isaca.org/Content/NavigationMenu/Members_and_Leaders/COBIT6/COBIT_Mapping1/COBIT_Mapping.htm

There are also specialized enterprise solutions and featured products, such as IBM Tivoli Availability Process Manager or IBM Tivoli Unified Process Composer. In many cases are those products use as the endpoint technologies tools, such as NIM, HMC, HACMP, and so on.

1.2.6 Resource flexibility

The technology implemented in IBM System p5 hardware provides immense flexibility for resource management in terms of resource sharing:

- With Advanced POWER Virtualization and/or LPAR technologies there is limited need for requesting new hardware with every new application or development environment. Existing hardware can be used to create another instance of the operating system (Linux, AIX) with given resources (CPU, RAM, Storage, Network connection) online.
- Micro-partitioning and VIOS are perfect for functional development environments. In instances where performance is not the focus, mirco-partitions can be used to provision small amounts of resources for such environments.
- Request for increased CPU and memory resources can be fulfilled without any downtime.

The flexibility is achieved by using technologies, such as Advanced POWER Virtualization, LPAR, DLPAR and VIOS.

You should keep in mind that the flexibility is confined within the limits of the physical server. While virtualization and resource granularity provide a more intelligent method for resource allocation and abstraction, they can only leverage existing resources.

2

Planning and concepts

This chapter discusses elements which need careful consideration during your planning phases. We illustrate the fact that multiple solutions can fulfill your requirements. Only through contextual understanding can you appreciate that certain solutions will be better suited to your particular environment. Additionally, we discuss migration from existing RS/6000 SP or IBM System p4 deployments; noting that you need to understand the demands your application currently makes of the existing environment, before planning a migration.

We also build on the information in Chapter 1, providing more detailed descriptions of CSM and RSCT.

This chapter discusses the following:

2.1 Planning considerations

This section provides information for planning your IBM System p environment for hardware, software and connectivity.

2.1.1 Machine type and models

On the IBM[1] Web site or presentations you can find many sources about the IBM portfolio including features reports. We outline some of the less obvious machine features which can influence the overall planning process:

Machine types

For small environments you can use entry-level servers with 1-4 cores (processors). With the introduction of IVM technology, you can now use System p virtualization without an HMC, thereby reducing costs. If the workload grows beyond the physical hardware, there is limited scalability. However, for smaller deployments the entry-level servers are very cost-effective. Typically entry-level servers include three-year standard hardware maintenance. These servers are customer-installable, therefore initial hardware setup and base microcode upgrades are customer responsibilities.

Mid-range machines with 4-16 cores (processors) combine the features of the entry-level servers with greater scalability and expansion potential. While you may not require the increased power of a mid-range server, you may require the increased I/O capabilities (number of I/O slots, bandwidth) and RAS features to meet your expectations.

The scalability of large enterprise-level machines covers higher demands, but there is an associated cost to such deployments. High-end machines are installed by IBM specialized personnel, use redundant service processors and mandate HMC management. For large size environments, sharing and consolidating resources can result in lowered expenses than using numerous smaller machines.

In certain cases an IBM BladeCenter -based solution is a valid alternative to traditional IBM System p servers. In a single chassis you can mix blades hosting a range of operating systems: AIX, Linux, and Windows®. There is also a range of available CPU technologies: Intel®, AMD and POWER. IBM BladeCenter hardware has integrated SAN and LAN infrastructure and this can result in saving infrastructure costs.

[1] http://www.ibm.com

High performance computing applications require specific configurations. Such environments use clusters of IBM System p5 or System x servers, or special purpose designs, such as IBM System Blue Gene Solution.

2.1.2 Advanced POWER Virtualization planning

There are many ways to leverage the features of Advanced POWER Virtualization. Careful planning and thought should be used for your deployments. Some of the following considerations will only apply if you choose to leverage those features.

- Micro-partition and virtual CPU:

  Depending on your application characteristics, additional effort and extra care maybe required for planning micro-partitions. Depending on the type and design of application, the LPAR configuration could impact the operational application performance. In addition to the required amount of processor resource (units of capacity), thought should also be given to capped and uncapped configurations - and the implications of each.

  Another concern is Virtual processor configuration. Virtual processors are the whole number of concurrent operations which can be utilized by an Operating System. Some applications will benefit from concurrency of execution, others will prefer raw power. By default, the HMC will assign as many Virtual CPUs required to fulfill the required Processor units of capacity; for example an LPAR configured with 0.5 units will be allocated one virtual processor; an LPAR with 2.5 units will be allocated three virtual processors. The default allocation can be changed if required. Testing may be the only way to assess the impact of such a configuration.

- Virtual I/O Server LPAR sizing:

  As previously outlined, VIOS provides the ability to virtualize both network and storage to LPARs. While the allocation of physical I/O resources should be a simple task, CPU and RAM sizing will be a potential challenge. The number of client LPARs served and the required throughput will apply load on the VIOS itself. Insufficient or incorrectly sized VIOS resources will impact the responsiveness and capability of the VIOS and its clients. Due to the large number of potential configurations a best practice is to configure the VIOS with dedicated resources, then test by applying an actual workload and monitor the individual resources and overall performance as workload increases. Testing and monitoring will determine the optimum configuration for your individual deployment.

- VIOS implementation:

  Careful thought should be given to the VIOS architecture itself. For example, appreciate the implications of using a single VIOS in a given deployment, as this is a single point of failure. Also you should consider dual VIOS to balance or distribute I/O demand. It is more efficient to virtualize physical volumes, logical volumes or storage pools? Will you be serving HACMP cluster nodes with VIOS resources?

Note: For a detailed discussion about Advanced POWER Virtualization planning (including the above topics), refer to Chapter 5 of *IBM System p Advanced POWER Virtualization Best Practices*, REDP-4194.

2.1.3 Integrated Virtualization Manager (IVM)

As previously mentioned in 1.1.9, “Integrated Virtualization Manager (IVM)” on page 6, Integrated Virtualization Manager (IVM) is an optional feature available on certain entry- and mid-level IBM System p5 systems. For certain Customer environments IVM may be a preferable option to an HMC-managed solution. There are both major and minor differences between both approaches. Table 2-1 provides a detailed comparison between the two alternatives.

Table 2-1 IVM and HMC comparison

	Integrated Virtualization Manager (IVM)[a]	**Hardware Management Console (HMC)**[b]
Physical footprint	Integrated into a hosted LPAR on the server	A desktop or rack-mounted appliance
Installation	Installed with the VIOS (optical media or network). Pre-install option available on some systems.	Appliance software is pre-installed. Reinstall via optical media or network is supported.
Managed operating systems supported	AIX 5L and Linux	AIX 5L, Linux and i5/OS®
Virtual console support	AIX 5L and Linux virtual console support	AIX 5L, Linux and i5/OS virtual console support
User security	Password authentication with support for either full or read only authorities	Password authentication with granular control of task based authorities and object based authorities
Network security	-No integrated firewall -Web server SSL support	-Integrated firewall -SSL support for clients and for communications with managed systems

	Integrated Virtualization Manager (IVM)[a]	Hardware Management Console (HMC)[b]
Servers supported	System p5™ 505 Express System p5 510 and 510Q Express System p5 520 and 520Q Express System p5 550 and 550Q Express System p5 560Q Express p5 510 and 510 Express p5 520 and 520 Express p5 550 and 550 Express OpenPower™ 710 and 720 BladeCenter® JS21	All POWER5™ and POWER5+™ processor-based servers: System p5 and System p5 Express p5 and p5 Express OpenPower i5
Multiple system support	No. Only the single physical server on which the IVM is hosted	Yes. One HMC can manage multiple servers[c]
Redundancy	No. One IVM per server	Multiple HMCs can manage the same system for HMC redundancy
Maximum number of partitions supported	Firmware maximum	Firmware maximum[d]
Uncapped partition support	Yes	Yes
Dynamic Resource Movement (dynamic LPAR)	DLPAR for memory and processors of managed partitions	Yes - full support
I/O Support for AIX 5L and Linux	Virtual optical, disk, Ethernet, and console	Virtual and Direct
I/O Support for i5/OS	None	Virtual and Direct
Maximum # of virtual LANs	Four	4096
Fix/Update process for Manager	VIOS fixes and updates	HMC corrective service and release updates
Adapter microcode updates	Inventory scout	Inventory scout

	Integrated Virtualization Manager (IVM)[a]	Hardware Management Console (HMC)[b]
Firmware updates	VIOS firmware update tools (not concurrent)	Service Focal Point with concurrent firmware updates
I/O Concurrent Maintenance	VIOS support for slot and device level concurrent maintenance via the diag hot plug support	Guided support in the "Repair and Verify" function on the HMC
Scripting and Automation	VIOS command line interface (CLI) and HMC compatible CLI	HMC CLI
Capacity on Demand	No Support	Full Support
User Interface	Web browser (no local graphical display)	WebSM (local or remote)
Partition Load Manager (PLM)	No Support	Full Support
Workload Management (WLM) Groups Supported	One	254
LPAR Configuration Data Backup and Restore	Yes	Yes
Support for multiple profiles per partition	No	Yes
Serviceable event management	Service Focal Point Light - Consolidated management of firmware and management partition detected errors	Service Focal Point support for consolidated management of operating system and firmware detected errors
Hypervisor and service processor dump support	Dump collection with support to do manual dump downloads	Dump collection and call home support
Remote support	No remote support connectivity	Full remote support for the HMC and connectivity for firmware remote support

[a] IVM Version 1.3.0

[b] HMC Version 4.5.0 and higher

[c] At the time of writing, a single HMC will support up to 48 non p590/595, or 32 p590/595 servers.

d. At the time of writing, a single HMC will support a maximum of 254 LPARs across the managed systems.

Due to the inherent differences between IVM and HMC, there are some operational aspects you should be aware of. These are correct at the time of writing:

- IVM is a single point of failure in a managed system (granted, the physical box itself is also a single point of failure). IVM is also the VIOS and the hardware control point. At the time of writing redundant IVM LPARs are not supported. Also, if you shutdown the IVM LPAR, you will not be able to restart it remotely.
- Software upgrade from Version 1.2.0 to 1.3.0 requires a reboot of the IVM LPAR; If you reboot the IVM LPAR this will impact hosted clients as they will loose access to any virtual devices. Also the LPARs will remain active, which may present potential application data exposure. Thus we recommend you to shutdown client LPARs during IVM maintenance.
- If you shutdown (but not restart) the IVM LPAR, IVM will also shutdown the client LPARs.
- It is possible to allocate a storage device to multiple clients (that is, shared disk between HACMP nodes). At the time of writing it is only possible to define such a configuration via the CLI, nevertheless the IVM GUI will correctly display the configuration.
- Due to the relationship between IVM and the IBM System p5 hardware a simple migration path is not possible between an IVM-managed and HMC-managed system. For example there is no available mechanism to migrate LPAR definitions between the two technologies.

Note: For a more detailed discussion of Integrated Virtualization Manager (IVM), refer to *Virtual I/O Server Integrated Virtualization Manager*, REDP-4061.

2.1.4 Networks

Production network environments are usually constrained by security and/or organizational policies.

The network infrastructure often imposes various constraints upon the network configuration of the systems integrated in your environment. You should clarify every detail that might interfere with the communication between cluster nodes and/or cluster nodes and clients.

We present some issues that could have a functional impact on your environment.

- **Ethernet adapter connection settings**

  In general, if the adapter supports, it should be configured to not auto negotiate connection characteristics, rather to always run at the desired speed and duplex value. The switch port to which the adapter is connected should be set to the same fixed speed and duplex value.

- **EtherChannel**

  Using EtherChannel imposes some restrictions including:

  - The aggregated links must go to the same switch.
  - The network switch must be configured to identify which ports are used for EtherChannel.
  - Mixing adapters of different speeds is not supported.

- **VLAN**

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

- **Network latency**

  If cluster nodes are connected by a potentially slow network, such as a geographically dispersed network, you should also take into account network latency. You should adjust heart beat interval (interval in seconds between heartbeats) and heart beat sensitivity (number of missed heartbeats) so that no accidental delay would induce unwanted events in the cluster.

- **Address Resolution Protocol (ARP)**

  If you use alternate hardware address feature, when HACMP moves IP labels it can move a locally administered MAC addresses (LAA) as well. In this case, the ARP cache remains correct.

  If you use IP address takeover without Hardware Address Takeover, the MAC address associated with the IP address from the ARP cache may become outdated. All TCP/IP systems, such as routers or client systems that reside on the same subnet as the cluster nodes must have their Address Resolution Protocol (ARP) cache updated. HACMP provides means to overcome this problem.

  You must also take into account some factors which could affect ARP cache accuracy, such as:

  - Delays caused by ARP cache Timeout.
  - Proxy ARP service running in the collision domain and providing incorrect data.
  - Spanning tree algorithm running on network switches.
  - Clients and network appliances that do not support promiscuous mode.

- **Route definition and persistence**

  Ensure that all necessary routes are defined on all cluster nodes. Every cluster node should be able to route IP datagrams properly even after the failure of a network interface, cluster node, or switch.

  For instance, 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 that 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.

- **IP Filtering**

  If the network contains devices such routers or firewalls that can either filter, encapsulate or convert various packets consider the following:

  - ICMP ECHO may be used for ARP cache updating.
  - UDP broadcasts and other packets are sent between cluster nodes.
  - The RSCT Topology Services subsystem uses several types of communications:
    - UDP port numbers for intra-cluster communications between Topology Services daemons.
    - UNIX domain sockets for communication between Topology Services Clients and Topology Services daemon.

  It is usually safe to use bridges, hubs, and other passive devices that do not interfere with IP packets.

Note: Always test your network speed to see whether it meets general criteria. If your settings will be wrong, the network can in some way work, but you can get unpredictable errors during for HACMP synchronization, and so on.

2.1.5 Hardware Management Console (HMC)

As mentioned in 1.1.5, “Hardware Management Console (HMC)” on page 4, an HMC provides hardware control of the managed systems. In this section we highlight a couple of operational points and discuss network architecture implications.

- We recommend that you implement and use procedures to maintain your HMCs with updates and corrective services. At the time of writing the current versions are Version 3.3.7 for IBM System p4 HMCs and Version 6.1.0 for IBM System p5.

Note: You can subscribe for e-mail notifications of updates. For more information, visit the IBM Fix Central site:

http://www-912.ibm.com/eserver/support/fixes/fixcentral

- When installing IBM System p5 servers, ensure the Central electronic Complex (CEC) is appropriately cabled to at least one operational HMC prior to cabling the CEC to a power source. If power is provided prior to cabling the CEC to an HMC, then the DHCP client on the CEC will timeout and revert to the default static address, resulting in CEC will not be detected by the HMC. The resolution to this problem depends on machine type.

  Note: For resolution on this and other HMC configuration issues, refer to the document entitled "Troubleshooting HMC setup" within the IBM Systems Hardware Information Center Web site:

  http://publib.boulder.ibm.com/inforcenter/eserver/v1r3s/index.jsp

- Consideration should be given to the networks required by an IBM System p5 HMC. There are a number of potential scenarios and you need to be able to evaluate the implications of every option. There are many variables, which individually and in combination may result in a number of ways to correctly design and implement your environment. Consider the following:
  - How many physical IBM System p5 servers will you be deploying. This will dictate the minimum number of HMCs you require.
  - For logistical or security reasons you may require additional HMCs to host smaller groups of servers. For example, grouping CECs by network zone, or hosted application.
  - Will you be deploying dual HMCs for redundancy? If so, we recommend not to cable both into the same network switch, as this moves the single point of failure from the HMC to the network. Implementing a single switch could also cause unpredictable results if both HMCs are providing DHCP services. For more information about redundant HMCs, refer to *IBM System p5 Approaches to 24x7 Availability Including AIX 5L*, SG24-7196.
  - The network configuration for an HMC needs to accommodate all the attached managed systems. If the HMC is acting as a DHCP server, its configured DHCP IP range needs to be large enough. A similar consideration exists if your HMCs, CECs and LPARs are all hosted on the same network (VLAN) or subnet (IP segment).
  - If you have enough CECs to employ multiple HMCs, will you be using separate networks (one per HMC) or a shared network? Considering the

preceding implications, which is the most appropriate option for your environment?

  - Will you be using CSM with the Cluster-Ready Hardware Server (CRHS) feature? If so, there are a number of documented prerequisites and environment requirements which must be adhered to. Implementing CRHS mandates a single service network and DHCP server. As the single network is shared by the CSM Management Server, HMCs and CEC connections, you should decide carefully which machine will be the DHCP server.
  - Also when considering any of the preceding points, take into account also potential future growth. Careful planning will avoid disruptive re-configuration in the future.

> **Note:** For additional information about HMC networking options, refer to the document entitled “Type of HMC network connection” within the IBM Systems Hardware Information Center Web site:
>
> `http://publib.boulder.ibm.com/inforcenter/eserver/v1r3s/index.jsp`

2.1.6 Administration, backup strategy, monitoring

When planning deployments, especially large environments, it is convenient to plan and prepare in advance. Standardization facilitates simpler administration and problem resolution. It also reduces time and costs involved to expand the system over time. Monitoring and backup policies should be part of the planning of a new deployment. While this may seem obvious, it is common that these system management aspects are overlooked or mot given proper attention. The following illustrates some example strategies:

- Sample AIX operating system backup scenario:
  - AIX operating system backups are stored into a single shared NFS file system.

    The NFS definition is present within the operating system image which is used to install new LPARs, therefore no additional administration tasks are needed when restoring operating system or installing a new LPAR.
  - The exported NFS directory resides on the NIM server.

    This simplifies the restore process, as the mksysb images are already available on the NIM server; they can be defined to and used by NIM without additional restore activities.

  - Name of the backup image convention

    /path_to_nfs_directory/<hostname>.bos. This naming standard allows you to create unified procedure for defining the NIM resources when restore is needed, or as the base procedure when adding a new LPAR to the cluster. This makes easy to understand which LPAR is the source for the mksysb.
  - Backups are scheduled via cron using the **`dsh -f 1`** command so that backups are run serially across the cluster nodes. This reduces the risk of the NFS exported directory and/or the network being overloaded - but will increase the overall backup window.
  - As a complement to the backup process itself, monitoring will be added to check if all of the CSM nodes defined are performing their scheduled backup to the NFS exported directory.
- Filesystem-level TSM backups:
  - We recommend the use of a standard TSM Client configuration template. This template is configured in such a way that all certain standard filesystems are required to be backed up by default (/usr, /var, /home etc.), whereas others are excluded by default (/proc, /dev etc.).

    If a server has additional filesystems, these can be added as required to the local TSM configuration. The TSM Client also provides a scheduler which should be used to automate regular backups. Having a standard practice for using TSM will ensure correct and regular backups are taken. Using a standard template as a default will ensure that at least standard filesystems will be backed up.
  - You should also set up a TSM scheduler for HACMP related filesystems (stored on shared disks and only accessible on the node that holds the resource group).

    For HACMP shared filesystems you should create special HACMP applications servers, which start/stop the TSM scheduled backups on the appropriate HACMP node. This type of configuration allows you to backup/restore files even when HACMP takeover occurs.
- Sample monitoring scenario:
  - It is convenient to create a basic set of monitors which apply for all nodes in the cluster. Such a monitoring profile can be applied to a newly created LPAR, by default. Having a standard distribution means that you do not need to create a custom profile for each LPAR.
  - For example, in our scenario we have created the base set of monitors for: standard AIX filesystems, CPU load, AIX error log, process status (cluster manager, ssh daemon, sendmail) and HACMP cluster status.

- Administration using CSM:
  - Due to the relationship between CSM and NIM, any new nodes installed by NIM can be automatically added to the existing CSM cluster. Once registered to CSM, node group membership dictates the monitoring and configuration files distribution. This facilitates that all nodes (within a node group) have consistent configuration and monitoring.

2.1.7 Environmental and external factors

When planning your environment you also have to consider other factors besides performance sizing, as some of those factors can have a significant influence on the overall system design. We want to highlight some of the problems which you may have to face:

- Physical characteristics of the servers (dimensions, weight): Some data centres do not allow the placement of all type of enclosures due to their weight exceeding the maximum allowed in the computer room, or their dimensions exceeding the access door size.
- Electricity consumption: Check your power supply lines, UPS, circuit breakers ratings against IBM hardware requirements.
- Air conditioning: Excessive computer generated heat may require upgrades to the air conditioning system. One good estimation rule: most of the power consumed by IT equipment turns into heat, thus the amount of power consumed by the IT equipment can be considered the base of calculation for the air conditioning cooling capacity.
- Network infrastructure: Check if you have sufficient LAN, WAN ports and if they are supported with System p5 adapters.

> **Note:** Virtualization using VIOS can save Infrastructure SAN and LAN ports. Also if third party software licenses are required for accessing non-IBM storage arrays, the licenses are only needed for the VIOS partition not for client micro-partitions.

You can find more information about the Systems p5 requirements in the Fact and features report mentioned in 2.1.11, “Capacity planning and sizing” on page 33 or on the IBM product Web site. Planning is discussed in more detail in *IBM System p5 Approaches to 24x7 Availability Including AIX 5L*, SG24-7196.

2.1.8 High availability level

There are several reasons to add elements of HA into your design. Hardware component failure tolerance is only one reason, and the probability of such failures (and their impact) needs to be considered and addressed during the planning phase of your environment. It is very difficult to offer specific recommendations, as much will depend on your environment and individual requirements. From previous experience, the following list highlights common reasons to implement and exploit HA elements:

- Hardware or Software maintenance on part of the infrastructure (for example SAN or LAN hardware)
- Software maintenance on a subset of the cluster nodes; for example Operating System APARs or patches
- Single I/O card or hard disk failure
- Administration error, resulting in operating system (and therefore application) failure
- Application failure
- Operating system problem, resulting in application failure
- Total physical IBM System p5 failure or Service Processor failure
- Infrastructure site failure or outage, for example, power outage or natural disaster

While planning an HA solution or considering the decision to implement HA elements, you should evaluate the price of service downtime. In some cases this will be a known metric as your company may have direct contracts for unplanned downtime and agreed schedule for planned downtime. However in new deployments or non-critical environments the cost of unplanned downtime might not be as obvious. Using the previously listed points as guidance, you should be able to quantify downtime impact, and use this for in the decision making process. There are also other considerations, such as company reputation and overall customer satisfaction which can be taken into consideration.

There are many publications describing how to protect your systems against the component failure and we refer to them later in this chapter. Table 2-2 outlines the steps for designing an HA infrastructure.

Table 2-2 HA Protection features

Outage type	**Possible solution**
Hardware or software Infrastructure Maintenance	Duplicated I/O adapter cards, redundant Power supplies, using redundant VIOS servers, in some cases HACMP software and redundant CECs
Operating system software Maintenance of the node	HACMP cluster inside one or multiple physical CECs

Outage type	Possible solution
Single I/O card, hard disk or power break failure	Redundant IO adapter cards, disks, and/or using Ether channel or MPIO features. Using redundant VIO servers, disk mirroring, redundant Power supplies
Application or operating system failure, Administration mistake	HACMP cluster deployed onto one or multiple physical CECs
Single CPU or memory chip failure	Using IBM System p5 native features as Chipkill™ technology or CuOD, redundant CECs and HACMP in more critical environments
Overall IBM System p5 failure	Redundant boxes + HACMP software within one or more sites
Natural disaster or site failure	Redundant boxes + HACMP software with multiple sites

Data mirroring can be implemented by using AIX Logical Volume Manager, Disk storage mirroring, such as PPRC or HAGEO capabilities.

Virtualization brings potential of increased high availability levels. Using VIOS, Etherchannel, MPIO and other IBM System p5 features and technologies, system resilience and availability can be improved by multiplying components and distributing them to the target OS. For example, if one of the CPU in a shared processor pool fails, it will be de-allocated by IBM System p5 RAS functions. The overall system load is re-distributed according to the policy, or CuOD CPU is added to the shared pool to replace the failing unit.

Almost all components can be redundant on a physical IBM System p5 machine. On some models it is also possible to configure redundant service processors to keep the server running should one frame service processor (FSP) fail. Note that machine microcode release upgrades are applied on both FSPs at the same time. Additional network configuration is required as redundant FSPs require specific DHCP setup on the HMCs. For specific information, refer to *IBM System p5 Approaches to 24x7 Availability Including AIX 5L*, SG24-7196.

For additional HACMP information and reference materials, refer to:

- *IBM System p5 Approaches to 24x7 Availability Including AIX 5L*, SG24-7196
- *HACMP 5.3, Dynamic LPAR, and Virtualization*, REDP-4027

The official HACMP documentation can be found at:

http://www-03.ibm.com/systems/p/library/hacmp_docs.html

2.1.9 Hardware and software maintenance

A significant part of the overall IT infrastructure price is the cost of ongoing hardware and software maintenance.

Hardware maintenance

Hardware maintenance is a multiple purpose service with many options. A complete list can be obtained from your IBM representative. It is typically purchased with the hardware itself, and renewed periodically. Some of its main features are:

- Repairing or replacing failed parts using field replaceable units (FRU).
- Consultation services including problem determination.
- Physical machine setup and maintenance including microcode updates for some type of the machines, which are usually called IBM setup machines.
- Periodical system diagnostics and/or log analysis

IBM offers are several levels of hardware maintenance, with adjustable coverage parameters:

- Fix time: You can choose non-guaranteed or guaranteed fix time. Guaranteed fix time means that in a case of failure, parts are exchanged within a guaranteed time. Options exist for the required fix time, with an associated price. Example options are 8 hours, 24 hours or next business day.
- Time of day interval (mode): This is the period of time for which the services are provided, for example mode 7x24 means that the service is provided everyday including weekends and holidays, covering all 24 hours, every day. Other modes can offer only business hours, excluding holidays and weekends.

  > **Note:** If you only have limited mode maintenance and require assistance of the service engineers outside the hours covered by your agreement, support can still engage at additional cost (time and material based fee).

- Contract duration: Maintenance is purchased for given time periods; typically one or three years.

Some machine types have already included hardware maintenance in their purchase price. Entry and mid-range machines have 3-year maintenance with a base mode and fix time included in their base price. High-end machines usually have 1-year maintenance included in their purchase price.

Note: The previous statement may not apply to all contracts. For special terms and conditions, negotiated individually, service coverage may not be included in the purchase price.

Maintenance is provided by trained IBM technicians which can dramatically help decrease the overall downtime. In some cases, using HA infrastructure can compensate and substitute the requirement for more comprehensive maintenance modes.

Software maintenance

Software maintenance provides some of the following (not limited to) features:

- Free download or CD-ROM ordering of the new software releases
- Fixes (PTFs) and maintenance packages download
- Telephone support for problem determination
- Problem fixing

We want to distinguish between the software maintenance for the IBM Passport Advantage® based software, such as DB2®, Tivoli or WebSphere® and the base operating software, such as AIX, HACMP or CSM. Base software is much more adherent with IBM System p hardware, as it is typically part of the product order. Some software is not transferable between the machines (such as AIX operating system) as they are linked with the machine serial number.

Do not forget to include the price of the hardware and software maintenance while planning the overall deployment and ongoing costs.

Note: The previous statements regarding hardware and software maintenance can change in the future. Please ask your IBM representative for current offerings.

2.1.10 Financing and leasing possibilities

IBM offers remarkable financial offerings for purchasing hardware and software. Financing for three years with monthly payments can be very convenient. Leasing maybe another way to acquire the hardware you need. For more information, please ask your local IBM representative.

2.1.11 Capacity planning and sizing

There are numerous situations where it is necessary plan resource capacity, we describe some fundamental cases.

Migration from legacy IBM hardware, workload unchanged

While this may not be such a common scenario, we wanted to demonstrate the problems. For planning purposes it is possible to translate the performance metric of your legacy hardware to current units; this allows you to estimate the power of new technology with regard to your known workload. IBM performance units called *rPerf* can help you to compare current hardware against your legacy environment. Please note that the transformation is not so trivial and you have to consider several factors:

- Performance varies not only by CPU type, frequency or quantity, but there are also suitable differences caused by using different memory modules, amount, memory bus width, and overall architecture (for example, number of CPU cards).
- Performance can vary while using Simultaneous Multi-Threading (SMT). This is available on AIX 5L V5.3 and higher; performance difference can be 30% or higher - depending on type of workload. For example database applications usually benefit using SMT while some of the HPC applications do not. The best recommendation would be to perform a benchmark test using your own application or ask your IBM representative for guidance.
- Consider using shared or virtualized resources, such as a shared processor pools. There are certain cases when your application performance peaks are staggered over time or are completely exclusive. In such situations you can leverage the benefit of IBM System p5 virtualization.

> **Note:** Always contact your IBM representative to validate your configuration and sizing considerations. rPerf performance data is very helpful but may not represent performance with regard to your applications.

A recommended source of information for IBM System p features including rPerf and other benchmarks are the “Fact and features reports*”* published on the IBM Web site:

http://www-03.ibm.com/systems/p/hardware/factsfeatures.html

Migration from old hardware, workload changes during time

In this case you can use the same methods as mentioned in the previous paragraph, with the following note.

Note: Applications have also have a life cycle and growth patterns. Carefully monitor and record the following:

- Performance load of the CPU, memory and I/O traffic as shown in Figure 3-26 on page 116
- Configuration data about the CPU, memory, I/O cards assignment

If you have to estimate proposed Hardware growth for 3 or more years and you do not know what is your application performance load footprint, it is very difficult to predict the expected capacity requirements over time.

There are also other indirect methods for estimating application growth over time. For example:

- Size of your database
- Backup duration
- Duration of user response times, for typical tasks (for example in SAP® system)

If you keep track of those metrics or you have access to historical data, you have at least the base line for future growth.

As we use virtualization technologies where you can dynamically move CPU and memory, or to use shared processor pool, remember to keep track of your resource usage and configuration. These are required for your planning.

As the part of the planning procedure you should also consider application specific changes:

- Plans for additional production system instances, test or consolidation environments.
- Plans for major application changes, such as database migration to UNICODE or new version or release.
- Plans for rapid local application development or application customization. These changes usually significantly affect the overall sizing considerations. It is good practice to know for example the portion of the local developed programs in the overall SAP base and keep track of the sudden changes (change management).

New applications with possibility to estimate the workload

Sizing tools exist for many applications (SAP, WebSphere Application Server, and so on). Please contact your IBM representative for more information and assistance.

New applications with unknown workload

There are more “chances” for inaccurate sizing in this category. Some common examples:

- There are known sizing procedures but it is impossible to get historic and current information.
- Applications are new and there is limited experience with their production utilization.
- Performance investigation is dependent on application development, which is not finished (or worse, has not even started yet).

Moreover, it is impossible to precisely estimate workload without extensive analysis and you have to usually rely on approximations. Virtualization technologies give you chance to eliminate the overall risk of inaccurate workload estimations by consolidating more applications on a single physical machine and adjusting the resource usage according of the actual needs.

Other cases

There are many other cases, such as sizing HPC applications where it is possible to test your workload in an IBM benchmark center. However, such cases are beyond the scope of this publication. Please contact your local IBM representative for more information.

Capacity benefits from Advanced POWER Virtualization

Advanced POWER Virtualization has several benefits which cannot be achieved by other technologies:

- Cost saving: Having less physical resources and sharing them saves money, if the situation allows this. Note that using less I/O devices, such as Ethernet or FC adapters also reduces the infrastructure costs.
- Flexibility: Unplanned demand requirements can be fulfilled by virtualization with minimal, if any, costs.
- Performance boosts: As some situations may require peak brute power, consolidated server infrastructure is a good solution.
- Consolidation: There are numerous disadvantages to managing many small machines. By reducing the number of physical servers and simplifying administration, consolidation reduces or removes these disadvantages.
- High availability: Even though you can achieve certain HA levels for I/O, CPU and RAM resources while implementing in VIOS features, such as Etherchannel, dual VIOS or using CUoD features, note that the whole CEC (or the OS) is still a single point of failure.

Advanced POWER Virtualization disadvantages

Virtualization adds another layer between hardware and the hosted applications. This leads to additional considerations:

- Uncertainty: It can be difficult to estimate how many micro-partitions with specific workload can share a certain amount of physical resources. There are customer cases successfully running 40 micro-partitions with Web-hosting and WebSphere applications on a 16-way machine with dual VIOS. However, underestimation of capacity requirements can result in VIOS consuming more physical resources (CPU in particular) than the client partitions, which is undesirable.
- Configuration complexity: Technically diverse environments require detailed understanding and planning. Combining new technologies requires understanding of both to appreciate mutual implications and impacts. HMC, VIOS and PLM require additional planning and configuration.
- Security complexity: Depending on the local security policies it maybe necessary to review the sharing (or virtualization) of the physical resources; for example, some environments may not allow “shared” traffic through one physical Ethernet adapter. Those problems can be solved by using dedicated adapters or by using separate VIOS for different network zones.

2.1.12 Cost: Price versus performance

Value is both relative and subjective. The purchase price, running costs, capacity and functionality of that server all can be defined as value, depending on the viewpoint. All appropriate viewpoints should be considered when evaluating options. The decision will usually result in a balance between price and performance.

We have previously mentioned that is it common to have several solutions when considering new purchases. While all solutions may be capable of performing the required role, additional factors need to be considered. Table 2-3 outlines some of these viewpoints.

Table 2-3 Evaluation considerations

Viewpoint	Description
Environmental requirements and considerations	Floor space and loading, power, temperature and ventilation
Server capacity	Available CPU/RAM configurations, maximum number of LPARs, storage/IO capacity

Viewpoint	Description
Future expansion	Ability to upgrade and add CPUs/RAM? Storage and I/O expansion, are external I/O draws supported for your choices?
System Administration	Options for Hardware management? Can new technology be integrated into your existing management infrastructure? Will your administrators require education? Will new technology be familiar or alien to them?
Performance	Can the server fulfill your current and future throughput requirements? Can it provide the CPU, Storage and Network bandwidth requirements?

Factoring the preceding considerations, the cost of purchase and ongoing maintenance needs to be evaluated in the context. A cheaper alternative might not have the required expansion capabilities, whereas a larger solutions may not physically fit your data center.

Sizing example considering price/performance

To illustrate the point, we will compare two IBM System p5 models: IBM System p5 550 and IBM System p5 570.

Notes:

- The following examples illustrate correct pricing at the time of writing (October 2006). The pricing represents the cost for the given configuration, in US Dollars for the US market.
- Also, prices can vary by geography and are subject to change without notice. Other configurations are available.

An IBM System p5 550 can be utilized in three modes:

- Non-managed, system runs as a full system partition utilizing all resources. LPAR, DLPAR and Advanced POWER Virtualization features are not supported in this configuration.
- HMC-managed, where an Hardware Management Console (HMC) is used to configure and administer the IBM System p5 550 (and potentially other servers). From the HMC LPARs can be carved and managed.
- IVM-managed, where an Integrated Virtualization Manager (IVM) LPAR is used to configure and administer the IBM System p5 550 in isolation. Using IVM, hosted LPARs can be carved and managed.

It is possible to switch between management style, however this requires server re-configuration and possible reinstallation.

> **Note:** Both HMC-managed and IVM-managed provide support for Advanced POWER Virtualization. Table 2-1 on page 20 illustrates the difference between the two alternatives.

Table 2-4 details five cumulative IBM System p5 550 configurations, which progressively add features or options.

Table 2-4 Example IBM System p5 550 configurations

	System	Description	List Price (USD)
1	p550	4-way 2.1Ghz CPU, 16GB RAM, 4x73GB 15k SCSI, 2x4Gb Fibre adapters, 2x 1Gb Ethernet adapters, Rack[a], AIX 5L V5.3[b]	$50,259
2	p550 + APV	As per #1 plus APV[c]	$54,131
3	p550 + APV + HMC	As per #2 plus HMC[d]	$62,025
4	p550 + APV + HMC + CSM	As per #3 plus CSM[e]	$62,761
5	p550 + APV + HMC + CSM + MS	As per #4 plus MS[f], 1xp51A. 2-way 2.1Ghz CPU, 32GB RAM, 4x73GB 15k SCSI, 2x1Gb Ethernet adapters, AIX 5L V5.3, rack-mount feature	$78,722

a. Rack Model T00, with all doors and Power Distribution Unit components
b. AIX 5L V5.3 Media and appropriate software licenses per CPU
c. Advanced POWER Virtualization feature, licensed per CPU
d. Hardware Management Console (HMC), 7310-CR3
e. Cluster Systems Management (CSM), licensed per CPU
f. CSM Management Server (MS), an IBM System p510 with appropriate software licenses

Here are comments to the information in Table 2-4:

- Option #1 is the non-managed configuration.
- Option #2 is the IVM-managed configuration; the APV feature is required to utilize IVM.
- Option #1 and #2 are equally valid configurations if you already have an operational IBM System p5 HMC. The server could be easily integrated into your existing environment. Remember that an HMC can manage several physical machines.

- Observe the difference in price between Option #2 and #3. For a single-server deployment an IVM-managed solution might be satisfactory option; however your requirements may mediate the additional features provided by an HMC.
- The CSM MS in Option #5 is an sample configuration. Different sizing may be required for different size clusters. An existing server of suitable configuration can be used as a MS.
- The cost associated with CSM is a small percentage of the overall cost; this is significant when compared to the centralized benefits it could provide to your environment.

In comparison, Table 2-5 presents an example using IBM System p5 570 configurations. An IBM System p5 570 requires an HMC since it cannot be managed by IVM. You can still run this system as a full system partition, but HMC is still required.

Table 2-5 Example IBM System p5 570 configurations

	System	**Description**	**List Price (USD)**
1	p570 + APV + HMC	8-way 2.2GHz CPU, 32GB RAM, 4x73GB 15K disks, 2x4Gb Fiber adapters, 2x1Gb Ethernet adapters, Rack[a], AIX 5L V5.3[b], APV[c], HMC[d]	$375,700
2	p570 + APV + HMC + CSM	As per #1 plus CSM[e]	$379,836
3	p570 + APV + HMC + CSM + MS	As per #2 plus MS[f]	$395,797

[a] Rack Model T00, with all doors and Power Distribution Unit components

[b] AIX 5L V5.3 Media and appropriate software licenses per CPU

[c] Advanced POWER Virtualization feature, licensed per CPU

[d] Hardware Management Console (HMC), 7310-CR3

[e] Cluster Systems Management (CSM), licensed per CPU

[f] CSM Management Server (MS), an IBM System p510 with appropriate software licenses

Here are comments to the information in Table 2-5:

- Advanced POWER Virtualization is a default option on an IBM System p5 570, and is included in the overall cost.
- While an HMC is mandatory, you can use an HMC from another (existing) IBM System p5.

When comparing the prices in Table 2-4 on page 39 to the prices in Table 2-5 you might ask: “Why would I buy an IBM System p5 570 when I could purchase eight

IBM System p5 550 systems for approximately the same cost as a single IBM System p5 570 system?”. This is where you need to consider a number of important factors:

1. The change in system management introduced through the Logical Partition (LPAR) technology. Hosting multiple operating system instances in one physical box reduces the amount of physical servers required. Reducing the number of physical boxes, decreases the complexity by reducing environment requirements: less boxes, less floor space, less power. Consolidating your environment reduces the efforts required of administering your system. While both servers support LPARs, their available resources will dictate how many LPARs you can host.
2. Individual capacity requirements. Some customers have applications which require small amount of CPU, RAM and I/O, while others may require large combinations of both. While LPARs allow your configurations to be flexible, you still have the ceiling of the overall system capacity. An LPAR cannot span physical boxes. You may have specific requirements which demand the bandwidth of multiple Ethernet adapters, and in such cases virtual I/O would not be suitable, so you would need a machine capable of hosting for large numbers of adapters (PCI slots).
3. Scalability. An IBM System p5 570 has the unique property of building blocks. One to four building blocks can be connected to expand the system. The blocks are managed as one single entity. Each block can host up to 4 CPUs and 128GB RAM. Additional capacity (and blocks) can be purchased and integrated to the system as demand rises. So you could purchase initially a single block system, and expand it over time. The initial extra cost is therefore investment for the future.

Table 2-6 provides a high-level comparison between IBM System p5 550 and IBM System p5 570. This information is considered to be correct at the time of writing (October 2006).

Table 2-6 Comparison of IBM System p5 550 and IBM System p5 570

	IBM System p5 550	**IBM System p5 570**
CPU[a]	2 or 4	2, 4, 8, 12 or 16 (4 per building block)
RAM	1 - 64GB	2 - 512GB (up to 128GB per building block)
PCI-X Slots	5	6 (per building block)
Internal SCSI disk bays	4 standard (4 optional)	6 (per building block)

	IBM System p5 550	**IBM System p5 570**
Internal disk storage	Up to 2.4TB	Up to 1.8TB (per building block)
Optional I/O expansion	Up to eight 7311-D20[b] I/O drawers	Up to eight 7311-D20 I/O drawers[c]
Power requirements	100v to 127v or 200v to 240v	200v to 240v

a. Range of available CPU speeds dependent on IBM System p5 model
b. A 7311-D20 drawer provides seven 64-bit PCI-X slots and up to 12 disk bays
c. Total for an overall IBM System p5 570 system, not per building block

So where combined initial capacity requirements are larger than a single IBM System p5 550, and it is expected that demand will increase over time, an IBM System p5 570 may be a wiser investment, as the cost of ownership will be lower and the server utilization will be higher compared to multiple IBM System p5 550.

Note: For detailed description of the referenced systems, refer to *IBM p5 570 Technical Overview and Introduction*, REDP-9117.

A similar price/feature comparison can be made between IBM System p5 570 and IBM System p5 590/595 servers. A *High-End* server will always cost more than a *Mid-Range* one. For a given model of server, you need to understand what capabilities and capacity it provides, and evaluate this against the price of purchase and ownership. Only then you can select which models would be suitable for your requirements. Traditionally, the choice will be made on price. However, as shown in previous example, the decision making process is far more complex than a single metric. Granted, the final decision may be made based on the most prevalent factor, but the other viewpoints should also be given consideration.

2.2 Migration from existing deployments

In this section we discuss migration from two existing legacy scenarios; we illustrate areas to consider, which demonstrate practical application of new technologies.

2.2.1 Migration paths from RS/6000 Scalable POWERparallel (SP)

The RS/6000 Scalable POWERparallel® (SP) was the original IBM UNIX Enterprise and High Performance Computing (HPC) solution, and is considered the forerunner to today's POWER4™ and POWER5 servers. The RS/6000 SP had a number of technological advantages at the time:

- Centralized hardware control: Parallel System Support Programs (PSSP) software provided the functions required to manage an SP system. It provided a single point of control for administrative task and helped increase productivity by letting administrators view, monitor, and control system operations. Designed originally for HPC use, the SP was also deployed in commercial environments due to its performance and capability to easily manage large number of AIX images from a single point of control (for example, it was much easier to manage a 32-node SP system than 32 standalone RS/6000 servers).
- SP Switch: A high-speed switched network interconnect between all nodes in an SP complex which provided the required speed for parallel applications, such as LoadLeveler® and GPFS™. The SP switch appeared as another network interface to the operating system, thus it could be used by for application and infrastructure traffic (for example, TSM backups, Lotus® Domino® replication, and so on).
- Scalability: Three different node sizes available (thin, wide and high), allowing up to 16 nodes per frame, depending on configuration. A single SP complex could scale up to 128 nodes (8 or more frames depending on deployed node types). Larger systems were available via special bid.

Customers still using SP deployments need to seriously consider a migration path, for a number of reasons:

- PSSP is only supported with AIX 5L V5.2.
- At the time of writing AIX 5.1 was withdrawn from support 1st April 2006.
- AIX 5L V5.2 is scheduled to be withdrawn on September 30th, 2008.
- PSSP V3.5 is the only supported version, and is scheduled to be withdrawn from support on April 30th, 2008.
- PSSP does not support the latest IBM System p5 hardware and HPC infrastructure.
- The hardware supported by PSSP has been withdrawn from marketing.

Those familiar with SP hardware, also might not be aware of the features and benefits of the current IBM System p5 family. In the following pages we will document some example migration paths.

> **Note:** The following discussions make reference to given models of IBM hardware. While such suggestions might be appropriate to your individual case, they equally may not. We provide suggested hardware options to illustrate the properties to consider when evaluating choices.

Scenario 1

Figure 2-1 shows our first example SP deployment; 5 Frames, 16 nodes per Frame, a combination of Thin and Wide nodes (based on POWERPC). SP Switch, running AIX 5L V5.1 and PSSP 3.4.

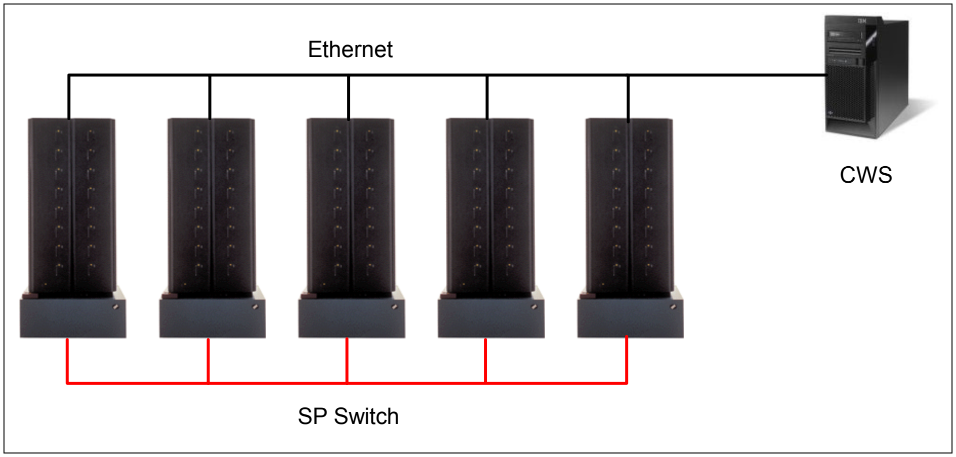


Figure 2-1 RS/6000 SP Example Deployment #1

Assumptions

In this case, the SP is used for management purposes and does not run parallel applications. Existing applications run happily on the CPU/RAM configurations of the nodes, therefore there is no urgent requirement for faster CPUs or larger amounts of RAM.

Depending on the final requirements, a number of potential migration paths exist. We will illustrate two:

1. Migrating applications to a multi-frame IBM System p5 570 complex.
   Old SP nodes had fixed CPU and memory capacity. Both IBM System p4 and IBM System p5 hardware can be carved into logical partitions (LPARs). This allows LPARs to have various capacities. Assuming the same amount of AIX instances is required (80), Micro-Partitioning could be leveraged to create the required number of LPARs. This feature allows LPARs to run with less than a whole physical CPU (maximum granularity is 0.1 CPU units).
   This allows a high level of utilization of the physical box. A switched gigabit Ethernet fabric could be used to provide a similar interconnect to the SP Switch, but at a fraction of the cost. Greater flexibility could be achieved by leveraging Partition Load Manager (PLM) to provide dynamic resource allocation between partitions. It is important to note, any migration away from SPs would require replacing AIX 5L V5.1 with AIX 5L V5.3 and PSSP with Cluster Systems Management (CSM) (if a Management Cluster was still

required). Such changes would need to be understood and evaluated. Multiple IBM System p5 570s would offer the potential of High Availability clustering, if required. If application high availability is not the prevalent decision factor, a larger single-frame IBM System p5 configuration could be an alternative.

> **Note:** For more information regarding CSM and a migration from PSSP, refer to *CSM Guide for the PSSP System Administrator*, SG24-6953 and *Transition from PSSP to Cluster Systems Management (CSM)*, SG24-6967. For information regarding IBM System p5 Virtualization capabilities, refer to *Advanced POWER Virtualization on IBM System p5*, SG24-7940.

2. Migration onto an IBM System BladeCenter using JS21 blade servers. Up to 14 JS21 blade servers can be hosted in a single BladeCenter chassis. A BladeCenter is either 7U or 9U in size, depending on model. So either 6 or 4 centers (providing up to 56 or 84 Blades respectively) can be hosted in a single 42U rack. The JS21 blades can be configured with up to 4 CPUs and have **some** of the virtualization features of their larger IBM System p5 cousins.

   Depending on your application requirements, IBM BladeCenter might be a more natural path than native IBM System p. There are a number of comparable principles between the legacy RS/6000 SP and the IBM BladeCenter. For example, the frame/node concept is comparable to the BladeCenter/Blade relationship. Another similar concept is the high-speed network interconnect.The SP had the SP Switch, whereas a Blade chassis connects all its hosted Blades with a high-speed Ethernet switch.

   In addition, the new IBM BladeCenter H Chassis provides support for Bridge modules which create a switched fabric between Chassis. External connectivity has many options including Gigabit Ethernet, 10 Gigabit Ethernet and 4x InfiniBand.

   > **Note:** For more information about IBM BladeCenter JS21, refer to *IBM BladeCenter JS21: The POWER of Blade Innovation*, SG24-7273 and *IBM BladeCenter JS21 Technical Overview and Introduction*, REDP-4130

In both cases you may have to re-evaluate your storage options and requirements. It was common for SP deployments to use IBM Serial Storage Architecture (SSA) disk storage. If your deployment still utilizes this technology you will need to replace it with a modern storage technology (SAN-based), such as IBM System Storage™ solutions. Similar to the server hardware itself, the storage hardware is equally important to a deployment; requirements, capacity, performance and cost should all be considered when implementing change.

Note: For more information about the range of disk storage products that IBM offers, visit the IBM Total Storage product Web site:
http://www-03.ibm.com/servers/storage/disk/

Scenario 2

Figure 2-2 shows a second example SP deployment: 12 Frames, 24 Power3 High nodes, 96 Winterhawk nodes, SP Switch running AIX 5L V5.2, PSSP 3.5, GPFS 2.2, HACMP 5.1, LoadLeveler 3.2.

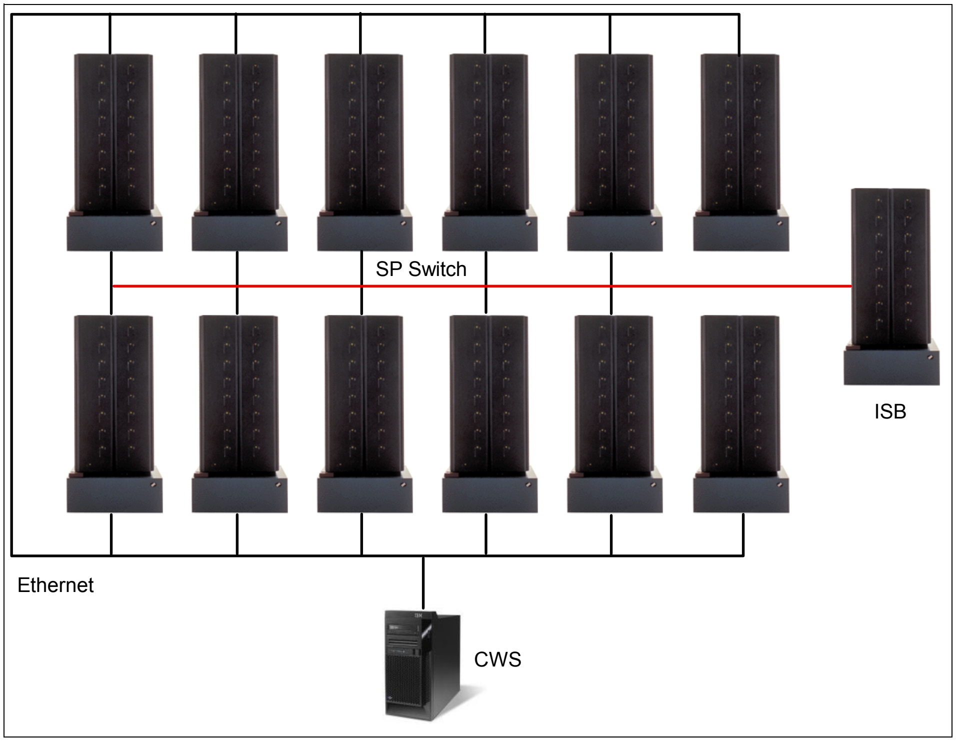


Figure 2-2 RS/6000 SP Example Deployment #2

Assumptions

In addition to the areas we covered in the first migration scenario, this 2nd example has an additional set of variables. HPC applications make higher demands on the underlying environment; the balanced requisites of the HPC software components also add a further topic to migration consideration. When considering migrating from AIX 5L V5.2 to AIX 5L V5.3, you also need to consider this impact on the HPC software components. You may require upgraded versions and therefore new licenses. At the time of writing HACMP 5.1 was withdrawn from support as of Sept. 1st, 2006, so this requires the purchase of a license to upgrade HACMP either Version 5.2, 5.3, or 5.4 (all currently supported on AIX 5L V5.3). Also LoadLeveler 3.3 is the least version supported on AIX 5L V5.3, so you need to upgrade it. Considering the number of operating system images and applications involved, PSSP needs to be replaced by Cluster Systems Management (CSM). GPFS would also require an upgrade to be supported on AIX 5L V5.3.

Due to the initial large number of (SP) nodes, and the varied model type, this environment would make efficient use of LPARs as combination of dedicated (whole CPU) and micro-partitioned LPARs. The number of required LPARs and associated resources would dictate the suitable models. For example, a new infrastructure based on a number of IBM System p5 595 frames could be a potential solution. Alternately, depending on requirements, combinations of IBM System p5 570 and IBM System p5 575 frames could be more suitable.

Note: Your available options of IBM System p5 hardware will depend on more than just CPU/RAM capacity. You also need to appreciate differences in size, power and weight requirements, and number of I/O slots. For example, an IBM System p5 570 requires a 19-inch rack, whereas a IBM System p5 575 requires a wider rack (24 inches). Therefore, in some cases it maybe more cost-effective to use multiple smaller IBM System p5 servers, as opposed to the larger systems. See Table 2-7 on page 49.

Also, as suggested in “Scenario 1” on page 44, an IBM BladeCenter JS21-based deployment could be a valid alternative. When considering a target platform, you need to understand your environment to determine the critical factors: is performance the driving requirement, or cost? Perhaps you have limited space to house new equipment, or you need a strategy for a staggered migration away from the legacy deployment?

The sample HPC deployment has a dependency on the SP Switch interconnect for which you need to consider alternatives. The bandwidth and latency requirements need to be understood and sized. An HPC application similar to our example would be dependent on the sustained point-to-point bandwidth capability of the SP Switch. With the advent of IBM System p4 p690 server, the SP Switch and SP Switch 2 were superseded in 2003 by the IBM eServer pSeries High Performance Switch (HPS). At the time of writing this is still an available option for certain IBM System p4 and IBM System p5 hardware.

> **Note:** Bandwidth and latency may not be the only network characteristics to be considered. Communication protocol used by the application is equally important (TCP/IP, LAPI, and so on). For more information about the HPS, refer to *An Introduction to the New IBM eServer pSeries High Performance Switch*, SG24-6978.

As previously mentioned, depending on your individual requirements, switched gigabit Ethernet, 10 Gbit Ethernet or InfiniBand could also be suitable replacements for the older SP switch.

2.2.2 Power 4 to Power 5 migration considerations

Question

If you have an existing IBM System p4 deployment, what would be the advantage in moving to IBM System p5 hardware?

The answers to this question may be:

- Broader range of hardware configurations (from 1 to 64 CPU cores).
- Faster POWER5 and POWER5+ CPUs, enhanced with simultaneous multi-threading (SMT) capabilities. This is an enhancement which allows two separate threads to execute simultaneously on the POWER5 CPU.
- Advanced POWER Virtualization features, such as Micro-partitioning, Virtual SCSI and Ethernet.
- Enhanced DLPAR capabilities; on p4 the smallest units are the whole physical CPUs and 256 MB blocks of RAM; on p5 this is improved to 0.1 of a CPU and 16 MB blocks of RAM.

> **Note:** For a more detailed explanation of the POWER5 architecture and Advanced POWER Virtualization feature, refer to *Advanced POWER Virtualization on IBM p5 Servers: Architecture and Performance Considerations*, SG24-5768.

Depending on your environment, the increased capacity (CPU, RAM or I/O) of System p5 may propose new consolidation, growth or architectural opportunities. For example, the extra capacity may allow you to host larger partitions or a greater number of partitions in a physical box. Also, sub-CPU allocation allows more efficient LPAR allocation, for example, if 2.5 CPU are adequate, then why allocate 3?

Keep in mind that an IBM System p4 and an IBM System p5 cannot be managed by the same HMC. This is due to the communication used for the CEC connection: RS-232 with IBM System p4, Ethernet with IBM System p5. So a different HMC would be required for a new IBM System p5 deployment.

> **Note:** If you have a 7315 model HMC managing your existing IBM System p4 environment, this can be reinstalled as with he code for IBM System p5 HMC. For more information, refer to the HMC portion of the IBM Fix Central site:
>
> `http://www-912.ibm.com/eserver/support/fixes/fixcentral`

If you were considering a transition from IBM System p4 to IBM System p5 and you currently have dual 7315 HMCs for your System p4, you could remove one HMC from the p4 complex, reinstall as a p5 HMC and reuse it for your new System p5 environment. This would provide an HMC for both complexes, removing the requirement to purchase a new HMC. Once transition of your applications to the new p5 environment has finished, you can also reuse the second p4 HMC as a redundant p5 HMC (after reinstall).

> **Note:** The platform-specific HMC requirement could present an additional challenge if your current IBM System p4 deployment uses the HPS interconnect. While HPS is supported on both IBM System p4 and IBM System p5, the relationship between HPS and HMC means the HPS cannot connect both platforms at the same time. You would need to have two separate HPS networks.

Table 2-7 outlines a high-level comparison in Virtualization features between System p4 and p5 product families.

Table 2-7 Comparison between IBM System p4 and p5

Feature	System p4	System p5
Logical partitioning (LPAR)	Y	Y
Micro partitioning	N	Y[a]
Dynamic LPAR (DLPAR)	Y[b]	Y
Partition Load Manager (PLM)	Y	Y
Virtual IO Server (VIO)	N	Y
Simultaneous multi-threading (SMT)	N	Y[c]

[a] Requires AIX 5L V5.3 or higher

[b] p4 DLPAR is limited to whole CPU and 256MB units

[c] Requires AIX 5L V5.3 or higher

2.3 Hints and tips

In this section we highlight important concerns, observations and recommendations. Even though some comments may appear obvious, we feel they may not be emphasized enough in the available documentation.

2.3.1 Virtual I/O Server (VIOS)

Upgrading a single VIOS

As previously mentioned in 2.1.3, “Integrated Virtualization Manager (IVM)” on page 20, and at the time of writing, upgrading from Version 1.2.0 to 1.3.0 requires an outage. If your system has a single VIOS, then the implications of an outage needs to be understood and accommodated.

Upgrading dual VIOS

If you are using a dual VIOS solution, and your environment has been designed to provide redundancy, an outage on one VIOS should not affect your client LPARs. We tested such a scenario to understand how the upgrades would appear from the client LPAR. While client impact is not expected, we were interested to see what errors or warnings are generated.

Note: Implementing dual VIOS and MPIO are discussed in Chapter 5 of *Advanced POWER Virtualization on IBM System p5*, SG24-7940. Our goals are not to repeat existing work, but to demonstrate practical implications of the theory.

Figure 2-3 on page 51, illustrates a single SAN LUN, allocated to dual VIOS, hosted to a single LPAR. From the figure you can see how the availability of the LUN is protected using dual paths, VIOS, and physical and virtual adapters.

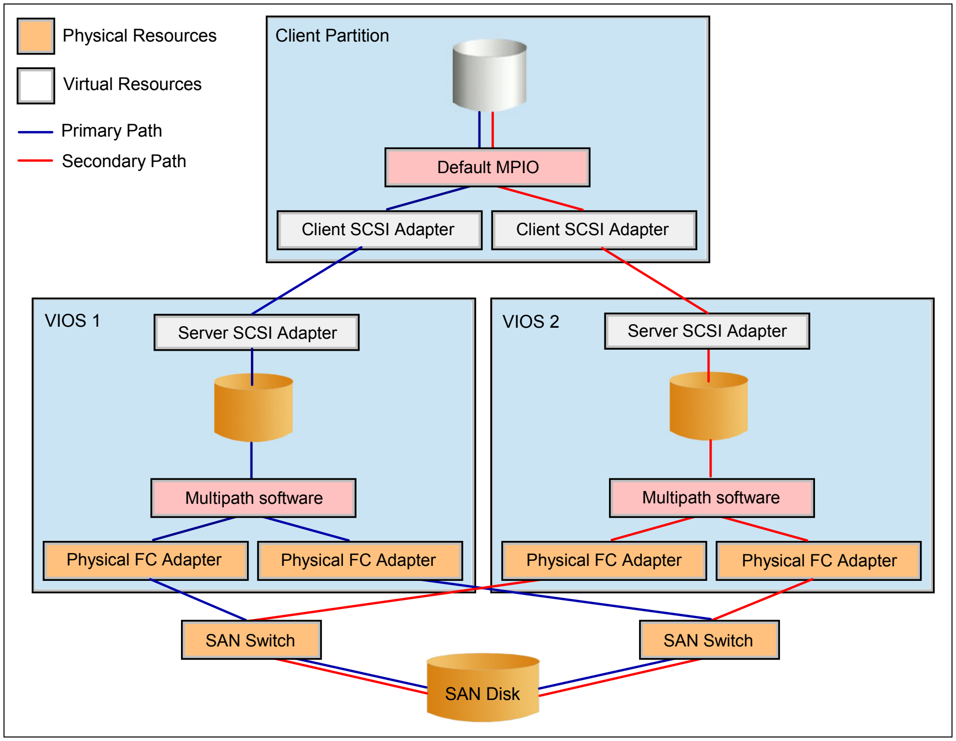


Figure 2-3 Single SAN LUN, dual VIOS allocation

Example 2-1 shows the **lsmap** command output from both VIOS. This illustrates the mapping of a single LUN to both VIOS, which both virtualize it to a single LPAR.

Example 2-1 lsmap output from both VIOS

```
VIO1 :
SVSA            Physloc                                      Client Partition
ID
--------------- -------------------------------------------- ------------------
vhost0          U9117.570.10C5D5C-V8-C10                     0x00000006

VTD                   virt_p5_mp
LUN                   0x8200000000000000
Backing device        hdisk10
Physloc
U7879.001.DQDKZNP-P1-C2-T1-W200300A0B812106F-LA000000000000

VIO2 :
```

```
SVSA            Physloc                                        Client Partition
ID
--------------- ---------------------------------------------- ------------------
vhost0          U9117.570.10C5D5C-V9-C20                       0x00000006

VTD                   virt_p5_mp
LUN                   0x8100000000000000
Backing device        hdisk10
Physloc
U7879.001.DQDKZNP-P1-C6-T1-W200300A0B812106F-LA000000000000
```

The allocated LUN appears as a single multi-pathed disk, hosted between virtual SCSI adapters on the client LPAR. Example 2-2 shows how this influences output of commonly used commands.

Example 2-2 Output from Client LPAR

```
# lspv
hdisk0          00cc5d5c3e364341                    rootvg          active

# lsdev -Cc disk
hdisk0 Available  Virtual SCSI Disk Drive

# bootlist -m normal -o
hdisk0 blv=hd5
hdisk0 blv=hd5

# lspath
Enabled hdisk0 vscsi0
Enabled hdisk0 vscsi1

# lspath -l hdisk0 -F"connection:parent:path_status:status"
820000000000:vscsi0:Available:Enabled
810000000000:vscsi1:Available:Enabled
```

We performed the VIO upgrade on the first VIOS and then rebooted it. The effect on the client LPAR was minimal. Example 2-3 shows the single reported error in the AIX Error Log.

Example 2-3 AIX error log from client LPAR, during first VIOS upgrade

```
DE3B8540   1012163206 P H hdisk0         PATH HAS FAILED
```

Once the upgraded VIOS back online, we repeated the process on the other VIOS. As before, there was no impact to availability of the client LPAR. Example 2-4 shows the additional errors generated.

Example 2-4 AIX error log from client LPAR, during second VIOS upgrade

```
410D7CDC   1012172106 T S vscsi0        Temporary VSCSI software error
DE3B8540   1012172106 P H hdisk0        PATH HAS FAILED
857033C6   1012172006 T S vscsi0        Underlying transport error
```

From this we can conclude that prior to both VIOS upgrades, the first VIOS was the backup path and the second was the primary.

Note: Aside from monitoring I/O performance, it does not appear possible to determine which is the active path.

To determine the active path, we checked the path priority. This can be listed using the **lspath** command, as shown in Example 2-5.

Example 2-5 lspath output showing priority

```
# lspath -AHE -l hdisk1 -p vscsi0
attribute value description user_settable

priority  1     Priority    True

# lspath -AHE -l hdisk1 -p vscsi1
attribute value description user_settable

priority  2     Priority    True
```

The lower the number, the higher the priority - one being the highest. The path with the highest priority will be used. Priorities can be changed using the **chpath** command. However, the priorities only give an indication to what should be primary or secondary - as opposed to what actually *is*.

Note: A reboot of the client LPAR is required to commit a change in path priority.

This explains the errors received during the two upgrades. It is important to appreciate how a VIOS outage is represented from the client's viewpoint. Once it is known that scheduled maintenance will produce given errors, this knowledge can be part of the planning, warning system operators and administrators that this is expected.

2.4 Reliable Scalable Clustering technology (RSCT)

Reliable Scalable Cluster Technology (RSCT) is software running on AIX 5L and Linux machines and provides infrastructure services for many IBM clustering technologies, such as Cluster Systems Management (CSM) or High Availability Cluster Multi-Processing (HACMP).

RSCT can be leveraged for monitoring of an individual machine or a group of machines via Resource Monitoring and Control (RMC). RSCT and RMC are part of AIX 5L and their use is included as part of the AIX 5L license. For Linux deployments, it comes as part of Cluster Systems Management (CSM) or other clustering software.

2.4.1 How RSCT works

The main purpose of RSCT is to determine the current state of resources (for example daemon state, node availability or hardware state), acknowledge a change then report and react within defined scope.

The RSCT subcomponent *Resource Monitoring and Control (RMC)* provides an interface to local system resources. For Management Clusters, RMC provides global access to cluster node resources.

RMC uses Resource Managers to interact with hardware and software resources. Resource Managers query resource status and report it to RMC daemon (for example, current file systems' size).

AIX 5L or Linux machines can be grouped via RSCT into groups which can behave in different ways:

- Management domains - where a controlling master communicates to its subordinated nodes, but the nodes do not interact with each other.
- Peer domains - where all nodes are equal and communicate to each other.

For Peer Domains there are two additional RSCT components: *Group Services*, a cross node/process coordination layer; *Topology Services* and a layer responsible for node/network failure detection.

While using any kind of RSCT domain, authentication is done via *Cluster Security Services* (CTSEC).

2.4.2 RSCT advantages and why to use RSCT and RMC

RSCT is included as standard in AIX 5L. It is possible to use RMC on a standalone node to monitor many predefined conditions. Figure 3-1 on page 65, illustrates a selection of the standard predefined conditions. It is possible to use predefined conditions, or customize them to monitor in a way which best fits into your environment.

You can also create your own sensors and conditions. For example, create a new sensor starting from an existing script.

Responses to conditions determine what action will be executed with an occurrence; sending an e-mail to given recipient being one such example. Custom responses provide a method to integrate with external monitoring products. For example, in the case of Tivoli Enterprise Console (TEC) you can use TEC command line utilities, such as **`wpostemsg`**.

In more complex situations, you can create your own peer domain and monitor a group of machines from single point of management without a need to define monitors on each individual machine.

> **Note:** For standalone machines you can use WebSM for managing and viewing events and error conditions (see Figure 3-6 on page 68). WebSM can also be used to centrally administer and monitor CSM-managed environments (a form of management domains). However, WebSM does not support the managing and monitoring of peer domains.

For more detailed discussions about RSCT monitoring and RMC usage, refer to 3.1, “RMC monitoring on standalone node” on page 64.

In many cases you already use RSCT within existing licensed products. For example, RSCT Management domains can be:

- HMC managing LPARs
- CSM managing nodes

Or, they can be RSCT peer domains:

- HMCs in Cluster Ready Hardware (CRH) cluster
- Tivoli System Automation (TSA)

HACMP nodes use a special kind of the domain which is called HACMP domain, which is similar to a Peer Domain, but does not store its configuration in RSCT registry, and reports events to HACMP cluster daemon who is responsible for reacting to events.

2.5 Cluster Systems Management (CSM)

Cluster System Manager (CSM) is a software package that enables central management of AIX 5L and Linux-based machines. Though CSM itself does not provide any kind of revenue-generating service, its centralized management greatly decreases the complexity of a system, facilitating efficient and logical administration and maintenance. The centralization promoted by CSM allows an environment of 500 servers to be managed as simply as 50.

In 3.3, “CSM implementation in an AIX environment” on page 81 we focus on the main benefits and demonstrate how to leverage them. We also experiment with some of the new features introduced in CSM V1.6 and outline the migration of an existing cluster from CSM V1.5 to V1.6. Our scenarios consist of IBM System p4 and IBM System p5 nodes running AIX 5L, but the same logic and principles apply to Linux or mixed clusters equally.

CSM has recently introduced support for integration of IBM System Blue Gene Solution, however this is not a topic of discussion for this book. For more information, refer to *IBM Blue Gene System Administration*, SG24-7352.

2.5.1 Distributed commands and node groups

Centrally running commands concurrently on many nodes greatly improves the efficiency of the overall cluster configuration. There are many occasions where the same command needs to be executed on all nodes, individual nodes or specific groups of nodes.

Node groups allow administrators to perform tasks on subsets of nodes without specifying each individual one. These can be either manually maintained or alternatively CSM provides predefined dynamic groups. Dynamic groups are arranged by persistent attribute, such as operating system or hardware type. For practical discussions regarding CSM, refer to 3.3, “CSM implementation in an AIX environment” on page 81.

To demonstrate the benefit of the **dsh** (*distributed shell*) command we show how to query the World Wide Port Name (WWPN) of the all Fibre channel cards the cluster (see Example 3-42 on page 113).

Note that there are various dsh options for filtering output and also other “d” commands, such as:

- **dcp**: copy files across your cluster nodes.
- **dshbak**: formats (consolidates) the output of the dsh command.
- **dping**: pings nodes or devices in parallel.

For command descriptions, see official product documentation *IBM Cluster Systems Management for AIX 5L and Linux V1.6 Command and Technical Reference*, SA23-1345. Also look for command reference updates as new commands or flags may be released between versions. In 2.5, “Cluster Systems Management (CSM)” on page 56 we highlight other CSM commands.

2.5.2 Distributed Command Execution Manager (DCEM)

DCEM provides a graphic interface (GUI) for running single or sets of the commands to cluster nodes. Here is the general procedure of the job preparation:

1. Prepare your command or script
2. If applicable, distribute required scripts to target nodes via **dcp** or CFM
3. Prepare the DCEM job within the DCEM GUI
4. Set the job parameters, such as timeout for job completion
5. Execute or schedule the prepared job
6. Monitor the job progress (refer to the example Figure 3-16 on page 91**)**
7. Review the execution logs

DCEM is perhaps not suitable for experienced Administrators using complicated command line options. However, in some cases DCEM can be a suitable substitute for job scheduling software. DCEM is also a suitable interface for Operators or First Level support, as it provides an interface for the less experienced. Routine commands can be saved for future use and re-used as required. Experienced Administrators can just prepare DCEM commands and let Operators work via the GUI.

For an example of using DCEM to perform an AIX Operating System backup, refer to 3.3.3, “Using DCEM as a backup strategy in an CSM cluster” on page 88.

2.5.3 Monitoring and WebSM

CSM ships with a very rich set of monitoring tools which include:

- File system monitors (attribute based; for example free space and inodes used).
- AIX 5L Error log and Linux syslog monitoring.
- Hardware monitors, including HMCs Service Focal Point (SFP) integration.
- Possibility to monitor non-node devices, such as HMCs and VIOS.
- SNMP monitoring.

- Monitoring via custom scripts.

Note: WebSM is another powerful GUI which in some cases could be a substitute for enterprise event management tools for your AIX and Linux nodes. This GUI is illustrated in Figure 3-19 on page 94, showing status of the nodes and also incoming events from monitored nodes.

Other features provide common monitoring and event handling tasks:

- Responses to the events:
  - Passive: for example, sending a predefined E-mail notification
  - Active: for example, running a command or a script
- Possible integration with SNMP manager, such as NetView® via **`snmptrap`** command.

In 3.3, “CSM implementation in an AIX environment” on page 81 we demonstrate how to use supplied CSM and existing RMC monitors in an CSM-managed environment.

2.5.4 Hardware control and remote console support

Hardware control enables centralized remote control (power on/off, startup, shutdown, restart) of managed nodes. For large environment this is a tremendous advantage, as individually logging onto hundreds of separate machines is a time consuming activity.

It is common with today’s larger machines and virtualization capabilities to run many AIX or Linux instances than the machines’ physical processors. For example, there are cases where a 16 CPU machine hosts 40 AIX LPARs. It is much simpler to have centralized control from a single point (CSM Management Server) than to logon on each HMC individually. CSM provides Hardware and LPAR control and monitoring from the command line.

CSM achieves remote hardware control by providing integration with the hardware control points; for example an HMC, IVM, IBM BladeCenter Management Module, RS/6000 SP Frame Supervisor or standalone IBM System p servers.

CSM provides support for remote consoles on Cluster nodes and devices. This is a similar principle to the HMC virtual terminals (VTERM) or the **`s1term`** PSSP command. Similar to the legacy **`s1term`** command, **`rconsole`** provides both read and read-write consoles. In the case of IBM System p LPARs remote consoles provide access LPARs’ SMS menu, used during installation and diagnostics.

Example 3-39 on page 112 illustrates the use of **`rconsole`** to monitor the installation progress of LPARs - without the need to log onto multiple HMCs or other console devices. In our example we used **`-r`** flag to have only read access and enable another user to gain read-write access to the console. For a given node/device one read-write and multiple read-only consoles are supported.

2.5.5 Configuration File Manager (CFM)

File synchronization can be very important as it can standardize system configuration on all or a subset of nodes; standardization can reduce support costs by mitigating configuration errors and differences. For example, an application failure after HACMP takeover due to the different /etc/services file on the HA nodes can be avoided maintaining /etc/services file using CFM**.** CFM has very straightforward, yet powerful method for synchronizing files across the nodes. You just need to place the files to be synchronized to the /cfmroot directory on the CSM Management Server. The /cfmroot directory represents the top directory in the file system hierarchy on the nodes. Therefore the /cfmroot/etc/hosts file will be distributed as /etc/hosts on the target nodes.

Other notable CFM features are:

- Global, Group and Node based policies for file synchronization; achieved by adding *._GroupName* or *._NodeName* to the source filename. If there is no group or node specification, the file is distributed to all cluster nodes.
- Support for pre- and post- distribution scripts. This has many practical applications; for example recycling subsystems may be required after updating the appropriate configuration file. Another example would be to run **`sendmail -bi`** command after the /etc/aliases file has changed. Scripts are associated with distributed files by **filename.pre** and **filename.post** extensions.
- Support for post operating system installation and first boot scripts. This is achieved by placing them into the /csminstall/csm/scripts directory structure. Scripts can also use the *._GroupName* or *._NodeName* extensions. This can be useful for customization of the operating system images for a specific hosted applications. You can also customize what happens after running an **`updatenode command`**.
- Users management and synchronization can be achieved by placing links to the /etc/passwd and other related-files.
- Symbolic-linked files within /cfmroot are distributed as actual files. Links on the managed nodes must be handled via pre- or post-distribution scripts.

The synchronization is triggered by executing the cfmupdatenode command on the Management Server, targeting all nodes, chosen groups or individual nodes according to the passed parameters. Cron jobs on the Management Server can be used to automate CFM distribution using the same method.

> **Note:** For more information and a sample configuration, refer to the CFM section in *IBM Cluster Systems Management for AIX 5L and Linux Administration Guide Version 1, Release 6*, SA23-1342-02. In 3.3.16, “Using CFM” on page 117 we present practical examples of using CFM.

2.5.6 Network Installation Manager (NIM) and CSM

As summarized in 1.1.15, “Network Installation Management (NIM)” on page 10, NIM provides a broad set of software maintenance tasks over and above base AIX network installation:

- Installation of the base OS
  Combining remote power tools (for example HMC access) it provides remote operating system installation without need of physical media or server manipulation.
- Integration with the Service Update Management Assistant (SUMA) tool
  This allows NIM to track the AIX technology or maintenance levels on the managed nodes and enable restore of the operating system to the appropriate level.
- Software maintenance tasks for the non base operating system software can also be performed via NIM; the tasks can also be scheduled (using NIM scheduler).
- Inventory comparison and reports
  These help to track software history and check software integrity across the managed nodes.
- Support restores of the certain non-node devices, such as IVM or VIOS.
- Easy installation via EZNIM.

Refer to 3.4, “AIX Installation strategy” on page 118 for an example of how to use NIM.

> **Note:** For extensive examples of how to use NIM, refer to *NIM From A to Z in AIX 5L*, SG24-7296.

As previously mentioned, NIM is shipped with AIX 5L, thus it is not part of CSM. However, you can leverage NIM functionality independently in a CSM environment, or you can integrate NIM with CSM:

- Integration with remote power control and console through **`netboot command`**. Using NIM together with **`netboot`** command enables you to perform a non-prompted AIX operating system install.
- CSM node and node groups synchronization with NIM groups via **`csm2nimgrps`** and **`csm2nimnodes`** commands.
- Node installation via NIM can be simplified using **`csmsetupnim`** command. This creates NIM client definitions from the CSM Node definitions.
- Integration of the CSM adapters definition and NIM secondary adapters support for defining all required adapters during the operating system installation process. This reduces the manual adapter configuration in the node post-installation phase.

If your deployment contains just AIX 5L nodes, we recommended that your NIM Master is hosted on your CSM Management server. The idea is similar to a legacy PSSP Cluster, where the Control Workstation (CWS) was also the NIM Master. This makes administrative tasks much simpler, as you only have to maintain one operating system image.

Although CSM is not dependent on NIM (as PSSP was), it is equally viable to run your NIM Master and Management Server on separate machines. If you are building a new deployment with CSM, it is more likely to build the NIM Master and Management Server as the same server. However, if you are implementing CSM on an existing deployment you may already have an operational NIM Master. In such a case it is logical to leverage your existing NIM Master. We detail this approach in 3.3.11, “CSM Integration with the existing NIM server” on page 108.

3

Monitoring and system management scenarios

In this chapter we implement a subset of the topics discussed in the previous chapters. While highlighting given features and products, we demonstrate that certain aspects are not so complex to implement. This chapter describes:

- “RMC monitoring on standalone node” on page 64
- “RMC monitoring in a peer domain” on page 73
- “CSM implementation in an AIX environment” on page 81
- “AIX Installation strategy” on page 118

3.1 RMC monitoring on standalone node

Our first scenario involves base monitoring for an AIX LPAR hosted on IBM System p4 p650 hardware. The goal is to monitor file system usage and the status of the sendmail daemon. We will use a predefined response to deliver e-mail to root when an event occurs. Additionally, we will create a custom response to run a given script when an event occurs.

3.1.1 RSCT installation

For this example we have used RSCT Version 2.4.5 running on AIX 5L V5.3. However, this version is not a prerequisite for the demonstrated tasks, earlier RSCT versions supporting the same functionality.

The RSCT filesets are typically installed as part of the base AIX image. Example 3-1 lists the installed filesets in our given case; we have omitted the message filesets for readability.

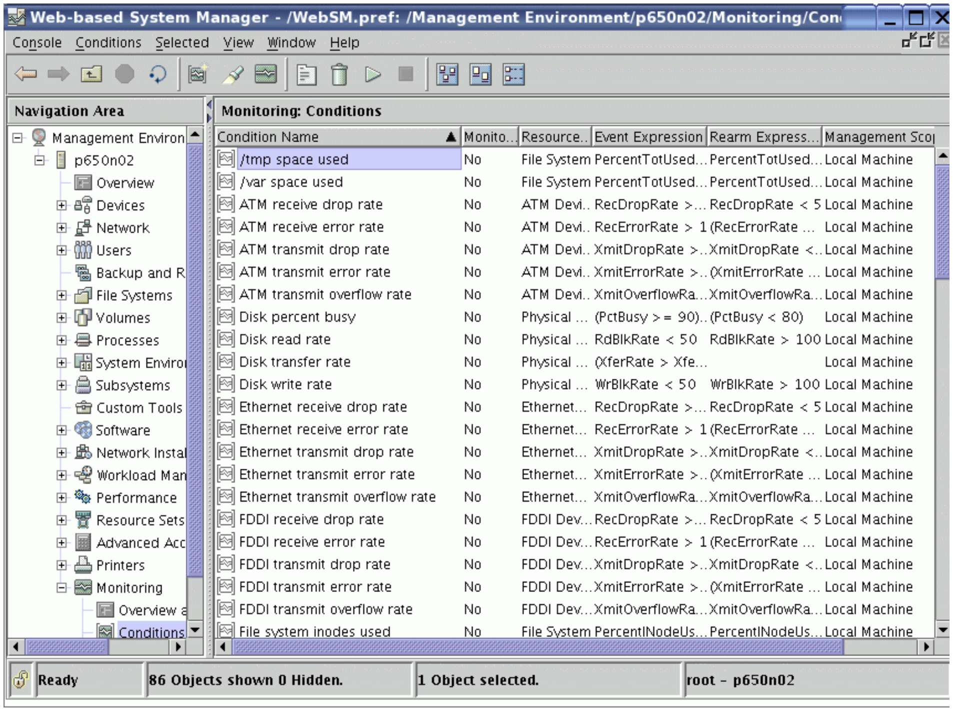
Example 3-1 RSCT version used in the scenario with standalone RMC monitoring

```
virt_p7@root:/# lslpp -L rsct*
  Fileset                        Level  State  Type  Description
(Uninstaller)
--------------------------------------------------------------------
  rsct.core.auditrm            2.4.5.0    C     F    RSCT Audit Log
Resource Manager
  rsct.core.errm               2.4.5.1    C     F    RSCT Event Response
Resource Manager
  rsct.core.fsrm               2.4.5.0    C     F    RSCT File System
Resource Manager
  rsct.core.gui                2.4.5.1    C     F    RSCT Graphical User
Interface
  rsct.core.hostrm             2.4.5.1    C     F    RSCT Host Resource
Manager
  rsct.core.lprm              2.4.5.0    C     F    RSCT Least Privilege
Resource Manager
  rsct.core.rmc                2.4.5.3    C     F    RSCT Resource
Monitoring and Control
  rsct.core.sec                2.4.5.2    C     F    RSCT Security
  rsct.core.sensorrm          2.4.5.0    C     F    RSCT Sensor Resource
Manager
  rsct.core.sr                 2.4.5.0    C     F    RSCT Registry
  rsct.core.utils              2.4.5.3    C     F    RSCT Utilities
```

3.1.2 Using predefined monitors

We use Web-based System Manager (WebSM) to define our monitor resource for monitoring of the free space of several base file systems. It is not possible to define monitors via **smitty**. We used the following steps to define our monitor:

1. Open the WebSM tool and select the **Monitoring** menu in the left-hand Navigation Area. Then chose the **Conditions** menu, you should see the predefined conditions as shown on Figure 3-1.

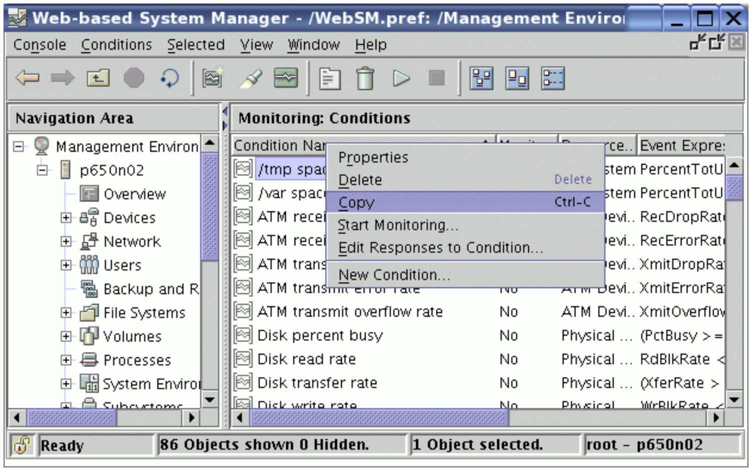


Condition Name	Monito...	Resource...	Event Expression	Rearm Express...	Management Sco
/tmp space used	No	File System	PercentTotUsed...	PercentTotUsed...	Local Machine
/var space used	No	File System	PercentTotUsed...	PercentTotUsed...	Local Machine
ATM receive drop rate	No	ATM Devi..	RecDropRate >...	RecDropRate < 5	Local Machine
ATM receive error rate	No	ATM Devi..	RecErrorRate > 1	(RecErrorRate ...	Local Machine
ATM transmit drop rate	No	ATM Devi..	XmitDropRate >..	XmitDropRate <..	Local Machine
ATM transmit error rate	No	ATM Devi..	XmitErrorRate >..	(XmitErrorRate ...	Local Machine
ATM transmit overflow rate	No	ATM Devi..	XmitOverflowRa...	XmitOverflowRa...	Local Machine
Disk percent busy	No	Physical ...	(PctBusy >= 90)..	(PctBusy < 80)	Local Machine
Disk read rate	No	Physical ...	RdBlkRate < 50	RdBlkRate > 100	Local Machine
Disk transfer rate	No	Physical ...	(XferRate > Xfe...		Local Machine
Disk write rate	No	Physical ...	WrBlkRate < 50	WrBlkRate > 100	Local Machine
Ethernet receive drop rate	No	Ethernet...	RecDropRate >...	RecDropRate < 5	Local Machine
Ethernet receive error rate	No	Ethernet...	RecErrorRate > 1	(RecErrorRate ...	Local Machine
Ethernet transmit drop rate	No	Ethernet...	XmitDropRate >..	XmitDropRate <..	Local Machine
Ethernet transmit error rate	No	Ethernet...	XmitErrorRate >..	(XmitErrorRate ...	Local Machine
Ethernet transmit overflow rate	No	Ethernet...	XmitOverflowRa...	XmitOverflowRa...	Local Machine
FDDI receive drop rate	No	FDDI Dev...	RecDropRate >...	RecDropRate < 5	Local Machine
FDDI receive error rate	No	FDDI Dev...	RecErrorRate > 1	(RecErrorRate ...	Local Machine
FDDI transmit drop rate	No	FDDI Dev...	XmitDropRate >..	XmitDropRate <..	Local Machine
FDDI transmit error rate	No	FDDI Dev...	XmitErrorRate >..	(XmitErrorRate ...	Local Machine
FDDI transmit overflow rate	No	FDDI Dev...	XmitOverflowRa...	XmitOverflowRa...	Local Machine
File system inodes used	No	File System	PercentINodeUs...	PercentINodeUs...	Local Machine

Ready | 86 Objects shown 0 Hidden. | 1 Object selected. | root - p650n02

Figure 3-1 RMC Predefined conditions in WebSM

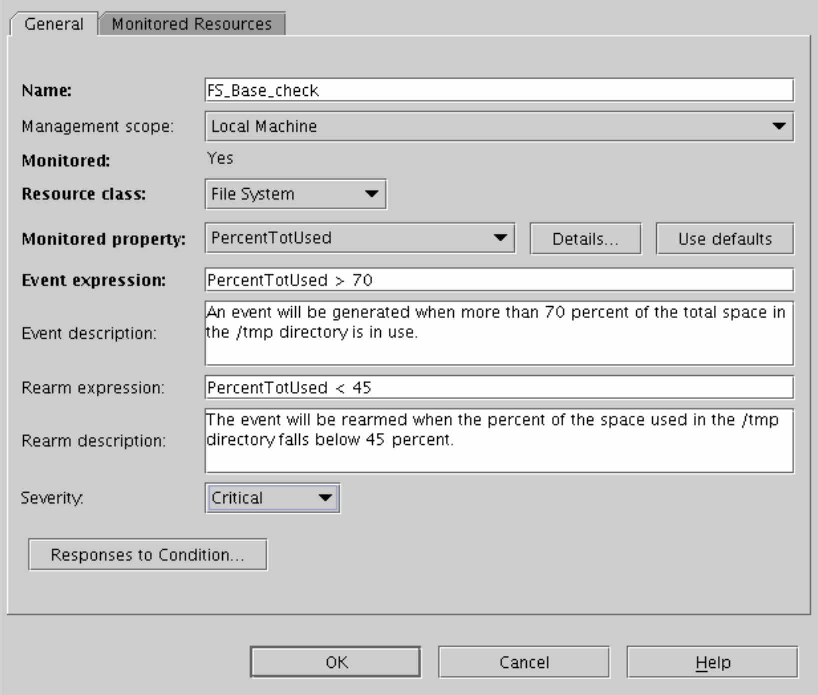
2. We will copy and customize the predefined condition "`/tmp space used`" to provide monitoring for several filesystems. Right-click the condition and choose **Copy**, as shown on Figure 3-2 on page 66.



Condition Nar	Moni...	Resource..	Event Expre:
/tmp spac		stem	PercentTotU
/var spac		stem	PercentTotU
ATM recei		evi..	RecDropRat
ATM recei		evi..	RecErrorRat
ATM trans		evi..	XmitDropRa
ATM transmit error rate	No	ATM Devi..	XmitErrorRa
ATM transmit overflow rate	No	ATM Devi..	XmitOverflow
Disk percent busy	No	Physical ...	(PctBusy > =
Disk read rate	No	Physical ...	RdBlkRate <
Disk transfer rate	No	Physical ...	(XferRate >
Disk write rate	No	Physical	WrBlkRate <

Figure 3-2 Copy the predefined "/tmp space used" condition

3. Open the copied Condition, double click to edit the condition properties; here you can edit the Name, Severity, Event and Rearm expression as shown on Figure 3-3.



Name: FS_Base_check

Management scope: Local Machine

Monitored: Yes

Resource class: File System

Monitored property: PercentTotUsed

Event expression: PercentTotUsed > 70

Event description: An event will be generated when more than 70 percent of the total space in the /tmp directory is in use.

Rearm expression: PercentTotUsed < 45

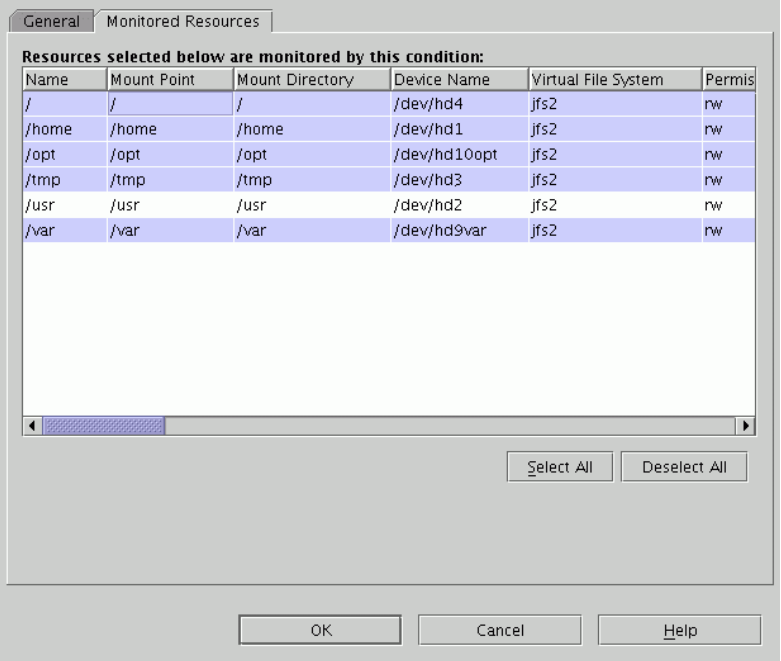
Rearm description: The event will be rearmed when the percent of the space used in the /tmp directory falls below 45 percent.

Severity: Critical

Figure 3-3 Edit the new RMC condition property

Note: If defined, a Rearm expression is evaluated once the Event expression has occurred. Once the Rearm expression tests true, the condition is considered to have “Rearmed.” Once rearmed RMC will revert back to evaluating the Event expression.

4. Click the Monitored Resources tab and select the filesystems you want to monitor as shown in Figure 3-4. From the command line interface it is possible to add additional file systems which are not predefined here, refer to Example 3-10 on page 77.

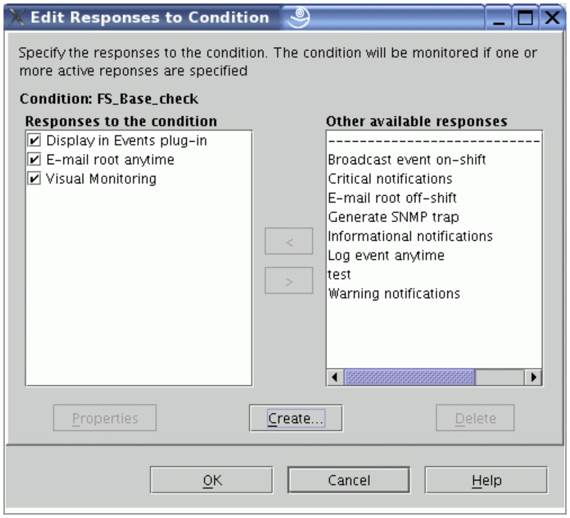


Resources selected below are monitored by this condition:

Name	Mount Point	Mount Directory	Device Name	Virtual File System	Permis
/	/	/	/dev/hd4	jfs2	rw
/home	/home	/home	/dev/hd1	jfs2	rw
/opt	/opt	/opt	/dev/hd10opt	jfs2	rw
/tmp	/tmp	/tmp	/dev/hd3	jfs2	rw
/usr	/usr	/usr	/dev/hd2	jfs2	rw
/var	/var	/var	/dev/hd9var	jfs2	rw

Figure 3-4 Choose the predefined file systems to monitor

5. On the **General** tab you can customize the responses by clicking on **Responses to Condition**. We choose to show the events in WebSM GUI and send an E-mail to root user when both the Event condition and Rearm condition occurs, as shown in Figure 3-5 on page 68.



Edit Responses to Condition

Specify the responses to the condition. The condition will be monitored if one or more active reponses are specified

Condition: FS_Base_check

Responses to the condition

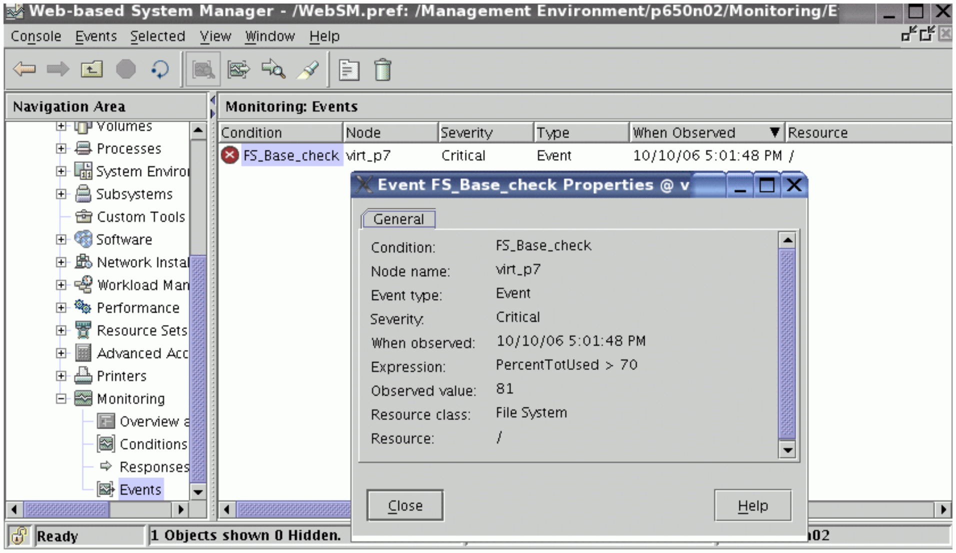
☑ Display in Events plug-in
☑ E-mail root anytime
☑ Visual Monitoring

Other available responses

Broadcast event on-shift
Critical notifications
E-mail root off-shift
Generate SNMP trap
Informational notifications
Log event anytime
test
Warning notifications

Figure 3-5 Responses definition for condition

6. Once you click **OK**, monitoring of the new Condition is activated. In our case there is little free space / (root) file system. Clicking on **Events** option in WebSM Navigation Area we can see the error. Clicking on the Event we see detailed information, as shown in Figure 3-6, and root user receives an E-mail as illustrated in Example 3-2 on page 69.



Web-based System Manager - /WebSM.pref: /Management Environment/p650n02/Monitoring/E

Monitoring: Events

Condition	Node	Severity	Type	When Observed	Resource
FS_Base_check	virt_p7	Critical	Event	10/10/06 5:01:48 PM	/

Event FS_Base_check Properties @ v

General

Condition:	FS_Base_check
Node name:	virt_p7
Event type:	Event
Severity:	Critical
When observed:	10/10/06 5:01:48 PM
Expression:	PercentTotUsed > 70
Observed value:	81
Resource class:	File System
Resource:	/

Figure 3-6 Error Event of the low file system space shown in WebSM with detail

Example 3-2   E-mail message delivered by RMC monitoring

```
rt_p7@root:/var/ct/cfg# mail
Mail [5.2 UCB] [AIX 5.X]  Type ? for help.
"/var/spool/mail/root": 2 messages 1 new 2 unread
 U  1 root              Tue Oct 10 15:51  27/657  "FS_Base_check"
>N  2 root              Tue Oct 10 16:02  26/642  "FS_Base_check"
? 2
Message  2:
From root Tue Oct 10 16:02:48 2006
Date: Tue, 10 Oct 2006 16:02:48 -0500
From: root
To: root
Subject: FS_Base_check

=====================================

Tuesday 10/10/06 16:01:48

Condition Name: FS_Base_check
Severity: Critical
Event Type: Event
Expression: PercentTotUsed > 70

Resource Name: /
Resource Class: IBM.FileSystem
Data Type: CT_INT32
Data Value: 81
Node Name: virt_p7
Node NameList: {virt_p7}
Resource Type: 0
=====================================
```

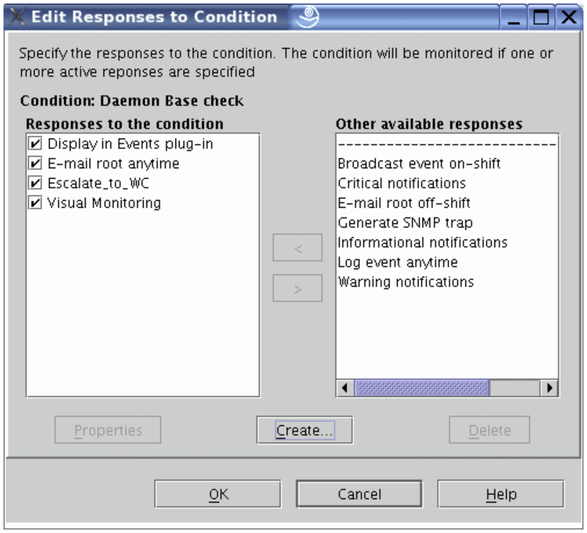
Note that it is possible to create your own custom responses; for example, using the TEC command **wpostemsg** with correct parameters into our custom response script to send message to Tivoli Enterprise™ Console. Response scripts can use variables which are automatically defined by RMC, see a sample in Example 3-3.

Example 3-3   Script variables automatically generated by RMC

```
ERRM_ATTR_NAME='Percent Total Space Used'
ERRM_ATTR_PNAME=PercentTotUsed
ERRM_COND_HANDLE='0x6004 0xffff 0xdccaa515 0x163b9d39 0x101aaca5
0x7ea191b1'
ERRM_COND_NAME='Base FS check'
ERRM_COND_SEVERITY=Critical
```

```
ERRM_COND_SEVERITYID=2
ERRM_DATA_TYPE=CT_INT32
ERRM_ER_HANDLE='0x6006 0xffff 0xdccaa515 0x163b9d39 0x101aaca5
0xa80389cc'
ERRM_ER_NAME='Escalate to Watch Centrum'
ERRM_EXPR='PercentTotUsed > 40'
ERRM_NODE_NAME=virt_p2_persistent
ERRM_NODE_NAMELIST={virt_p2_persistent}
ERRM_RSRC_CLASS_NAME='File System'
ERRM_RSRC_CLASS_PNAME=IBM.FileSystem
ERRM_RSRC_HANDLE='0x6009 0xffff 0xb4bbd78a 0x69f6fee3 0x101aa804
0x066dd212'
ERRM_RSRC_NAME=/tmp
ERRM_RSRC_TYPE=0
ERRM_TIME=1160429751,742711
ERRM_TYPE=Event
ERRM_TYPEID=0
ERRM_VALUE=100
ERRNO=25
```

Figure 3-7 shows how to add custom responses to an existing Condition:



Edit Responses to Condition

Specify the responses to the condition. The condition will be monitored if one or more active reponses are specified

Condition: Daemon Base check

Responses to the condition

☑ Display in Events plug-in
☑ E-mail root anytime
☑ Escalate_to_WC
☑ Visual Monitoring

Other available responses

Broadcast event on-shift
Critical notifications
E-mail root off-shift
Generate SNMP trap
Informational notifications
Log event anytime
Warning notifications

Figure 3-7 Custom response for conditions

It is also possible to specify what time of day or day of week the response should be applicable. This provides flexibility to have different responses for given periods; for example business hours, maintenance windows and weekends.

Note: By default only the root user can delete events that show up in the WebSM GUI; if you require another user to do this task, copy the `/usr/sbin/rsct/cfg/ctrmc.acls` file to the `/var/ct/cfg/` directory and add the stanza regarding your user ID. For example adding the following to the ACSLS section will allow the user martin to also close events:

```
martin@LOCALHOST     *      rw
```

3.1.3 Command line interface for creating RMC sensors

As a complement to the WebSM GUI, RMC provides a command line interface. While command line is generally considered more difficult to use, we found some of the commands more efficient. For example, you can quickly list all defined monitors as shown in Example 3-4.

Example 3-4 Listing the monitored RMC conditions using command line interface

```
virt_p7@root:/home/vanous# lscondition | grep "Monitored and event
monitored"
"Daemon Base check"                 "Monitored and event monitored"
"FS_Base_check"                     "Monitored and event monitored"
```

It is also a simple task to copy existing condition and responses definitions on other nodes. From the command line you can create custom String Selections, for example if you require to monitor a daemon or filesystem which is not already predefined. The first step is to view the content of the predefined condition as shown in Example 3-5 and then create a copy.

Example 3-5 Viewing the predefined condition via command line

```
virt_p7@root:/# lscondition "Daemon Base check"
Displaying condition information:

condition 1:
        Name             = "Daemon Base check"
        MonitorStatus    = "Monitored and event monitored"
        ResourceClass    = "IBM.Program"
        EventExpression  = "Processes.CurPidCount <= 0"
        EventDescription = "An event will be generated whenever the
sendmail daemon is not running."
        RearmExpression  = "Processes.CurPidCount  > 0"
        RearmDescription = "The event will be rearmed when the sendmail
deamon is running."
```

```
        SelectionString  = "ProgramName == \"sendmail\" && Filter ==
\"ruser==\\\"root\\\"\""
        Severity         = "c"
        NodeNames        = {}
        MgtScope         = "l"
virt_p7@root:/#
```

It is also possible to create your own sensor which can be custom script. We prepared the script which defines the new sensor and creates the new monitor, as shown in Example 3-6.

Using command line interface reveals how the RMC monitoring works. There is a clear sequence of steps you need to execute to establish monitoring:

1. Remove previously defined sensor: **rmsensor** command.
2. Create new sensor definition: **mksensor** command.
3. Remove previously defined condition: **rmcondition** command.
4. Create a new condition: **mkcondition** command.
5. Add a response to your newly-defined condition: **mkcondresp** command.
6. Start using new condition with its defined response: **startcondresp** command.

Example 3-6 Script for RMC sensor and condition creation

```
virt_p7@root:/# cat cr_sensor.sh
#!/bin/sh


/usr/sbin/rsct/bin/rmsensor sensor1
/usr/sbin/rsct/bin/mksensor -i 300 sensor1 /home/vanous/sensor1
rmcondition -f "Does the file /tmp/CONTROLFILE exist"
mkcondition -r "IBM.Sensor" \
-e "Int32=1" \
-E "Int32=0" \
-d "Event will be generated when the file /tmp/CONTROLFILE is created."
\
-D "Event will be rearmed when the file /tmp/CONTROLFILE is deleted." \
-s 'Name == "sensor1"' -S "c" "Does the file /tmp/CONTROLFILE exist"

lscondition "Does the file /tmp/CONTROLFILE exist"

mkcondresp "Does the file /tmp/CONTROLFILE exist" "E-mail root anytime"
```

```
startcondresp "Does the file /tmp/CONTROLFILE exist"
```

Note: It is not possible to easily change the monitoring interval of predefined sensors. However when creating custom sensors, you can mandate your desired interval with the **`-i`** parameter of the **`mksensor`** command.

Our newly created monitor checks for a presence of the file `/tmp/CONTROLFILE`. We use a custom sensor script to achieve this, shown in Example 3-7.

Example 3-7 Custom sensor sample

```
#!/bin/sh

 if [ -a /tmp/CONTROLFILE ]
   then
   RET="Int32=1"
   else
   RET="Int32=0"
fi

echo "String=\"check_file\""
echo $RET

exit 0
```

Note: RMC does not validate custom scripts. For example, if your script contains syntax errors, the sensor may not work as expected.

For more information about RMC usage and commands, refer to *RSCT Administration Guide*, SA22-7889, and *RSCT for AIX 5L Technical Reference*, SA22-7890.

3.2 RMC monitoring in a peer domain

While RMC monitoring on a standalone machine can be useful, it is still contained locally on that machine, but, if you have multiple machines, management will become time consuming. The solution is defining monitors on multiple nodes, managed from a single node, using RSCT peer domains.

Peer domain monitoring is similar to the standalone RMC monitoring, however there are some points to note:

- You must create a RSCT peer domain (RPD), and bring the domain online, prior to using RMC monitoring.
- A shell variable sets the management scope to the RSCT peer domain (RSCT command execution scope).In a Korn shell you can set it using:

  ```
  export CT_MANAGEMENT_SCOPE=2
  ```

- There are minor differences in the command line usage. Some examples are detailed later in this chapter.

Peer domain RMC monitoring also has its own limitations:

- It is not possible to use WebSM GUI for sensor, conditions, responses definition or event viewing.
- Peer domains should not include Managed nodes that are not all in the same Management domain. Therefore one peer domain should not include managed nodes from two separate CSM domains.
- Management domain servers cannot be in the same peer domain as their Managed nodes. This means that the CSM server cannot be in the same peer domain as CSM managed nodes.

Notes:

- An HACMP cluster creates a form of peer domain, but unlike an RSCT peer domain (RPD), HACMP does not store the cluster configuration in RSCT registry, rather retrieves config data from HACMP’s ODM. Also the daemons’ names are different (HACMP: topsvcs, grpsvcs; RPD:cthats, cthags), and the HACMP topology and group services do not use RSCT cluster security services (CTCAS).
- While there are no obvious reasons why the nodes part of an RPD cannot be also part of an HACMP cluster, currently there is no functional reason (application) that can benefit this configuration.

3.2.1 Scenario using peer domain RMC monitoring

In our scenario we setup an initial peer domain between two AIX 5L V5.3 micro-partitions on our IBM System p5 p550 managed by an IVM server, see the diagram in Figure 3-8 on page 75.

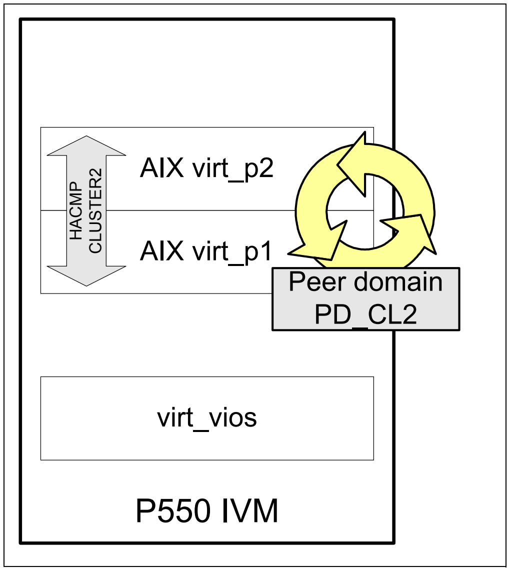


Figure 3-8 Scenario RMC peer domain monitoring with 2 HACMP nodes

1. We prepare nodes which will become domain members by running the following command on BOTH nodes:

   ```
   # preprpnode virt_p1_persistent virt_p2_persistent
   ```

   > **Note:** Correct and identical name resolution on both nodes is a prerequisite for successful peer domain creation.

2. We create the peer domain using the following command:

   ```
   # mkrpdomain PD_CL2 virt_p1_persistent virt_p2_persistent
   ```

3. Successful domain creation can be validated with the **lsrpdomain** and **lsrpnode** commands, as shown in Example 3-8.

Example 3-8 Peer domain validation

```
[virt_p1][/home/vanous]> lsrpdomain
Name   OpState RSCTActiveVersion MixedVersions TSPort GSPort
PD_CL2 Online  2.4.5.3           No            12347  12348
[virt_p1][/home/vanous]> lsrpnode
Name               OpState RSCTVersion
virt_p1_persistent Online  2.4.5.3
virt_p2_persistent Online  2.4.5.3
[virt_p1][/home/vanous]>
```

4. We create a monitor to watch the base file systems. The procedure is shown in Example 3-9. Although the command is run only on the virt_p1 node, the file systems will be monitored on both virt_p1 and virt_p2. Escalation of the conditions will be made only on the virt_p1 node as it is defined by **-p** parameter. This parameter specifies which node in the domain which will store the definition. It also means that the script for our custom response **/usr/local/bin/tell_to_watch_centrum.sh** must be present only on virt_p1 node.

   Note the differences from the standalone RMC monitoring:

   – Export the *CT_MANAGEMENT_SCOPE* variable.
   – Using **-mp** parameter of the **mkcondition** command. In this case, the **m** parameter denotes a management domain.

Example 3-9 Script for creating monitor of the base file systems

```
#!/bin/sh

export CT_MANAGEMENT_SCOPE=2

rmcondition -f "Base FS check"
mkcondition -r "IBM.FileSystem" \
-mp \
-n virt_p1_persistent,virt_p2_persistent \
-p virt_p1_persistent \
-e "PercentTotUsed > 60" \
-E "PercentTotUsed < 40" \
-d "Low File system space" \
-D "Low File system space problem resolved" \
-s 'Name == "/" || Name == "/tmp" || Name == "/var"' -S "c" "Base FS
check"

lscondition "Base FS check"

rmresponse -q "Escalate to Watch Centrum"
mkresponse -n "Escalate WC" -d 1-7 -t 0000-2400 -s
/usr/local/bin/tell_to_watch_centrum.sh -e b -r 1 "Escalate to Watch
Centrum"
stopcondresp -q "Base FS check"
mkcondresp "Base FS check" "E-mail root anytime" "Escalate to Watch
Centrum"
startcondresp "Base FS check"
```

5. Next we define a monitor for the shared file systems which migrate between the HACMP nodes as shown in Example 3-10 on page 77.

Note: When a resource is not present on a node where it is monitored, for example, a file system which is not mounted, you do not actually receive any error message. If a file system is in an error condition state and you unmount it, you will trigger the rearm condition event. However, this results in the Condition state shown as “*Running: Yes with error*” in the WebSM.

Example 3-10 Script for creating monitors of the HACMP shared file systems

```
#!/bin/sh

export CT_MANAGEMENT_SCOPE=2

rmcondition -f "CL2 FS check"
mkcondition -r "IBM.FileSystem" \
-mp \
-n virt_p1_persistent,virt_p2_persistent \
-p virt_p1_persistent \
-e "PercentTotUsed > 50" \
-E "PercentTotUsed < 10" \
-d "Low File system space" \
-D "Low File system space problem resolved" \
-s 'Name == "/fs1" || Name == "/fs2"' -S "c" "CL2 FS check"

lscondition "CL2 FS check"

stopcondresp -q "CL2 FS check"
mkcondresp "CL2 FS check" "E-mail root anytime" "Escalate to Watch 
Centrum"
startcondresp "CL2 FS check"
```

6. We create our custom sensor for detecting an unstable HACMP state. The sensor is a script which uses **clstat** command and we **grep** its output for specific keywords as shown in Example 3-12 on page 78. The monitor is created as shown in Example 3-11. Note that we create sensor on both of the nodes to detect the error condition.

Example 3-11 Script for creating an HACMP status monitor

```
#!/bin/sh

export CT_MANAGEMENT_SCOPE=2

/usr/sbin/rsct/bin/rmsensor sensor_check_hacmp
/usr/sbin/rsct/bin/mksensor -n virt_p1_persistent -i 300 
sensor_check_hacmp  /usr/local/bin/sensor_check_hacmp.sh
```

```
/usr/sbin/rsct/bin/mksensor -n virt_p2_persistent -i 300
sensor_check_hacmp  /usr/local/bin/sensor_check_hacmp.sh
rmcondition -f "HACMP Health check"
mkcondition -r "IBM.Sensor" \
-mp \
-n virt_p2_persistent,virt_p1_persistent \
-p virt_p1_persistent \
-e "Int32=1" \
-E "Int32=0" \
-d "An event will be generated when the HACMP status is not OK" \
-D "The event will be rearmed when HACMP is stable again" \
-s 'Name == "sensor_check_hacmp"' -S "c" "HACMP Health check"
# -s 'Name == ""' -S "c" "HACMP Health check"

lscondition "HACMP Health check"
stopcondresp -q "HACMP Health check"

lsresponse "HACMP on CLUSTER2 Error"


mkcondresp "HACMP Health check" "E-mail root anytime" "Escalate to
Watch Centrum"
startcondresp "HACMP Health check"
```

The sensor script shown in Example 3-12 must be present on both of the nodes with appropriate permissions.

Example 3-12 Sensor script for monitoring HACMP status

```
#!/bin/ksh

 if [ `/usr/es/sbin/cluster/clstat -o | grep -qiE
"error|down|unstable|running";echo $?` -eq 0 ]
   then
   RET="Int32=1"
   else
   RET="Int32=0"
fi

/usr/bin/echo "String=\"check_hacmp\""
echo $RET
exit 0
```

3.2.2 Extending the peer domain

In our sample scenario we have a requirement to extend our monitoring to another node - virt_p3 - as shown in Figure 3-9.

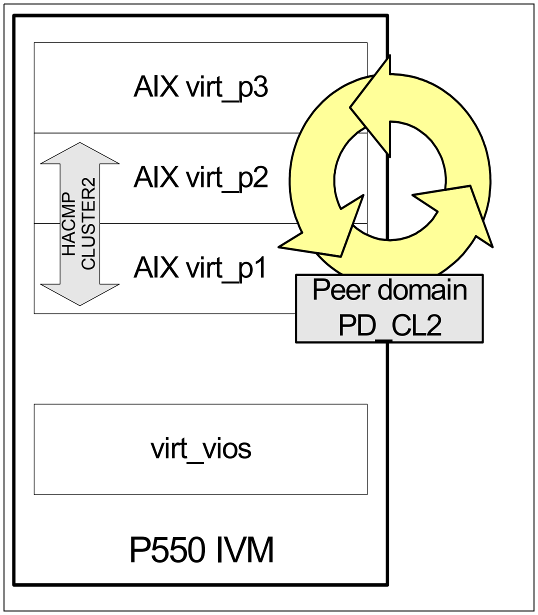


Figure 3-9 Scenario RMC peer domain monitoring with 3 nodes

To complete this task we prepare and add the node to the existing peer domain. This process is similar to the initial setup.

1. We run the following command on virt_p3:

   `# preprpnode virt_p1_persistent virt_p2_persistent.`

2. Add the node to the peer domain using:

   `# addrpnode virt_p3.`

3. Validate the node addition was successful and start the node virt_p3 as shown in Example 3-13:

Example 3-13 Peer domain after node addition and starting the new node

```
[virt_p1][/]> addrpnode virt_p3
[virt_p1][/]> lsrpdomain
Name   OpState RSCTActiveVersion MixedVersions TSPort GSPort
PD_CL2 Online  2.4.5.3           No            12347  12348
[virt_p1][/]> lsrpnode
Name                OpState RSCTVersion
```

```
virt_p3           Offline 2.4.5.3
virt_p1_persistent Online  2.4.5.3
virt_p2_persistent Online  2.4.5.3
[virt_p1][/]> startrpnode virt_p3
[virt_p1][/]> lsrpnode
Name              OpState RSCTVersion
virt_p3           Online  2.4.5.3
virt_p1_persistent Online  2.4.5.3
virt_p2_persistent Online  2.4.5.3
[virt_p1][/]>
```

If you get an error message while running the **addrpnode** command (similar to Example 3-14), check the communications groups by **lscomg** command. Nodes you want to add to the domain must be reachable from the existing communication group network.

Example 3-14 Peer domain communication groups

```
virt_p1][/]> addrpnode virt_p3
2632-057 The requested configuration change is not allowed since it
will result in network partition(s) that will prevent some nodes from
being brought online.
[virt_p1][/]> lscomg -i CG2
Name NodeName           IPAddress    Subnet      SubnetMask
en1  virt_p2_persistent 172.16.51.74 172.16.51.0 255.255.255.0
en1  virt_p1_persistent 172.16.51.73 172.16.51.0 255.255.255.0
[virt_p1][/]> lscomg -i CG1
Name NodeName           IPAddress    Subnet      SubnetMask
en0  virt_p1_persistent 172.16.50.73 172.16.50.0 255.255.255.0
en0  virt_p2_persistent 172.16.50.74 172.16.50.0 255.255.255.0
```

3.2.3 Other considerations using RMC monitoring

As RMC ships as part of AIX 5L, there is no extra licensing cost associated with its use. The functions are available, you simply need to invest the time to implement it as per your requirements. RMC can be leveraged to provide functional monitoring for smaller environments.

RMC monitoring can also be used as middle layer to an existing monitoring infrastructure; so alerts can be easily integrated to Tivoli monitored environments.

There is no GUI to view current node state for a peer domain. At the time of writing WebSM does not provide this functionality. The question comes: to see what is the current node state, is it possible to look at the EventFlags element of the **LastEvent** somehow?. The event flags indicates if the event notification was generated from the re-arm expression. If this flag is not set in the LastEvent, and there is re-arm expression, then the next event will be generated by the re-arm expression, unless some error occurs.

The **lsaudrec** command is recommended for problem determination of RMC monitoring and related activities. This command lists records from the RMC audit log. Records are added to the audit log by the event response subsystem.

While based on AIX 5L V5.1, *A Practical Guide for Resource Monitoring and Control (RMC)*, SG24-6615, contains practical information and some implementation examples.

3.3 CSM implementation in an AIX environment

Our scenario becomes large enough to warrant implementing CSM management software. As shown in Figure 3-10 on page 82, our environment is a mixture of IBM System p4 and IBM System p5 machines (HMC-managed, IVM-managed and standalone) hosting dedicated LPARs, micro-partitions and VIOS LPARs. There are two IBM System p4 p630 servers, from which one will become our CSM Management Server, while the other is an existing, operational NIM server.

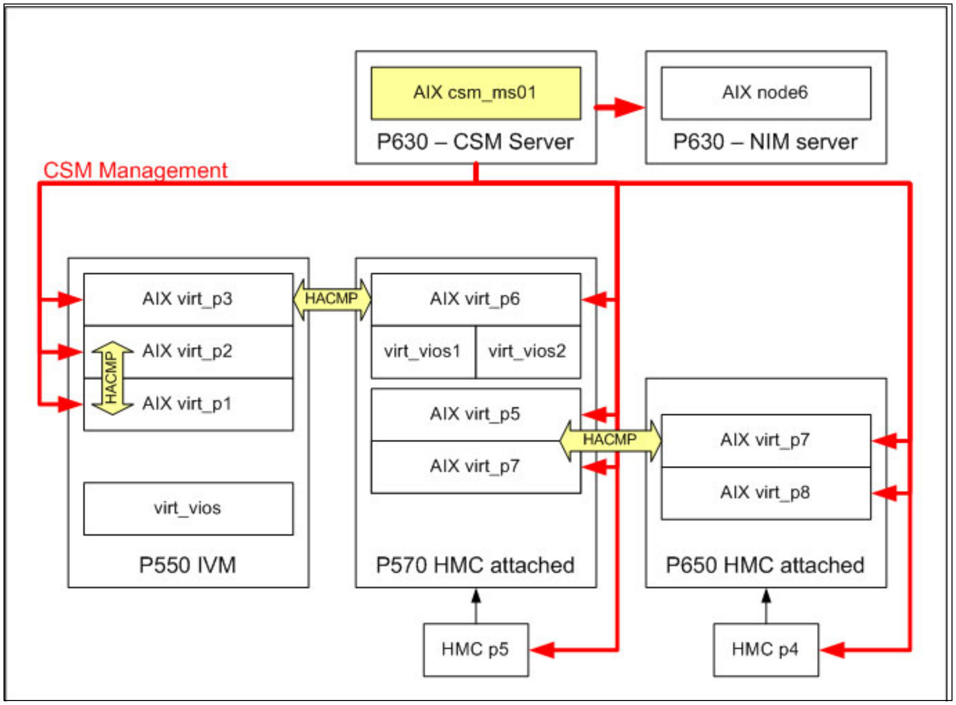


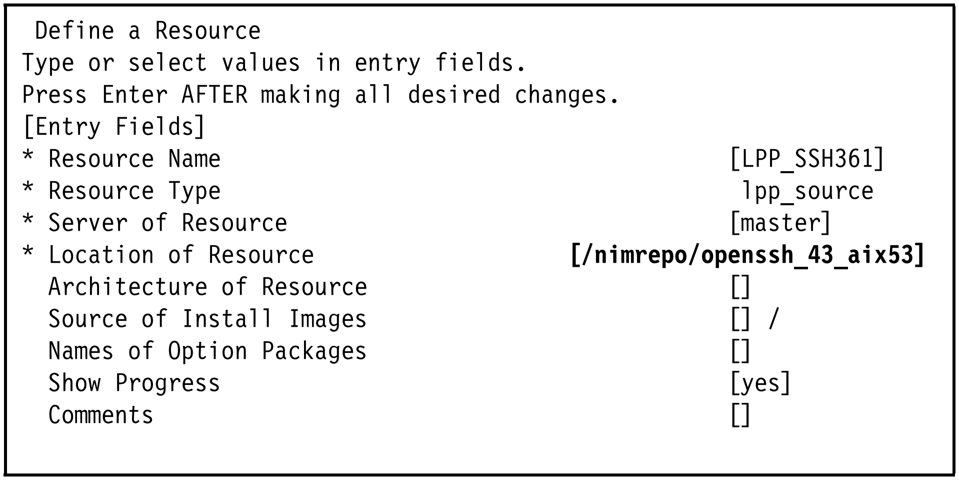
Figure 3-10 CSM management scenario

3.3.1 CSM Version 1.5 installation using NIM

The steps required for installation are documented in *IBM Cluster Systems Management for AIX 5L and Linux 1.5 Planning and Installation Guide*, SA23-1344. In this section we highlight specific steps, based on our particular scenario.

After creating the /csminstall filesystem on our CSM Management Server (MS) we used our existing NIM server to perform the majority of the installation tasks.

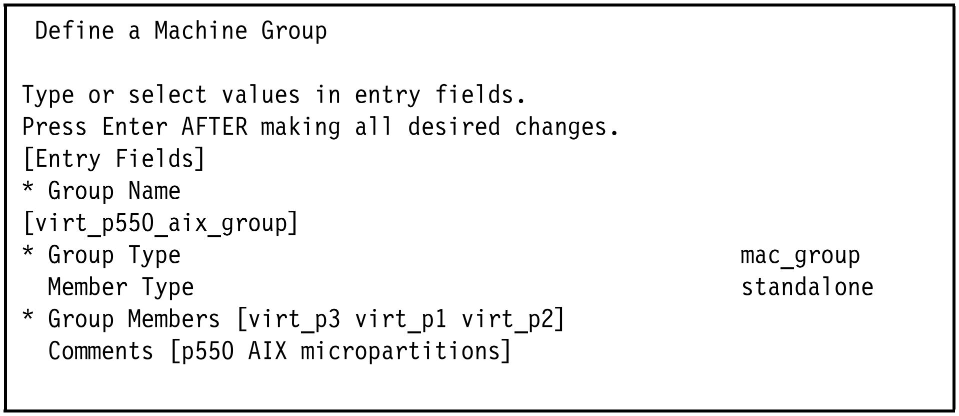
1. We wanted to use SSH for remote command execution. For this we had to install the required SSH/SSL software onto all nodes, including the one that would become our MS. To achieve this we simply created a NIM lpp_source containing the SSH required packages. We copied all of the SSH software including SSL prerequisites into the /nimrepo/openssh_43_aix53 directory on the NIM server and then defined a new lpp_source resource as shown in Figure 3-11 on page 83.



```
 Define a Resource
Type or select values in entry fields.
Press Enter AFTER making all desired changes.
[Entry Fields]
* Resource Name                                          [LPP_SSH361]
* Resource Type                                           lpp_source
* Server of Resource                                     [master]
* Location of Resource                       [/nimrepo/openssh_43_aix53]
  Architecture of Resource                               []
  Source of Install Images                               [] /
  Names of Option Packages                               []
  Show Progress                                          [yes]
  Comments                                               []
```

Figure 3-11 Defining an SSH NIM resource

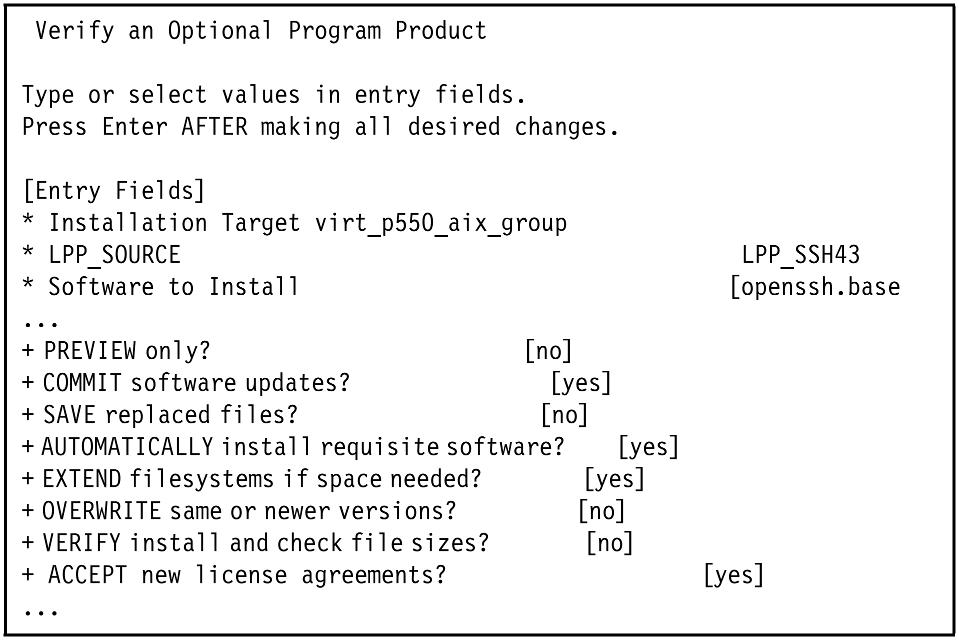
2. To simplify the installation we created NIM machine groups; this allows operations to be executed against all group members concurrently, as opposed to individually. A NIM machine group can be created using the smitty fastpath **`smitty nim_mkgrp_standalone`**, as show in Figure 3-12.



```
 Define a Machine Group

Type or select values in entry fields.
Press Enter AFTER making all desired changes.
[Entry Fields]
* Group Name
[virt_p550_aix_group]
* Group Type                                              mac_group
  Member Type                                             standalone
* Group Members [virt_p3 virt_p1 virt_p2]
  Comments [p550 AIX micropartitions]
```

Figure 3-12 Creating a NIM machine group

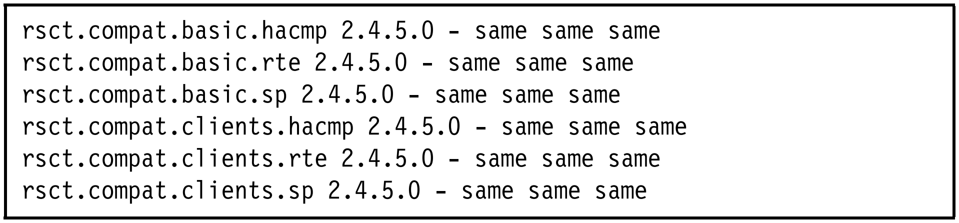
3. We install the SSH software via a NIM installation operation. The following example shows concurrent NIM installation on all of the virt_p550_aix_group group members using **`smitty nim_inst_latest`** fastpath, as shown in Figure 3-13 on page 84.



```
 Verify an Optional Program Product

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

[Entry Fields]
* Installation Target virt_p550_aix_group
* LPP_SOURCE                                                  LPP_SSH43
* Software to Install                                        [openssh.base
...
+ PREVIEW only?                              [no]
+ COMMIT software updates?                     [yes]
+ SAVE replaced files?                        [no]
+ AUTOMATICALLY install requisite software?     [yes]
+ EXTEND filesystems if space needed?         [yes]
+ OVERWRITE same or newer versions?          [no]
+ VERIFY install and check file sizes?        [no]
+ ACCEPT new license agreements?                       [yes]
...
```

Figure 3-13 SSH installation through NIM on multiple nodes

4. We copied all necessary CSM software and PTFs, including RPM software into the directory /nimrepo/csm/csm1513_server on the NIM server. By using a similar process to Step1, we created another lpp_source resource (just for CSM). Using this new lpp_source we installed CSM onto our MS.
5. One prerequisite of CSM is correct levels of the various RSCT filesets. We used a NIM software comparison report to check the levels across our nodes. Using the **smitty invcmp** fastpath and targeting the command on our AIX NIM node group produces output similar to Figure 3-14.



```
rsct.compat.basic.hacmp 2.4.5.0 - same same same
rsct.compat.basic.rte 2.4.5.0 - same same same
rsct.compat.basic.sp 2.4.5.0 - same same same
rsct.compat.clients.hacmp 2.4.5.0 - same same same
rsct.compat.clients.rte 2.4.5.0 - same same same
rsct.compat.clients.sp 2.4.5.0 - same same same
```

Figure 3-14 NIM software inventory report

6. CSM client is installed by default with the base AIX 5L installation. Using our CSM-specific lpp_source from Step4, we updated the existing CSM filesets on the nodes. This can be achieved with the **smitty nim_update_all** fastpath.

Note: When software is Installed or updated using NIM, the installation output messages are not written into the smit.log file on the NIM client. You can check the installation progress using the **nimlog** command. After the installation you can use the **lslpp -h** command to get the installation history for a given fileset.

3.3.2 Basic customizing, creating nodes and groups

1. We performed the base CSM configuration on the MS: accepting the license, configuring remote command preferences and establishing HMC communication using **systemid** command. Such steps are required before node definition. In Example 3-15 we illustrate summary output, just for reference. More verbose output can be found in *IBM Cluster Systems Management for AIX 5L and Linux 1.5 Planning and Installation Guide*, SA23-1344.

Example 3-15 Summary CSM configuration

```
csmconfig RemoteShell=/usr/bin/ssh
csmconfig RemoteCopyCmd="/usr/bin/scp"
csmconfig -L
...
systemid hmcitso hscroot
...
systemid hmcp570 hscroot
...
probemgr -p ibm.csm.ms -l 0
...
cat 15-define_all_p550_aix.cfg
default:
ManagementServer=csm_ms01
InstallOSName=AIX
virt_p1:
HWControlNodeId=virt_p1
Hostname=virt_p1
virt_p2:
HWControlNodeId=virt_p2
Hostname=virt_p2
virt_p3:
HWControlNodeId=virt_p3
Hostname=virt_p3
definenode 15-define_all_p550_aix.cfg
...
updatenode -P
```

```
...
```

2. The **updatenode command** is used to configure a given node as “managed” by CSM. During this process CSM configures and exchanges SSH keys with the node. After running the **updatenode** command and configuring HMC communication it is possible to use **csmstat** command to query node status; as shown in Example 3-16.

Example 3-16 Output of the **csmstat** *command*

```
root@csm_ms01:/home/vanous/w# csmstat
-------------------------------------------------------------------
Hostname HWControlPoint  Status PowerStatus Network-Interfaces
-------------------------------------------------------------------
node6                      on       not configured en2-Online
virt_p1                    on       not configured en0-Online
virt_p2                    on       not configured en0-Online
virt_p3                    on       not configured en0-Online
virt_p4  hmcp570           on       on             en0-Online
virt_p5  hmcp570           on       on             en0-Online
virt_p6  hmcp570           on       on             en0-Online
virt_p7  hmcitso           on       on             en0-Online
```

3. Communication setup between CSM MS and non-node devices such as HMCs, IVM servers, VIOS, network switches and other devices is achieved by the **definehwdev** command. The benefit of defining non-node devices is the possibility to run remote commands on them and monitor their base status. Example 3-17 shows how we defined a POWER4 HMC.

Example 3-17 Non-node devices definition into CSM cluster

```
root@csm_ms01:/# cat Power4_HMC.cfg
hmcitso:
        DeviceType=hmc

root@csm_ms01:/# definehwdev -f Power4_HMC.cfg
chhwdev -d hmcitso RemoteShell=/usr/bin/ssh RemoteCopyCmd=/usr/bin/scp
RemoteShellUser=hscroot
updatehwdev -k -d hmcitso
...
```

4. Definition of POWER5 HMC is very similar, but not the same; as seen by the different usage of the **updatehwdev** command on Example 3-18 on page 87.

Example 3-18 Defining POWER5 HMC machine into CSM cluster

```
root@csm_ms01:/# definehwdev -d hmcp570 RemoteShell=/usr/bin/ssh \
> RemoteCopyCmd=/usr/bin/scp RemoteShellUser=hscroot DeviceType=HMC
Defining CSM Devices:
Defining Device "hmcp570"
root@csm_ms01:/# updatehwdev -kId hmcp570
...
```

5. System p4 HMC and System p5 HMC are integrated in a different way in CSM. As you can see in Example 3-19, the HMC status is reported slightly different for each HMC. The POWER5 HMC definition includes RSCT integration, which is why the hmcp570 (System p5 HMC) has status of `Managed`.

Example 3-19 Output of the lshwdev command

```
root@csm_ms01:/# lshwdev -a Mode
hmcitso:  PreManaged
hmcp570:  Managed
```

> **Note:** Once the non-node devices are defined in CSM, their status is regularly checked by the **`ping`** command. You can change the polling frequency by setting DeviceStatusFrequency attribute of the **`csmconfig`** command.

6. We create node groups to fit our given requirements. We want to perform specific operations only on our HACMP cluster nodes, for example, we use CFM to distribute HACMP-specific files and the **`rpower`** command. To facilitate this, we created the groups as shown in Example 3-20 (the final command parameter is the group name). Observe also that we have a special group for all of the AIX nodes belonging to our department.

Example 3-20 CSM node group creation

```
nodegrp -a virt_p1,virt_p2,virt_p3,virt_p4,virt_p5,virt_p6,virt_p7
gr_virt_aix
nodegrp -a virt_p1,virt_p2,virt_p3 gr_p550_aix
nodegrp -a virt_p4,virt_p5,virt_p6 gr_p570_aix
nodegrp -a virt_p4,virt_p5,virt_p7 gr_cluster1
nodegrp -a virt_p1,virt_p2 gr_cluster2
nodegrp -a virt_p3,virt_p6 gr_cluster3
```

If you are familiar with previous versions of CSM, you may find Table 3-1 on page 88 helpful. The table details the relationship between various types of System p hardware supported with CSM Version 1.6.0. You can note the slight differences between p4 and p5 releases of HMCs, and the subtle difference between VIOS and IVM.

Table 3-1 CSM support for our test environment

	Type	Mode	Listed in csmstat	Listed in lsnode or hwdev
AIX LPAR	Node	Managed	Y	lsnode
Linux LPAR[a]	Node	Managed	Y	lsnode
p4 HMC	Device	PreManaged	N	lshwdev
p5 HMC	Device	Managed	N	lshwdev
IVM	Device	Managed	N	lshwdev
VIOS[b]	Device	Managed	N	lshwdev

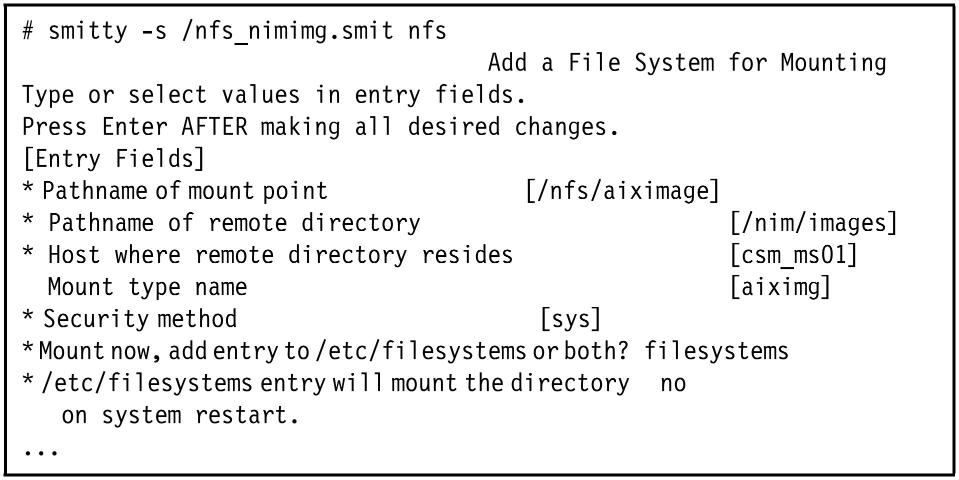
a. For a CSM-supported version of Linux
b. VIOS Version 1.3.0 (therefore same for the previous IVM row)

3.3.3 Using DCEM as a backup strategy in an CSM cluster

In this section we demonstrate the advantages of using DCEM, parallel commands, file distribution, and consolidation of the backup procedures. The goal is to have a user friendly interface for creating AIX operating system image backups using naming convention, facilitating and speeding up restore process.

1. First, we configure our NIM server as the backup repository for all of the backup images; therefore the images will already be hosted on the NIM server, making for efficient restores. For this purpose, we create a new filesystem and export it via NFS to the client nodes (not shown here).

2. Next, we add the NFS filesystem on the client. For easier mounting we use *Mount type* name option. To create NFS file system on all nodes we first generate a command script using SMIT command redirection (the `-s` parameter) to a file (see Figure 3-15 on page 89), then we distribute and execute the script on other nodes.



```
# smitty -s /nfs_nimimg.smit nfs
                                   Add a File System for Mounting
Type or select values in entry fields.
Press Enter AFTER making all desired changes.
[Entry Fields]
* Pathname of mount point                 [/nfs/aiximage]
* Pathname of remote directory                          [/nim/images]
* Host where remote directory resides                   [csm_ms01]
  Mount type name                                       [aiximg]
* Security method                          [sys]
* Mount now, add entry to /etc/filesystems or both?  filesystems
* /etc/filesystems entry will mount the directory   no
   on system restart.
...
```

Figure 3-15 Using smitty command redirection to a file

3. Then test the newly created NFS file system on our test node as shown on Example 3-21.

Example 3-21 Mounting a NFS file system using Mount type name option

```
root@virt_p7:/# mount -t aiximg
root@virt_p7:/# mount |grep nfs
  node       mounted        mounted over    vfs       date
options
-------- ---------------  ----------  ------ ------- ----------
csm_ms01 /nim/images      /nfs/aiximage    nfs3   Oct 16 12:36
bg,hard,intr,sec=sys,rw
```

4. In this step we copy the redirected smitty script file to all remaining nodes; then remotely execute it such that the NFS file system creation will be replicated across the cluster, as shown in Example 3-22.

Example 3-22 Using parallel and remote commands to distribute and execute a script

```
root@csm_ms01:/# dcp -N gr_virt_aix /tmp/nfs_nimimg.smit /tmp
root@csm_ms01:/# dsh -n virt_p1,virt_p2,virt_p3,virt_p4,virt_p5,virt_p6
/tmp/nfs_nimimg.smit
root@csm_ms01:/# dsh -a mkdir -p /nfs/aiximage
root@csm_ms01:/# dsh -n virt_p1,virt_p2,virt_p3,virt_p4,virt_p5,virt_p6
/tmp/nfs_nimimg.smit
```

5. Now we are ready to distribute the backup script to all nodes using CFM. We place the backup script and its configuration file with appropriate extension

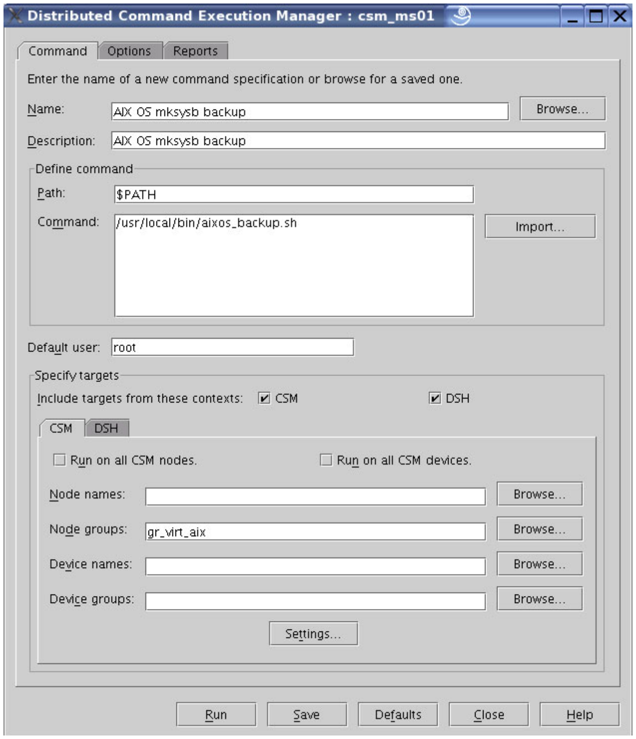
into the /cfmroot directory (on our MS), so the absolute path of the files is: /cfmroot/usr/local/etc/aixos_backup.ini._gr_virt_aix and /cfmroot/usr/local/bin/aixos_backup.sh._gr_virt_aix.
Then we run **cfmupdatenode** to distribute files to the nodes, as shown in Example 3-23. You can find the actual script in A.1, "DCEM backup in CSM cluster" on page 198.

Example 3-23 Distributing files via CFM

```
root@csm_ms01:/cfmroot# cfmupdatenode -a
node6: No CFM files were transferred to this machine.
virt_p6: At least 1 CFM file was transferred to this machine.
virt_p1: At least 1 CFM file was transferred to this machine.
virt_p3: At least 1 CFM file was transferred to this machine.
virt_p2: At least 1 CFM file was transferred to this machine.
virt_p4: At least 1 CFM file was transferred to this machine.
virt_p5: At least 1 CFM file was transferred to this machine.
virt_p7: At least 1 CFM file was transferred to this machine.
```

Our environment is now ready to run the backup utility. In our case, to avoid overloading our NIM server, we choose to run the job serially across the nodes. Using the DCEM dialog enables the backup script to be run even by unexperienced users, however it is also possible to use the **dcem** command to achieve the same end result.

6. To use DCEM we have to first prepare and save the job definition. The job can then be used by either WebSM GUI or **dcem** command. The preparation step is shown in Figure 3-16 on page 91.



Enter the name of a new command specification or browse for a saved one.

Name: AIX OS mksysb backup

Description: AIX OS mksysb backup

Define command

Path: $PATH

Command: /usr/local/bin/aixos_backup.sh

Default user: root

Specify targets

Include targets from these contexts: ☑ CSM ☑ DSH

□ Run on all CSM nodes. □ Run on all CSM devices.

Node names:

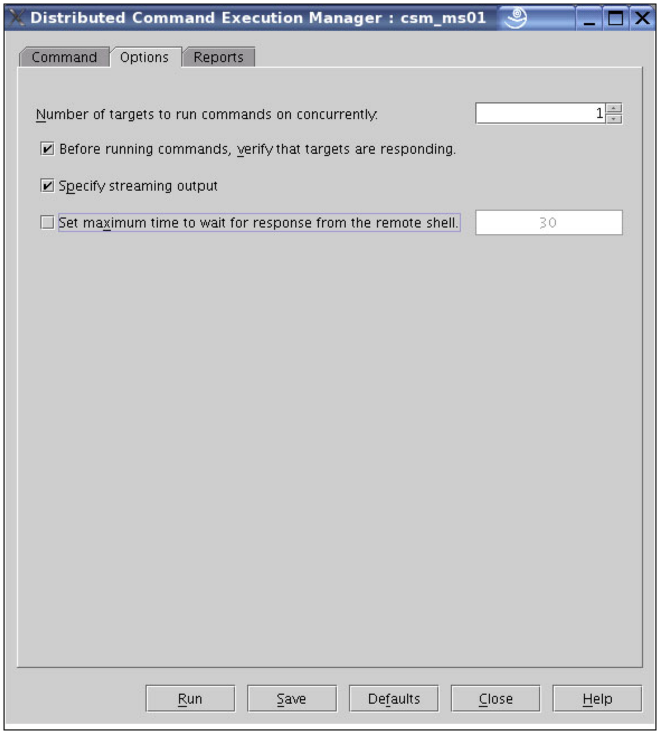
Node groups: gr_virt_aix

Device names:

Device groups:

Figure 3-16 Preparation of the DCEM command

7. We have to adjust the command parameters to run the job serially. On the **Options** tab we set the **Number of targets to run commands on concurrently** to **1** as shown in Figure 3-17 on page 92.


Distributed Command Execution Manager : csm_ms01

Number of targets to run commands on concurrently: 1

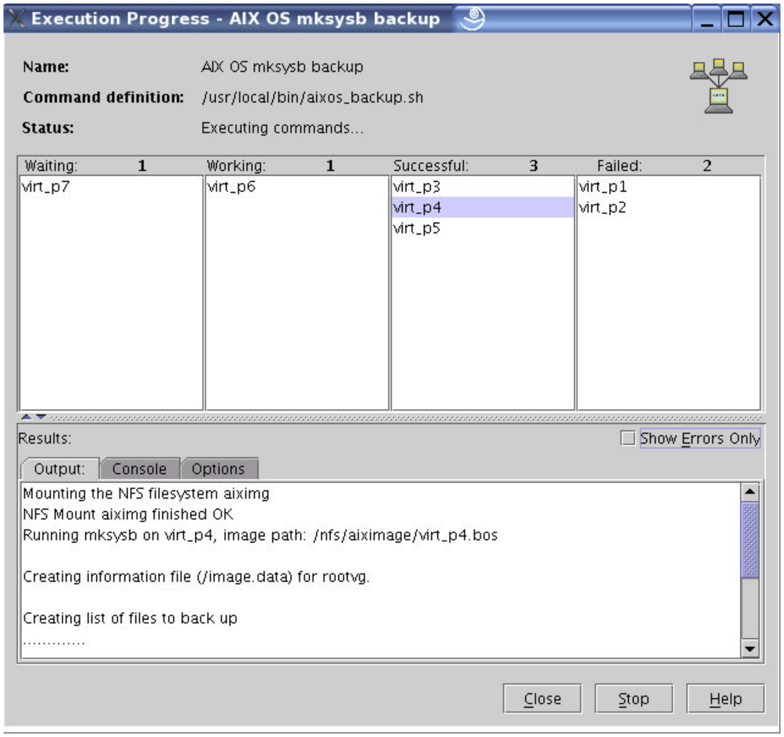
☑ Before running commands, verify that targets are responding.

☑ Specify streaming output

□ Set maximum time to wait for response from the remote shell. 30

Figure 3-17 DCEM command options

8. Now we use WebSM to run the command and monitor the output. Note that there is also an HTML format of the command output. In Figure 3-18 on page 93 you see the command progress, with a list of waiting nodes and those which finished successfully or failed.



Name: AIX OS mksysb backup
Command definition: /usr/local/bin/aixos_backup.sh
Status: Executing commands...

Waiting: 1	Working: 1	Successful: 3	Failed: 2
virt_p7	virt_p6	virt_p3	virt_p1
		virt_p4	virt_p2
		virt_p5	

Results: □ Show Errors Only

```
Mounting the NFS filesystem aiximg
NFS Mount aiximg finished OK
Running mksysb on virt_p4, image path: /nfs/aiximage/virt_p4.bos

Creating information file (/image.data) for rootvg.

Creating list of files to back up
.............
```

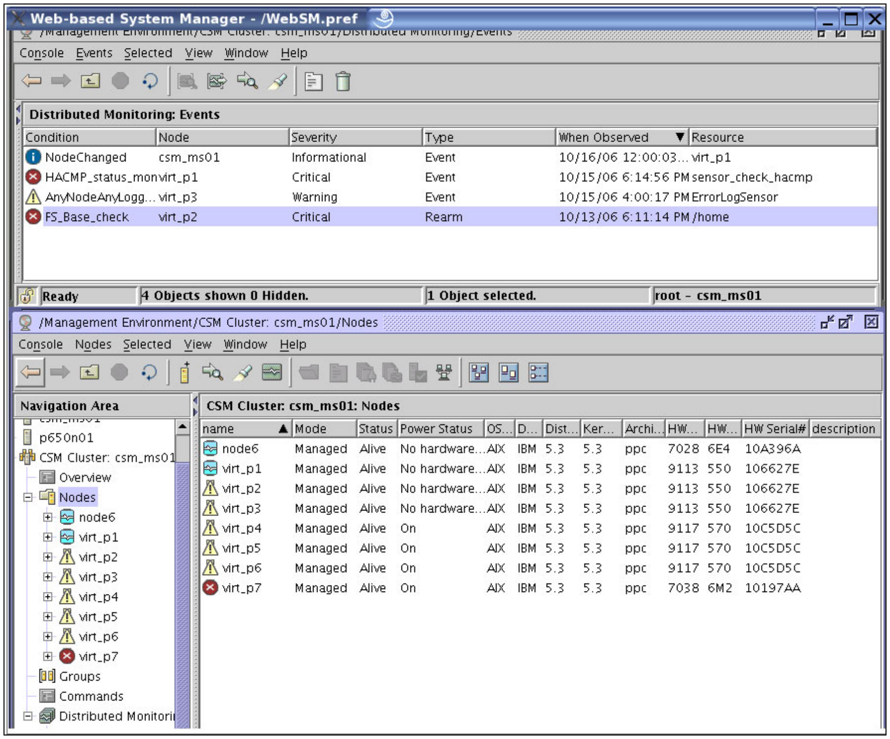
Figure 3-18 DCEM command progress

We found DCEM to be very intuitive, for example you can get node specific output by simply clicking on a given node. There are many default predefined jobs in the DCEM, for example creation of HMC users. Try clicking on the **Sample Commands** in the **Browse Command** window to see what is available.

3.3.4 Monitoring CSM the nodes

CSM provides a broad set of predefined monitors. These monitors can be configured and controlled through WebSM. Some of the monitor tasks are (see also Figure 3-19 on page 94):

- Checking node status (CSM status).
- Monitoring incoming events (with details) for all CSM cluster nodes.
- Event acknowledgment
- Filtering incoming events for readability, for example according to their priority.
- Defining new conditions and responses for use cluster nodes.


Web-based System Manager - /WebSM.pref
/Management Environment/CSM Cluster: csm_ms01/Distributed Monitoring/Events
Console Events Selected View Window Help

Distributed Monitoring: Events

Condition	Node	Severity	Type	When Observed	Resource
NodeChanged	csm_ms01	Informational	Event	10/16/06 12:00:03...	virt_p1
HACMP_status_mon	virt_p1	Critical	Event	10/15/06 6:14:56 PM	sensor_check_hacmp
AnyNodeAnyLogg...	virt_p3	Warning	Event	10/15/06 4:00:17 PM	ErrorLogSensor
FS_Base_check	virt_p2	Critical	Rearm	10/13/06 6:11:14 PM	/home

Ready | 4 Objects shown 0 Hidden. | 1 Object selected. | root - csm_ms01

CSM Cluster: csm_ms01: Nodes

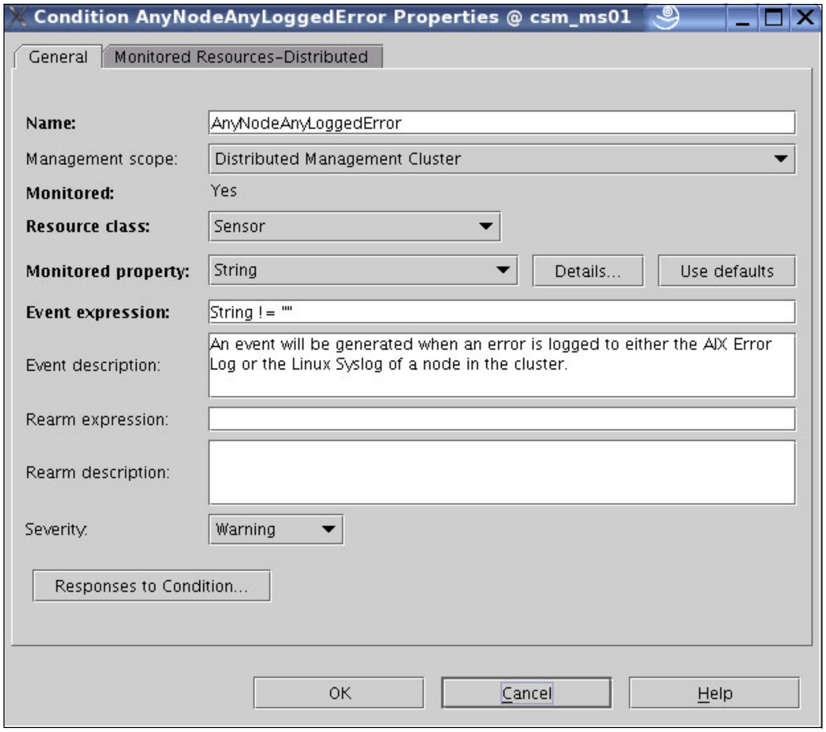
name	Mode	Status	Power Status	OS...	D...	Dist...	Ker...	Archi...	HW...	HW...	HW Serial#	description
node6	Managed	Alive	No hardware...	AIX	IBM	5.3	5.3	ppc	7028	6E4	10A396A	
virt_p1	Managed	Alive	No hardware...	AIX	IBM	5.3	5.3	ppc	9113	550	106627E	
virt_p2	Managed	Alive	No hardware...	AIX	IBM	5.3	5.3	ppc	9113	550	106627E	
virt_p3	Managed	Alive	No hardware...	AIX	IBM	5.3	5.3	ppc	9113	550	106627E	
virt_p4	Managed	Alive	On	AIX	IBM	5.3	5.3	ppc	9117	570	10C5D5C	
virt_p5	Managed	Alive	On	AIX	IBM	5.3	5.3	ppc	9117	570	10C5D5C	
virt_p6	Managed	Alive	On	AIX	IBM	5.3	5.3	ppc	9117	570	10C5D5C	
virt_p7	Managed	Alive	On	AIX	IBM	5.3	5.3	ppc	7038	6M2	10197AA	

Figure 3-19 CSM monitoring WebSM

There are numerous other predefined conditions in addition to the ones provided by standalone RMC; for example conditions for monitoring of the AIX error log or Linux /var/log/messages file.

In our test scenario we use the Error log condition to monitor new Error log entries. The procedure is similar to condition configuration in standalone RMC monitoring, thus we highlight the main differences:

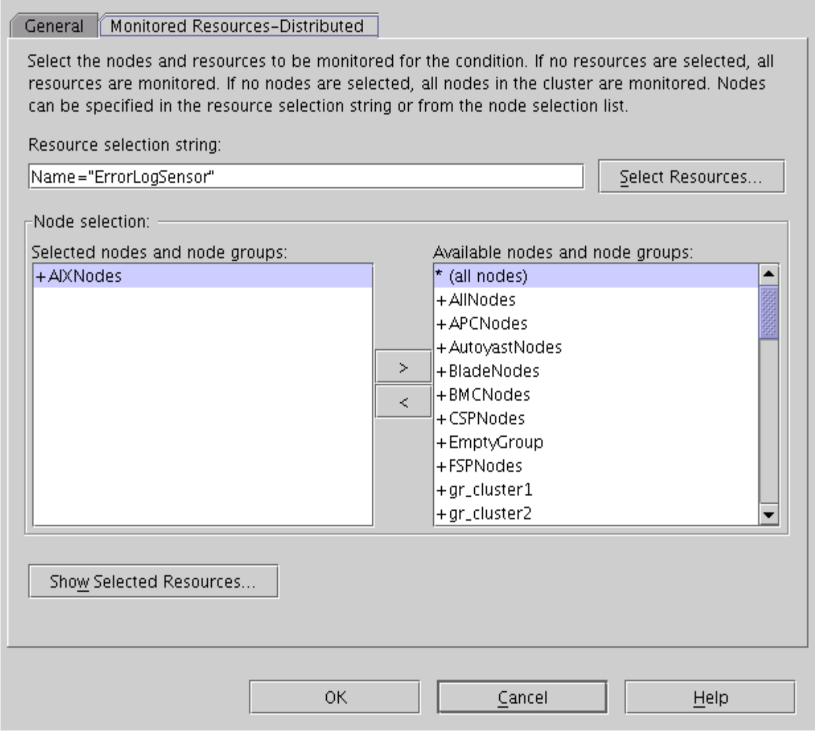
1. Open WebSM and select **CSM Cluster** → **Distributed Monitoring** → **Conditions** in the **Navigation Area**.
2. Open the **AnyNodeAnyLoggedError** condition and adjust it as shown in Figure 3-20 on page 95. Observe that the Management Scope is set to Distributed Management Cluster.


Condition AnyNodeAnyLoggedError Properties @ csm_ms01
General
Monitored Resources-Distributed
Details...
Use defaults
Responses to Condition...
OK
Cancel
Help

Name: AnyNodeAnyLoggedError
Management scope: Distributed Management Cluster
Monitored: Yes
Resource class: Sensor
Monitored property: String
Event expression: String != ""
Event description: An event will be generated when an error is logged to either the AIX Error Log or the Linux Syslog of a node in the cluster.
Rearm expression:
Rearm description:
Severity: Warning

Figure 3-20 Error log condition

3. Open the **Monitored Resources-Distributed** tab and adjust it as shown in Figure 3-21 on page 96. Variable will be changed based on your own requirements.



Select the nodes and resources to be monitored for the condition. If no resources are selected, all resources are monitored. If no nodes are selected, all nodes in the cluster are monitored. Nodes can be specified in the resource selection string or from the node selection list.

Resource selection string: Name="ErrorLogSensor"

Node selection:

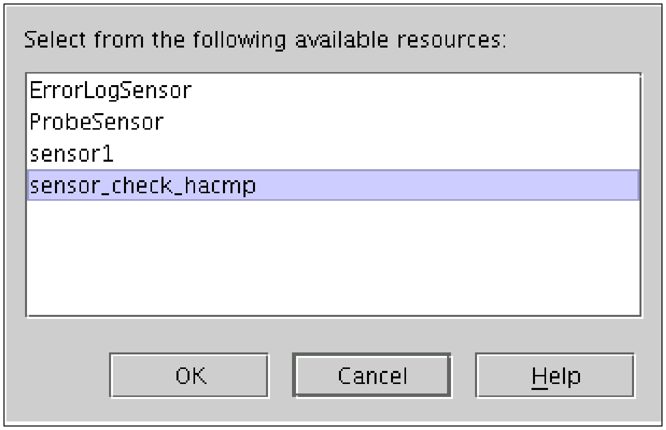
Selected nodes and node groups:
+AIXNodes

Available nodes and node groups:
* (all nodes)
+AllNodes
+APCNodes
+AutoyastNodes
+BladeNodes
+BMCNodes
+CSPNodes
+EmptyGroup
+FSPNodes
+gr_cluster1
+gr_cluster2

Figure 3-21 CSM Error log monitor properties

4. Define your responses and activate the new monitor.

In addition to the predefined or new sensors, we can also reuse our custom sensors as defined in our RSCT peer domain scenario. We simply copy an existing sensor monitor and, in the Monitored Resources-Distributed tab, click the **Select Resources** button, then select our HACMP sensor as shown in Figure 3-22 on page 97.


Select from the following available resources:

ErrorLogSensor
ProbeSensor
sensor1
sensor_check_hacmp

Figure 3-22 Custom defined sensor in CSM GUI

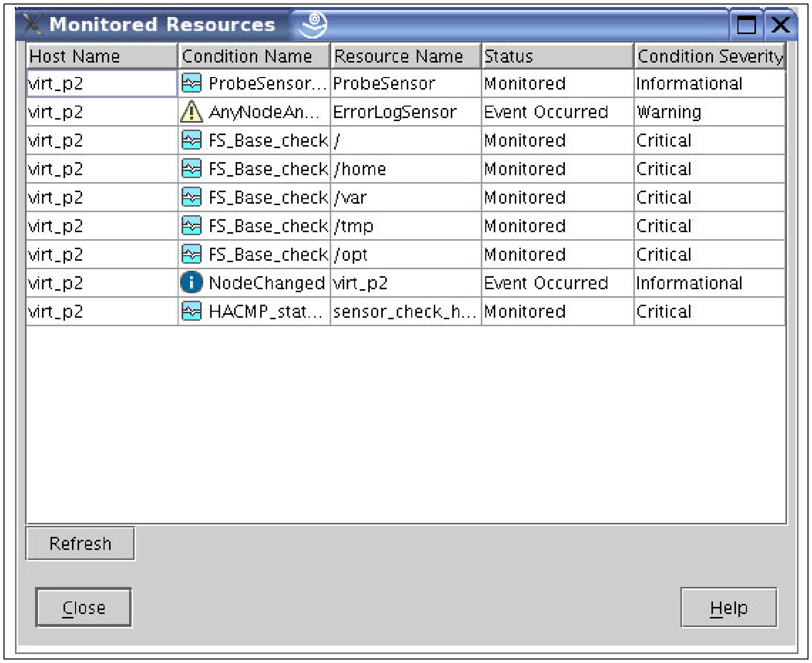
We still have the possibility to define our conditions and responses via command line. In Example 3-24 we illustrate how we defined a file system condition to monitor free space in /usr filesystem. Note the **CT_MANAGEMENT_SCOPE=3** variable and the **`-mm`** parameter of the **`mkcondition`** command - both denoting management domain scope.

Example 3-24 Setting conditions and responses via CLI

```
#!/bin/sh

  export CT_MANAGEMENT_SCOPE=3

rmcondition -f "Base FS check 2"
mkcondition -r "IBM.FileSystem" \
-mm \
-n AIXNodes \
-e "PercentTotUsed > 99" \
-E "PercentTotUsed < 98" \
-d "Low File system space" \
-D "Low File system space problem resolved" \
-s 'Name == "/usr"' -S "c" "Base FS check 2"

 lscondition "Base FS check 2"

rmresponse -q "Escalate to Watch Centrum"
mkresponse -n "Escalate WC" -d 1-7 -t 0000-2400 -s
/usr/local/bin/tell_to_watch_centrum.sh -e b -r 1 "Escalate to Watch
Centrum"
```

```
stopcondresp -q "Base FS check 2"
mkcondresp "Base FS check 2" "E-mail root anytime" "Escalate to Watch
Centrum"
startcondresp "Base FS check 2"
```

When you complete the set of monitored conditions for your nodes, you can view them together with their status. Right-click the node and select **Monitored Resources** as shown in Figure 3-23.



Host Name	Condition Name	Resource Name	Status	Condition Severity
virt_p2	ProbeSensor...	ProbeSensor	Monitored	Informational
virt_p2	⚠ AnyNodeAn...	ErrorLogSensor	Event Occurred	Warning
virt_p2	FS_Base_check	/	Monitored	Critical
virt_p2	FS_Base_check	/home	Monitored	Critical
virt_p2	FS_Base_check	/var	Monitored	Critical
virt_p2	FS_Base_check	/tmp	Monitored	Critical
virt_p2	FS_Base_check	/opt	Monitored	Critical
virt_p2	ℹ NodeChanged	virt_p2	Event Occurred	Informational
virt_p2	HACMP_stat...	sensor_check_h...	Monitored	Critical

Figure 3-23 Monitored resources on a node

3.3.5 Monitoring SFP hardware events via HMC

For this tasks we run remote SSH commands to the HMCs and query the HMC for Hardware events delivered through Service Focal Point (SFP), using the `lssvcevents` command on HMC.

Note: The lssvcevents command syntax is different between System p4 and System p5 HMC versions. Moreover, the command syntax may be updated to reflect SFP and System p changes. This is why we present two different monitors for HMC Version 3.7 (System p4) and HMC Version 5.2.1 (System p5).

1. Establish the remote SSH execution between the CSM MS and the HMC to work without prompting for password or passphrase.
2. Create the script on the CSM to run the remote shell commands. The list of HMCs to check is in the body of the script, see sample in Example A-2 on page 201.
3. Create the RMC monitors as shown in Example 3-25 and Example 3-26 on page 100.

Example 3-25 Script for RMC monitoring of the HMCs Version 3.7 (System p4)

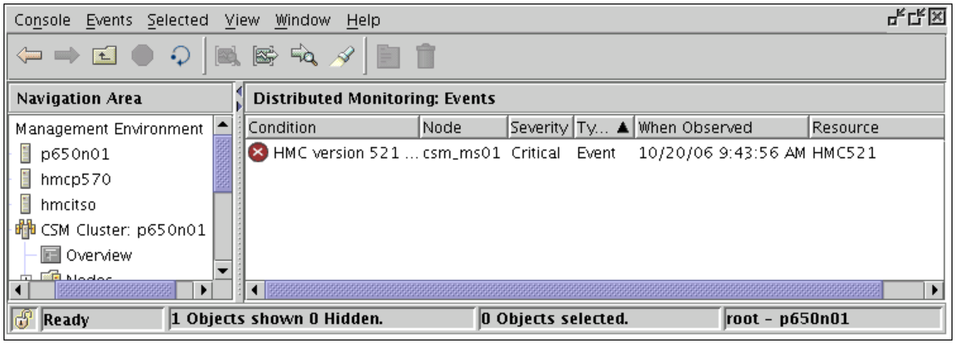
```
#!/bin/sh


CONDITION="HMC version 370 SFP"

export CT_MANAGEMENT_SCOPE=

/usr/sbin/rsct/bin/rmsensor HMC370
/usr/sbin/rsct/bin/mksensor -i 600 -e 0 HMC370
/usr/local/bin/hmc4_monitor.sh


rmcondition -f "$CONDITION"
mkcondition -r "IBM.Sensor" \
-ml \
-e "String != \"\"" \
-d "An event will be generated when SFP on HMC for P4 machines has
error" \
-s 'Name == "HMC370"' -S "c" "$CONDITION"

lscondition "$CONDITION"
stopcondresp -q "$CONDITION"

mkcondresp "$CONDITION" "E-mail root anytime" "Escalate to Watch
Centrum"
startcondresp "$CONDITION"
```

For more details about how to monitor SFP events see “Monitoring SFP events” in “Diagnostics” in *IBM Cluster Systems Management for AIX 5L and Linux Administration Guide Version 1, Release 6*, SA23-1342.

Example 3-26 Script for RMC monitoring of the HMCs Version 5.2.1 (System p5)

```
#!/bin/sh


CONDITION="HMC version 521 SFP"

export CT_MANAGEMENT_SCOPE=

/usr/sbin/rsct/bin/rmsensor HMC521
/usr/sbin/rsct/bin/mksensor -i 600 -e 0 HMC521
/usr/local/bin/hmc5_monitor.sh


rmcondition -f "$CONDITION"
mkcondition -r "IBM.Sensor" \
-ml \
-e "String != \"\"" \
-d "An event will be generated when SFP on HMC for P5 machines has 
error" \
-s 'Name == "HMC521"' -S "c" "$CONDITION"

lscondition "$CONDITION"
stopcondresp -q "$CONDITION"

mkcondresp "$CONDITION" "E-mail root anytime" "Escalate to Watch 
Centrum"
startcondresp "$CONDITION"
```

4. If the shell script returns any output, this will be evaluated as an error, as seen in Figure 3-24 on page 101.



Condition	Node	Severity	Ty... ▲	When Observed	Resource
HMC version 521 ...	csm_ms01	Critical	Event	10/20/06 9:43:56 AM	HMC521

Ready | 1 Objects shown 0 Hidden. | 0 Objects selected. | root - p650n01

Figure 3-24 Service Focal Point Event detected by RMC monitor on CSM server

Note: The time interval defined for the script to run defined by -i parameter of the **`mksensor`** command should be the same as time to backward query events for the **`lssvcevents`** command in the scripts.

3.3.6 Monitoring of the VIOS error log

Note: The VIO server is considered a closed box, and thus it is not recommended to run any user applications. However, some of the errors that appear on the VIO LPAR are not propagated to the client partitions. For example, a failed MPIO path or EtherChannel will appear in the VIO server error log, but the client partitions will not be aware of these events.

The procedure shown in this paragraph is not officially supported and requires also the installation and configuration of the secure shell. For this, you need to login as “root” on the VIO server (oem_setup_env), install and configure ssh daemon (/etc/ssh/sshd_config) as shown in Example 3-27.

Example 3-27 SSH daemon configuration to allow remote SSH command execution

```
PermitUserEnvironment yes
```

Assuming that the VIO server is already defined to the CSM MS and this can run remote commands via ssh without prompting for the password for the padmin user, you have to do the following steps:

1. Create the monitoring script on the CSM server as shown in Example 3-28.

Example 3-28 CSM script to monitor VIOS

```
#!/bin/ksh
```

```
VIOs="virt_vios1"

  for vio in $VIOs
    do
     ssh padmin@${vio} VIO_errorlog.pl
      if [ $? -ne 0 ]
        then
   echo "The ssh command to $hmc did not complete \
successfully, check the command line options in $0 and $vio
reachability"
      fi
  done
exit 0
```

2. On the VIO LPAR copy the IO_errorlog.pl script into the /usr/ios/utils/ directory. The script is shown in A.5, “VIO Version 1.3 monitor script” on page 204.
3. To define the RMC monitor on the CSM management server, run script shown in Example 3-29.

Example 3-29 VIO RMC monitor definition on CSM

```
#!/bin/sh


CONDITION="VIO 31 ERRORLOG"

export CT_MANAGEMENT_SCOPE=

/usr/sbin/rsct/bin/rmsensor VIO31
/usr/sbin/rsct/bin/mksensor -i 300 -e 0 VIO31
/usr/local/bin/vio_monitor.sh


rmcondition -f "$CONDITION"
mkcondition -r "IBM.Sensor" \
-m1 \
-e "String != \"\"" \
-d "An event will be generated when VIO has new entry in error log" \
-s 'Name == "VIO31"' -S "c" "$CONDITION"

lscondition "$CONDITION"
stopcondresp -q "$CONDITION"
```

```
mkcondresp "$CONDITION" "E-mail root anytime" "Escalate to Watch
Centrum"
startcondresp "$CONDITION"
```

Note: We did not test this procedure to monitor an IVM LPAR, but the procedure should be the same as for the VIO server.

3.3.7 CSM Applications support

The chapter "Using CSM to administer cluster applications" of the manual *IBM Cluster Systems Management for AIX 5L and Linux Administration Guide Version 1, Release 6*, SA23-1342, includes examples of how to use CSM features to support multiple nodes running the same application. Some of these features are:

- Creating custom dynamic node groups
- Creating custom post installation scripts to install and configure your application
- Configuring and monitoring specific start and stop procedures for applications

Using these features can benefit not only HPC but also commercial applications, such as IHS or Lotus Domino servers.

3.3.8 Migration to CSM Version 1.6

Migration to CSM Version 1.6 from previous versions is simple and straightforward procedure and is well documented in *IBM Cluster Systems Management for AIX 5L and Linux 1.6 Planning and Installation Guide*, SA23-1344-02. An upgrade consists of:

- Updating CSM code on the CSM MS.
- Updating CSM code on the nodes.
- Running **updatenode** commands on the nodes.

One prerequisite of CSM Version 1.6 is RSCT Version 2.4.6. You will need to upgrade RSCT on your MS and nodes either prior or during the CSM upgrade.

Note: After updating the RSCT filesets you will be reminded to recycle HACMP software if you use it.

3.3.9 VIOS LPAR CSM Integration

CSM provides support for VIOS (both IVM and non-IVM) partitions. This section relates to VIOS (non-IVM) integration. In our test environment we have two VIOS (non-IVM) LPARs with the IP addresses shown in Table 3-2.

Table 3-2 VIOS LPARs

Hostname	IP Address
virt_vios1	192.168.100.71
virt_vios2	192.168.100.72

We created a configuration file to define both VIO servers to CSM. Example 3-30 shows how we used the `default` attribute to group shared characteristics. Note the `DeviceType` of `IOServer`. The hardware control point (`hmcp570`) has been previously defined and is known to CSM.

Example 3-30 VIOS device definition file

```
default:
        PowerMethod=hmc
        HWControlPoint=hmcp570
        ConsoleMethod=hmc
        ConsoleServerName=hmcp570
        PhysicalLocation="ITSO Data Room Poughkeepsie"
        DeviceType=IOServer
        RemoteShell=/usr/bin/ssh
        RemoteCopyCmd=/usr/bin/scp
        RemoteShellUser=padmin

virt_vios1:
        UserComment="Virtual IO Server 1 on p570"
        HWControlDeviceId="VIO LPAR1"

virt_vios2:
        UserComment="Virtual IO Server 2 on p570"
        HWControlDeviceId="VIO LPAR2"
```

Using this definition, we defined the two devices into our CSM cluster. After the successful definition (shown in Example 3-31), they can be listed or queried using the **`lshhdev`** command. Note that the device’s Mode is `PreManaged`.

Example 3-31 Defining VIOS devices to CSM

```
root@csm_ms01:# definehwdev -f ./50-VIOS_dev.cfg
Defining CSM Devices:
Defining Device "virt_vios1"
Defining Device "virt_vios2"

root@csm_ms01:# lshwdev
hmcitso
hmcp570
virt_vios1
virt_vios2

root@csm_ms01:# lshwdev -l virt_vios1
 Name = virt_vios1
 AllowManageRequest = 0 (no)
 ConsoleMethod = hmc
 ConsolePortNum =
 ConsoleServerName = hmcp570
 DeviceType = IOServer
 FWMgtProcBootROM =
 FWMgtProcMainApp =
 FWMgtProcRemCtrl =
 HWControlDeviceId = VIO LPAR1
 HWControlPoint = hmcp570
 HWModel =
 HWSerialNum =
 HWType =
 Macaddr =
 Mode = PreManaged
 PhysicalLocation = ITSO Data Room Poughkeepsie
 PowerMethod = hmc
 PowerStatus = 1 (on)
 Properties =
 RemoteCopyCmd = /usr/bin/scp
 RemoteShell = /usr/bin/ssh
 RemoteShellUser = padmin
 Status = 1 (alive)
 StatusMethod = ping
 UserComment = Virtual IO Server 1 on p570
```

The final integration step is to run the appropriate **`updatehwdev`** command. This will perform the required tasks to promote the device to `Managed` Mode. Example 3-32 shows the output for the virt_vio1 LPAR as an example.

Example 3-32 Running updatehwdev for VIOS LPAR

```
root@csm_ms01:# updatehwdev -v -k -I -d virt_vios1
Output log for updatehwdev is being written to
/var/log/csm/updatehwdev.log.
Running Command:  lsrsrc-api -D ':|:' -i -s
IBM.DeviceHwCtrl::::Name::RemoteShell::RemoteShellUser 2>&1
Running Command:  /opt/csm/csmbin/deviceremoteshell.expect -sd padmin
/usr/bin/ssh virt_vios1
Devices already configured for remote shell: /usr/bin/ssh, userid:
padmin and devices: virt_vios1.
Running Command:  /usr/bin/chrsrc-api -i -s IBM.DeviceHwCtrl::"Name IN
('virt_vios1')"::AllowManageRequest::1 2>&1
Running Command:  /opt/csm/csmbin/getSourceIP2Target virt_vios1 2>&1
Running Command:  /usr/bin/ssh padmin\@virt_vios1 runlpcmd mgmtsvr -k
-v -n virt_vios1 192.168.100.51 2>&1
rksh: runlpcmd:  not found.
Running Command:  echo "/opt/csm/bin/mgmtsvr -k -v -n virt_vios1
192.168.100.51" | /usr/bin/ssh padmin\@virt_vios1 ioscli oem_setup_env
2>&1
Running Command:  /usr/bin/lsrsrc-api -i -D ':|:' -s
IBM.ManagementServer::"ManagerType='CSM'"::Hostname::LocalHostname::HAS
tate 2>&1
Running Command:  /usr/bin/mkrsrc-api
IBM.ManagementServer::Hostname::"192.168.100.51"::ManagerType::"CSM"::L
ocalHostNamesList::'{"virt_vios1"}'::MgmtSvrHostNamesList::'{"192.168.1
00.51"}'::LocalHostname::"virt_vios1" 2>&1
Running Command:  /usr/bin/refresh -s ctrmc  2>&1
Running Command:  /usr/bin/chrsrc-api -i -s IBM.DeviceHwCtrl::"Name IN
('virt_vios1')"::AllowManageRequest::0 2>&1
```

If the VIOS LPAR was incorrectly configured, the **updatehwdev** command may fail, if our. For example we needed to ensure hostname to IP addresses is correctly resolved.

Note: Such managed-integration is only supported with CSM 1.6.x or later and VIOS Version 1.3.0 or later. For more information refer to *IBM Cluster Systems Management for AIX 5L and Linux 1.6 Planning and Installation Guide*, SA23-1344-02.

3.3.10 IVM CSM integration

CSM Version 1.6.0 also brings support (although yet limited) for Integrated Virtualization Manager (IVM). Appendix I in *IBM Cluster Systems Management* *for AIX 5L and Linux 1.6 Planning and Installation Guide*, SA23-1344-02, describes the process for installing an IVM and defining it into a CSM environment. However, not all steps described in the document are required if you plan to integrate an existing IVM.

We start with the systemid commands used to communicate with the FSP, shown in Example 3-33.

Example 3-33 Managing FSP authentication

```
systemid -f FSP_IP HMC          (default password is abc123)
systemid -f -s FSP_IP admin     (default password is admin)
systemid -f -s FSP_IP general   (default password is general)
```

The first command authenticates with the FSP. The subsequent two commands authenticate *and* set the admin and general passwords.

For an existing IVM you may not want to reset the FSP passwords, but instead to simply store and use the existing values. In this case, the **`-s`** parameter would not be required on the second or third **`systemid`** commands. Setting the passwords is only necessary if the FSP has the factory default settings.

We found the output of the **`lshwconn`** command helpful during this step of integration. Example 3-34 illustrates the difference in output during and after changing the FSP password.

Example 3-34 lshwconn example

```
# lshwconn -n virt_vios

Node Hostname  HWControlPoint Connection Status          PowerMethod
---------------------------------------------------------------------
virt_vios      192.168.100.85 CEC_PASSWORD_CHANGE_PENDING fsp

# lshwconn -n virt_vios

Node Hostname  HWControlPoint Connection Status          PowerMethod
---------------------------------------------------------------------
virt_vios      192.168.100.85 LINE_UP                     fsp
```

In Example 3-34, the `LINE_UP` status shows successful authentication with the FSP.

If you plan to integrate an existing IVM (and hosted LPARs) into a CSM Cluster, we strongly suggest reviewing the steps in the previously mentioned Appendix I (“Appendix I” in *IBM Cluster Systems Management for AIX 5L and Linux 1.6*

Planning and Installation Guide, SA23-1344-02). If it is unclear which steps you should take for your environment, contact your IBM support representative.

> **Note:** CSM Version 1.6.0 brings support for IBM BladeCenter JS21 Blades. The JS21 blades have the ability to host an IVM and resources can be carved into LPARs. Each blade can have its own IVM instance. However at the time of writing CSM does not support integration of IVM-managed JS21 blades.

3.3.11 CSM Integration with the existing NIM server

In our scenario we already have an existing NIM server and want to use it both independently from CSM and as an CSM Install Server. This task was easier than we expected.

Integration consists of a few steps documented here:

1. Create the `/csmserver` file system (this is optional, but is the default used by CSM. The directory can be changed via the **installserver** attribute) on the NIM node, which in our scenario this is *node6*. The file system should be the same size as the same filesystem on the MS.

   For the nodes to be installed or restored, we changed the **installserver** attribute to the *node6* using the **`chnode`** command. You can see the attribute as part of the **`lsnode -l virt_p4`** command output shown in Example 3-35.

Example 3-35 CSM node definition attributes

```
root@csm_ms01:/cfmroot/etc# lsnode -l virt_p4
 Hostname = virt_p4
 AdapterStanzaFile = /home/vanous/w/41-sec_adapters.cfg
...
ConsoleServerName = hmcp570
 ConsoleServerNumber =
HWControlNodeId = AIX LPAR3
 HWControlPoint = hmcp570
 HWModel = 570
...
InstallAdapterDuplex = auto
 InstallAdapterGateway = 192.168.100.76
 InstallAdapterHostname = virt_p4
 InstallAdapterMacaddr = 0002556A51EF
 InstallAdapterName = en0
 InstallAdapterNetmask = 255.255.255.0
 InstallAdapterSpeed = auto
 InstallAdapterType = ent
 InstallCSMVersion =
 InstallDisk =
 InstallDiskType =
 InstallDistributionName =
 InstallDistributionVersion = 5.3.0
 InstallKernelVersion =
 InstallMethod =
 InstallOSName = AIX
 InstallPkgArchitecture =
 InstallServer = node6:/csmserver
...
```

2. Run the **updateisvr -a** command on the CSM MS. This will copy the required files to /csminstall filesystem on the NIM server. Note that the post-installation or pre-reboot scripts will also be stored in this directory, on the NIM server
3. Update the node installation information. You have the option to use **getadapters** command to retrieve adapter information, as described in *IBM Cluster Systems Management for AIX 5L and Linux 1.6 Planning and Installation Guide*, SA23-1344-02, or update them from the command line using **chnode** command. Note that one of the adapters (if more than one) must have attribute **machine_type** set to **install**. Example 3-36 shows the output of the **getadapters** command with edited attribute **machine_type**.

Example 3-36 Secondary adapter stanzas

```
###CSM_ADAPTERS_STANZA_FILE###--do not remove this line
#---Stanza Summary----------------------
#   Date: Thu Oct 19 11:14:27 EDT 2006
#   Stanzas Added: 2
#---End Of Summary----------------------

virt_p4:
     MAC_address=0002556A51EF
     adapter_duplex=full
     adapter_speed=100
     adapter_type=ent
     cable_type=N/A
     install_gateway=192.168.100.60
     interface_name=en0
     location=U7879.001.DQDKZNP-P1-C3-T1
     machine_type=install
     netaddr=192.168.100.82
     interface_type=en
     subnet_mask=255.255.255.0
```

```
virt_p4:
     MAC_address=0002556A304A
     adapter_duplex=full
     adapter_speed=100
     adapter_type=ent
     cable_type=N/A
     install_gateway=192.168.100.60
     interface_name=en1
     location=U7879.001.DQDKZNP-P1-C1-T1
     machine_type=secondary
     netaddr=172.16.51.82
     interface_type=en
     subnet_mask=255.255.255.0
```

4. Define the NIM resource group **basic_res_grp** which is the collection of the resources to be installed on the nodes. This is an optional step, but simplifies NIM commands. This group consists of at least an lpp_source and a spot NIM resource. An optional mksysb and other optional NIM resources can be added to the resource group. Depending on how your NIM server is configured, you may need to create more than one resource group.
5. The **csmsetupnim** command is used to prepare the **basic_res_grp** resources for AIX nodes which will be installed. It also prepares customization scripts which are run after operating system installation and first reboot as resources on the NIM server. If you do not run these scripts, you cannot use automatic node definition to CSM. You would have to use the **updatenode command** manually after the node installation to configure them as CSM nodes.
6. After **csmsetupnim** command you next run **nim -o bosinst command on the NIM server** (either directly or using **dsh** command from MS), as documented in *IBM Cluster Systems Management for AIX 5L and Linux 1.6 Planning and Installation Guide*, SA23-1344-02.
7. We are now ready to initiate node installation. The **netboot command** performs the install by automating the hardware and SMS phases of the typical LPAR install process. This is summarized in Example 3-37.

Example 3-37 Network boot (CSM initiated) and monitoring the operating system installation progress

```
root@csm_ms01:/var/log/csm/netboot# netboot -n virt_p4
...
[root@csm_ms01:/ dsh -n node6 “lsnim -l virt_p4”
virt_p4:
   class          = machines
   type           = standalone
   connect        = shell
   platform       = chrp
   netboot_kernel = mp
   if1            = itso virt_p4 0
   cable_type1    = tp
   Cstate         = post install processing is being performed
   prev_state     = customization is being performed
   Mstate         = in the process of booting
   info           = BOS install 89% complete : Network installation
manager customization.
   boot           = boot
   bosinst_data   = BOSINST_53NOPROPMT
   lpp_source     = LPP_AIX53_TL5
   mksysb         = mksysb_virt_p4
   nim_script     = nim_script
   spot           = SPOT_53TL5
   cpuid          = 00CC5D5C4C00
   control        = master
```

Note: If you are having problems with installation and you have to cancel the current installation process, it is worth to prepare the NIM resources again. The steps for this are shown in Example 3-38.

Example 3-38   Steps to clear and restart a NIM installation on a given node

```
rpower -n virt_p4  off   #-- Power off the node
dsh -n node6 nim -o reset -F virt_p4 #-- Reset the NIM state on NIM
csmsetupnim -n virt_p4   #-- Prepare CSM resources
dsh -n node6 “nim -o bos_inst -a source=rte -a boot_client=no -a \
group=basic_res_grp virt_p4” #-- Prepare the NIM resources on NIM
rconsole -r -t virt_p4       #-- Open the console in read-only mode
netboot -n virt_p4           #-- Network boot of the node
```

3.3.12 Useful CSM commands

There are several useful command which can provide a quicker way to achieve results. It is also true that most of the tasks can be manually and directly executed on each node; however, using distributed commands from a single point can be much more efficient. As IT personnel may not be familiar with the power or functionality offered by distributed commands, the use of CSM alleviates administrative tasks and promotes efficient system management.

One particular example is AIX installation. CSM provides the **netboot command** which saves manual interaction with the HMC and target LPAR. This removes the need to manually power on the LPAR (from the HMC), and interact with the LPAR's SMS menu to initialize communication with the NIM server.

CSM provides centralized terminal access to cluster nodes. Both read and write consoles are supported. You can open multiple (tiled) consoles with read-only access to monitor the installation progress, as shown in Example 3-39.

Example 3-39 Running rconsole for read-only console for a node group

```
root@csm_ms01:/cfmroot# export RCONSOLE_FONT=fixed
root@csm_ms01:/cfmroot# rconsole -N gr_virt_aix -O 10
```

Note: For those preferring terminal sessions in text mode, you can use the **rconsole -t** option.

After the installation is finished you can close all consoles and check if the nodes are active by using the **dping** command as shown in Example 3-40.

Example 3-40 dping command usage

```
rconsole -x -A
root@csm_ms01:/# dping -N gr_p570_aix
virt_p5: ping (alive)
virt_p4: ping (alive)
virt_p6: ping (alive)
```

Check the manual *IBM Cluster Systems Management for AIX 5L and Linux V1.6 Command and Technical Reference*, SA23-1345, for command usage and additional examples.

3.3.13 Secondary adapters configuration

If you have multiple nodes with more then one adapter to be configured, you can use the CSM Secondary Adapters function. In fact, you define the adapters first in CSM and then by running **updatenode command** with the `-c` flag, the adapters are configured automatically on the nodes. This avoids much manual configuration. Moreover, if you are using CSM with NIM, the adapters are configured after the operating system installation or restore, which saves time and promotes consistency.

If you are unfamiliar with the syntax, one way to configure secondary adapters is by manually configure adapters on one node, then extract the definition via the **getadapters** command (with -z flag). Next, you create the definitions for all nodes and commit the new definitions into the CSM database with the **getadapters** command (with **-W** flag). For more information refer to *IBM Cluster*

Systems Management for AIX 5L and Linux 1.6 Planning and Installation Guide, SA23-1344-02.

3.3.14 Hardware and software inventory and configuration

There are a number of ways to use CSM to provide inventory data. We illustrate some of the approaches we found useful.

The **rfwscan** CSM command can be used to collect node hardware data as shown in Example 3-41.

Example 3-41 Hardware inventory data collected by rfwscan command

```
root@csm_ms01:/home/vanous/w# /opt/csm/bin/rfwscan -n virt_p6
Nodename = virt_p6
Managed System Release Code Level = 01SF240
Active Service Code Level = 261
Installed Service Code Level = 261
Accepted Service Code Level = 219
Power Subsystem MTMS =
Power Subsystem Release Code Level =
Active Service Code Level =
Installed Service Code Level =
Accepted Service Code Level =
```

If you write your own inventory scripts, these can easily be distributed by CFM and executed remotely on your cluster nodes. To improve the readability of the command output from multiple nodes, you can evaluate the use of **dshbak** as shown in Example 3-42. The script which produced this output is listed in A.7, “Inventory scripts” on page 208.

Example 3-42 Custom inventory script output (dsh and dshbak consolidated)

```
root@csm_ms01:/# dsh -N gr_p570_aix /usr/local/bin/get_fc_wwn.sh |
dshbak -c
HOSTS
--------------------------------------------------------------------
virt_p4
--------------------------------------------------------------------
  fcs0             U7879.001.DQDKZNV-P1-C4-T1  FC Adapter
        Network Address.............10000000C93301A1
  fcs1             U7879.001.DQDKZNV-P1-C2-T1  FC Adapter
        Network Address.............10000000C940778B

HOSTS
--------------------------------------------------------------------
```

```
virt_p5
--------------------------------------------------------------------
  fcs0             U7879.001.DQDKZNV-P1-C3-T1  FC Adapter
        Network Address.............10000000C93301D4
  fcs1             U7879.001.DQDKZNV-P1-C5-T1  FC Adapter
        Network Address.............10000000C931AF4B
```

For software inventory you can use the NIM comparison reports as shown in Figure 3-14 on page 84. Depending on what software you are want to query, there is one disadvantage: you have to have the software for which you are doing the comparison defined as a resource on the NIM server.

This means that you have to store the software on NIM server. If you have limited storage space on your NIM servers, one way around this would be to delete the software packages, from the NIM server after resource definition - leaving just the .toc file. NIM software comparison will continue to work, although you will not be able to use the NIM resource for install purposes.

Custom configuration inventory

In some cases the environment mandates that you track configuration changes over time:

- Changes at the operating system level, such as /etc/hosts table or user configuration.
- Changes at the hardware or LPAR level, such as number of CPU, CPU capacity entitlement and parameters of the shared CPU pool.
- Changes at the application layer, such as HACMP configuration.

Some enterprise tools exist (IBM Tivoli Change and Configuration Management Database (CCMDB) and others), however, if you do not want to spend additional money on software, you can use the following approach:

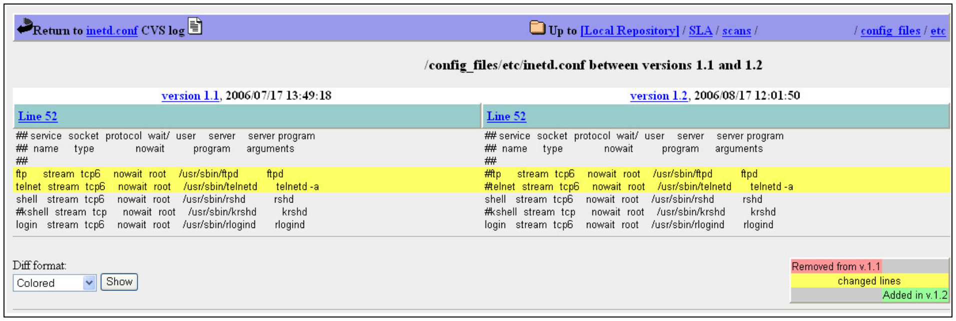
1. Run commands on your systems (for example using DCEM) to collect the information as configuration files or output of your script to a single tar file on your system.
2. Collect the individual tar files on the reporting server where Concurrent Versions System (CVS) server and Apache server is installed (both GPL). For CVS, see the Web page:

   http://ximbiot.com/cvs/

   For Apache, see:

   http://www.apache.org/
3. Using a script un-tar and import the output files into the CVS.

4. Then we can use the CVS Web interface to compare the status or the difference in configuration, as shown in Figure 3-25.



Return to inetd.conf CVS log

Up to [Local Repository] / SLA / scans / / config_files / etc

/config_files/etc/inetd.conf between versions 1.1 and 1.2

version 1.1, 2006/07/17 13:49:18	version 1.2, 2006/08/17 12:01:50
Line 52	Line 52

```
## service  socket  protocol  wait/  user   server   server program
## name     type             nowait        program   arguments
##
ftp     stream  tcp6   nowait  root   /usr/sbin/ftpd       ftpd
telnet  stream  tcp6   nowait  root   /usr/sbin/telnetd    telnetd -a
shell   stream  tcp6   nowait  root   /usr/sbin/rshd       rshd
#kshell  stream  tcp   nowait  root   /usr/sbin/krshd      krshd
login   stream  tcp6   nowait  root   /usr/sbin/rlogind    rlogind
```

```
## service  socket  protocol  wait/  user   server   server program
## name     type             nowait        program   arguments
##
#ftp    stream  tcp6   nowait  root   /usr/sbin/ftpd       ftpd
#telnet stream  tcp6   nowait  root   /usr/sbin/telnetd    telnetd -a
shell   stream  tcp6   nowait  root   /usr/sbin/rshd       rshd
#kshell  stream  tcp   nowait  root   /usr/sbin/krshd      krshd
login   stream  tcp6   nowait  root   /usr/sbin/rlogind    rlogind
```

Diff format: Colored [Show]

Figure 3-25 Change of the configuration if stored into the CVS

In A.7, “Inventory scripts” on page 208 we present sample scripts that can be used to retrieve the data into the CVS.

3.3.15 Historic performance data collection

In some cases it is important to store historical performance data. There are several ways to achieve this, including using Tivoli Enterprise Data Warehouse or similar multi-platform enterprise tools. If you need a simple solution for AIX or Linux you can use nmon tool with RRDTool database. RRDTool information can be found at:

```
http://oss.oetiker.ch/rrdtool/
```

What follows is an overview of the steps to collect the data:

1. Install and run nmon from a cron job; this will collect performance data locally on the node.
2. Collect the data to a central point; this can be easily achieved using dcp commands via DCEM.
3. Import the data into RRDTool database.
4. View the data in a browser as illustrated in Figure 3-26 on page 116. White spaces are periods where no data was collected or their data was lost; despite of that you can see from the example where at the week 25 there was a hardware migration of the SAP system from an IBM System p4 machine to IBM System p5. The new hardware also was configured with additional Fibre-Channel adapters to eliminate a high I/O load bottleneck. The nmon collected data shows that wait IO time dramatically decreased after this migration.

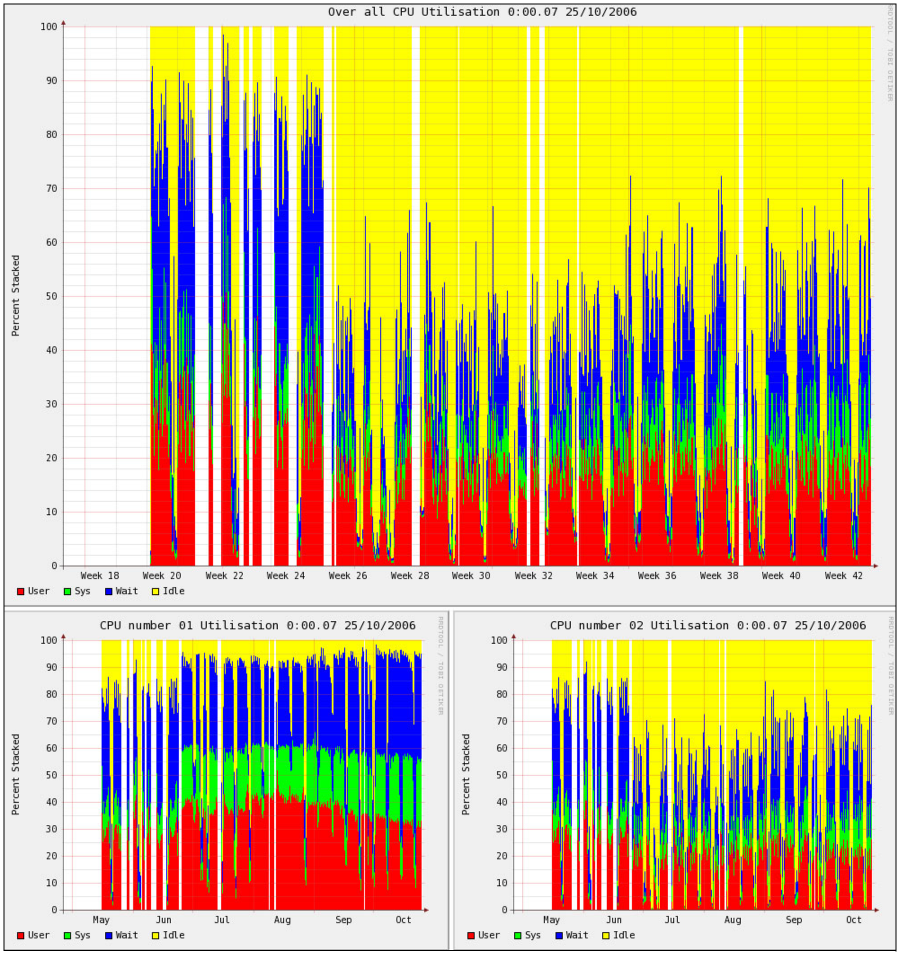


Figure 3-26 Example of the nmon historical data output

This solution has some limitations, and nmon is not officially supported by IBM. However, nmon collects a wealth of information, such as hardware configuration and operating system configuration. You can find our sample scripts for data creation in A.6, “Nmon performance collection scripts” on page 206.

> **Note:** While the nmon analyzer is provided “as is,” it does provide many features in a single tool. For more information and to download nmon, refer to the following Web page:
>
> `http://www-128.ibm.com/developerworks/aix/library/au-analyze_aix/`

Here are other alternative solutions for analyzing historic data:

- Use Workload Manager (WLM) in passive mode, as documented in *IBM System p Advanced POWER Virtualization Best Practices*, REDP-4194.
- Ganglia tool, which looks similar to nmon and moreover is able to consider performance of the entire machine; for more information see:

  http://ganglia.info/
- Performance Graph Viewer, which has similar features such as Ganglia. For more information, refer to the section about Performance Graph Viewer in the following AIX 5L Wiki:

  http://www-941.ibm.com/collaboration/wiki/display/WikiPtype/Performance+Graph+Viewer

3.3.16 Using CFM

There are plenty of situations when file distribution is very efficient; combining CFM with node groups is a natural progression of that efficiency. Its use also promotes standardization. Once you become familiar with the one-to-many relationship between the CSM MS and the managed nodes, many possibilities present themselves. In our test scenarios we used CFM for the following tasks:

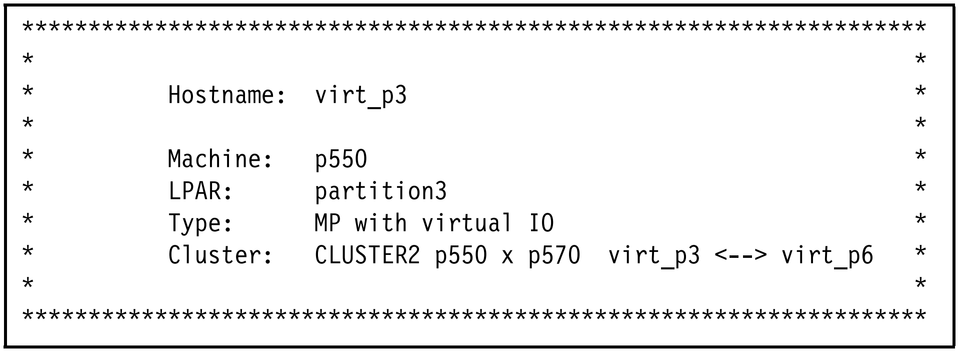
- Ensuring identical /etc/environment and root user profile files (~/.profile and ~/.env) to provide the same working environment on all cluster nodes.
- Configuring /etc/inetd.conf across the cluster for example to disable telnet, ftp, rsh and other undesirable services on decided nodes.
- Distributing user definitions, such as /etc/passwd, /etc/security/user or /etc/security/limits. Using this approach can provide distributed UserIDs, thus avoiding having to manually create IDs on new nodes or setting up an external user management system.
- Defining the HACMP specific updates to /etc/hosts or /etc/services files for HACMP cluster nodes. Therefore, when takeover occurs, we can be sure that the application uses the same IP hostname (label) and IP service name resolution.
- Distribution of subsystem configuration files, such as /etc/ntp.conf or /etc/mail/aliases can be enhanced with post-distribution scripts. For example, such a post-distribution script may run the **`sendmail -bi`** command to rebuild the sendmail aliases database.
- Central management of application-specific configuration files, such as the /usr/tivoli/tsm/client/ba/bin/dsm.sys file, used by Tivoli Storage Manager.
- Distribution and central control of other node specific files, such as /etc/motd, Example 3-43 on page 118 presents a sample of our /cfmroot/etc directory,

and, as you can see it is convenient to have specific files for given node groups.

Example 3-43 Part of our /cfmroot/etc directory on our MS

```
root@csm_ms01:/cfmroot/etc# ls
environment._gr_virt_aix
motd._virt_p1
motd._virt_p7
hosts._gr_cluster1
hosts._gr_cluster2
group motd._virt_p2
passwd
netsvc.conf._gr_virt_aix
hosts motd._virt_p3
ntp.conf._gr_virt_aix
...
```

Identifying the nodes by /etc/motd or welcome messages is a local policy decision. Some environments leave them as default, some customers update the defaults for security reasons. In our environment where there are more people working on the same systems, we wanted the users to understand which partitions are related to others, thus we provide identification of our LPARs by /etc/motd, as shown in Figure 3-27. We also customized our VIOS LPARs by **loginmsg** and **motd** commands.



```
*********************************************************************
*                                                                   *
*          Hostname:  virt_p3                                       *
*                                                                   *
*          Machine:   p550                                          *
*          LPAR:      partition3                                    *
*          Type:      MP with virtual IO                            *
*          Cluster:   CLUSTER2 p550 x p570  virt_p3 <--> virt_p6    *
*                                                                   *
*********************************************************************
```

Figure 3-27 /etc/motd customization

3.4 AIX Installation strategy

There are multiple, different ways to install AIX. Each has its own benefits, and will be better suited to a specific type of environment. In this section we summarize some of the ways to install AIX. It is not only important to install AIX but also the implications of *how* that goal is achieved.

We list some example scenarios to illustrate the variety of AIX installation methods:

1. Installing two independent standalone servers at a single customer location.
2. Installing 10 servers with custom applications in 10 different sites around the world, without any common network connectivity between them.
3. Installing 128 identical nodes for HPC purposes at a single site.
4. Installing 12 micropartitions including two VIO servers within a single IBM System p5 server.
5. Installing an eight-node HACMP cluster, with supporting applications, such as DB2 or WebSphere.
6. Building the complete backup and restore infrastructure for 10 IBM System p5 servers, including VIOS and IVM LPARs. Strategy must clearly include defined processes for backup and restore. Such operations must complete within a given time, for example an LPAR must be restored within a two-hour window.

While choosing your installation strategy you should also consider the need for a backup/restore strategy. Check the following issues for planning:

- What is the amount of nodes that need to be installed at once or in a determined time frame?
- Could a single image be cloned to install the base operating system on all nodes?
- Same application software must be deployed on al or on a large number of nodes?
- If application software to be installed is diverse, can you identify the amount of customization needed (post-install)?

If, after answering the previous questions you decide that cloning is the way to install your AIX nodes, you should note that a cloning approach may present the following advantages:

- You do not have to manually install required applications on every node; as they will be part of the cloned-image.
- Your applications and operating system will be installed in a standard uniform configuration. This is important for example in HACMP clusters where you want to have the same environment including software levels and configuration on all cluster nodes. However, if your master image (to be

cloned) has a configuration mistake, this will be replicated to any newly installed node.

Cloning also has also disadvantages, but this generally is a subjective topic. For example, someone familiar with CSM is likely to consider cloning an outdated technique, whereas someone who is only familiar with installing AIX via CD may find cloning an interesting technique. It may also happen that some configuration is not exactly the same on a cloned target. We experienced problems with the Asynchronous I/O setting which was disabled on the cloned images which hosted database applications dependent on that setting. In some scenarios, cloning can save time; in others it can add complications or manual effort.

In the following sections we outline some methods of AIX installation.

3.4.1 Standard media installation

This is the traditional approach. It requires physical access to target machine and manual intervention. System to be installed must have an available CD or DVD drive, therefore LPAR reconfiguration may be required to reassign media drawer to the appropriate storage controller. Only one server can be installed at a given time. All customization and configuration must be performed separately.

3.4.2 Custom media creation

You can also install AIX onto a server, configure the operating system, per your requirements, install required applications, and then create a bootable CD or DVD via the `mksysb` command. This could be considered a form of cloning. Your server will require a writable optical device.

> **Note:** For more information about creating bootable install media, refer to *AIX 5L V5.3 Installing AIX*, SC23-4887-02.

3.4.3 AIX alternate disk mksysb installation

Alternate disk installation is a powerful feature of AIX. It allows you to install or upgrade an operating system while it is up and running. This feature utilizes the AIX mksysb command. The operation can occur concurrently and it requires a target disk in addition to the source one. The procedure creates a mksysb of the running system and then restores onto the target disk. This is done by a single command. The same process can be used for migration installations (such as upgrading an AIX 5L V5.2 system to AIX 5L V5.3). Once the alternate image disk has been generated, reboot your server from the target disk to use the upgraded installation. Target disks can be physically or logically reallocated to other hosts.

This feature is particularly effective where environments use SAN or VIOS-hosted storage for their rootvg volume groups. You just have to change the SAN zoning and/or LUN masking to a target host.

> **Note:** Alternate disk installation is documented in detail within *AIX 5L V5.3 Installing AIX*, SC23-4887-02.

3.4.4 Network Installation Manager (NIM)

NIM is an often overlooked AIX feature. It has a reputation of being complicated to configure and use. However, once users become familiar with NIM they do not return to previous installation methods. Even though NIM provides many features and functions, you do not have to use all of them. While using NIM in environment with large numbers of servers requires special administration, for routine AIX installations it is very simple to setup and use. You can actually install a new AIX LPAR, then configure it as an operational NIM server within a couple of hours.

In previous sections we have already demonstrated benefits of NIM. In summary:

- NIM provides the installation of an AIX base operation system as shown in 3.3.11, “CSM Integration with the existing NIM server” on page 108. Note that CSM enhances the experience by automating hardware control during the network boot phase. Furthermore you can create specific customization scripts to be run after base install or first reboot, as shown in Example 3-44 on page 122.

  NIM also provides extensive flexibility for the customization of the installed client. It is able to configure AIX during the install process. For example using NIM you can automatically define and enable dump and paging space devices.
- NIM can run operations in parallel, making possible to install multiple servers concurrently.
- Software inventory reports. As shown in Figure 3-14 on page 84, it is possible to compare installed software levels on multiple nodes with existing NIM lpp_source repositories.
- Maintaining multiple software levels. Although not mentioned previously, NIM can easily support multiple lpp_source install resources, therefore allowing clients to be installed with different AIX versions or Technology Levels.
- Software bundles. You can create bundles for operating system components or additional components, such as SSH, as shown in 3.3.1, “CSM Version 1.5 installation using NIM” on page 82.

- Grouping nodes and resources for easier allocation and management. Refer to Example 3-44.
- Simplified NIM installation. EZNIM provides a practical interface to NIM; it offers the administrator task-oriented menus, hiding some of the complex operations.

> **Note:** NIM and EZNIM are documented in detail in *AIX 5L V5.3 Installing AIX*, SC23-4887-02. We particularly recommend the “NIM Task Roadmap” as a useful reference.

- Possibility to install and restore IVM and VIOS LPARs. This integration allows NIM to be a common management tool for your entire IBM System p5 environment.

Example 3-44 NIM Resource Group basic_res_grp example

```
Change/Show Characteristics of a Group
Type or select values in entry fields.
Press Enter AFTER making all desired changes.
[Entry Fields]
  Group Name                                         [basic_res_grp]
  Members to Remove                                  []
  Group Type                                          res_group
  SPOT       (Shared Product Object Tree)            [SPOT_53TL5]
  LPP_SOURCE (source for optional product images)    [LPP_AIX53_TL5]
  INSTALLP_BUNDLE (an installp bundle file)          []
  SCRIPT       (file which is executed on clients)   []
  BOSINST_DATA (config file for bos_inst operation)
[BOSINST_53NOPROPMT]
  IMAGE_DATA   (config file for bos_inst operation)  []
  VG_DATA      (config file for restvg operation.)   []
  RESOLV_CONF  (config file for name-server info.)   []
  MKSYSB       (a mksysb image)                      []
  SAVEVG       (an AIX savevg image)                 []
  FB_SCRIPT                                          []
  FIX_BUNDLE   (fix keyword input file)              []
  ROOT   (parent dir. for client / (root) dirs.)     []
  PAGING (parent dir. for client paging files)       []
  DUMP   (parent dir. for client dump files)         []
  HOME   (parent dir. for client /home dirs.)        []
  SHARED_HOME (/home dir. shared by clients)         []
  TMP    (parent dir. for client /tmp dir)           []
  ADAPTER_DEF (secondary adapter config files dir.)  []
  Use this Group for Default Allocation?             []
```

```
 Comments                                                []
```

> **Note:** For more detailed NIM information, refer to *NIM From A to Z in AIX 5L*, SG24-7296. This book presents examples that describe how to use NIM in a firewall environment and how NIM and VIOS integrate.

Going back to our example environments in 3.4, “AIX Installation strategy” on page 118, we can conclude that NIM would NOT be suitable for the first two scenarios, but very suited to the latter four. In contrast, custom DVD media would be a better choice for the first two scenarios. Also, for some customers, alternative disk installation is an easier concept to implement, compared to NIM, however, NIM can leverage normal mksysb (in addition to alternate disk) and CSM integration. It depends on your viewpoint, environment and how much assistance you desire from the technology.

3.4.5 PLM implementation

We consider Partition Load Manager useful in our particular scenario as it can manage virtual processors and entitlement capacity within IBM System p5 machines, and it is also is one of the few system management tools able to dynamically change resources inside IBM System p4 machines.

An alternative would be to use HACMP events to dynamically change CPU or memory resources. However such an approach would have the following limitations:

- All targeted nodes must be members of the HACMP cluster which may increase the number of required HACMP licenses for your environment.
- Resource changes occur only on HACMP event, such as *node_down* or HACMP resource group movement (*rg_move*).

We implemented PLM on the CSM management server so that we have single point of management, as shown in Figure 3-28 on page 124.

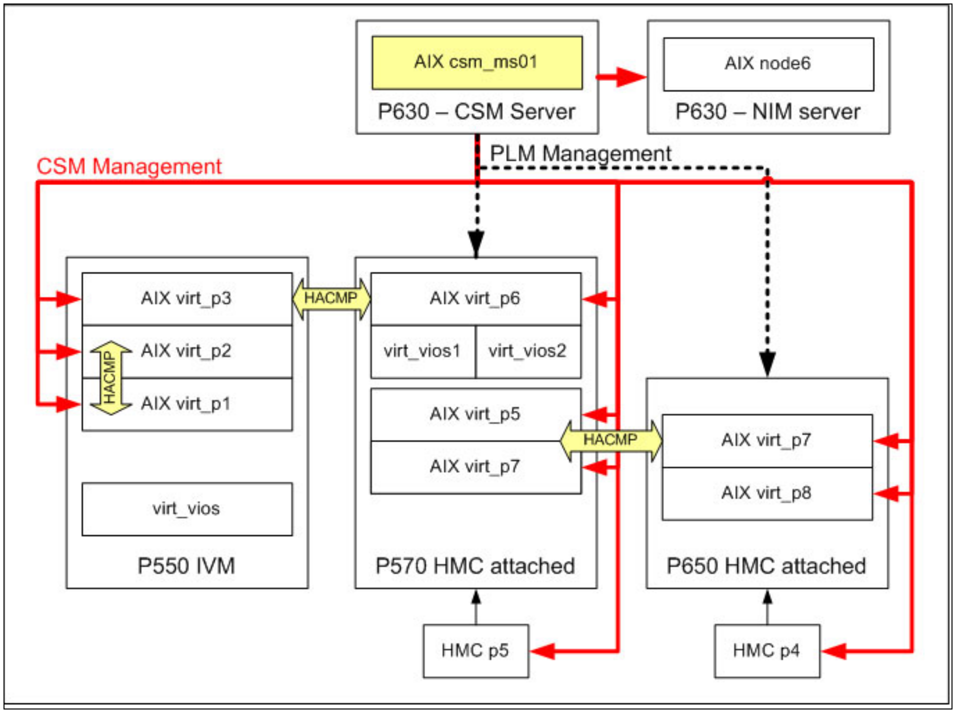


Figure 3-28 Scenario of the CSM and PLM management

We are running PLM in two instances, each for the System p5 box. At the time of writing, it is not possible to manage IVM LPARs through PLM. In some situations PLM can be used to reduce the number of idle CPUs used by licensed programs, therefore reducing the per-CPU application licensing costs.

For for detailed discussions of PLM configuration and implementation, refer to the following books:

- *Partitioning Implementations for IBM p5 Servers*, SG24-7039. Section 3.4 provides a practical comparison between the features of IBM System p Hypervisor and PLM.
- *Advanced POWER Virtualization on IBM System p5*, SG24-7940. Chapter 7 describes installation and configuration of PLM.

4

High availability and resource management scenarios

This chapter discusses a number of example implementations that illustrate the configurations and advantages of the given scenarios.

- “HACMP scenarios, one IVM-managed CEC” on page 126
- “IBM System p5 scenario using two CECs” on page 175

4.1 HACMP scenarios, one IVM-managed CEC

In this section we discuss and implement a small HACMP cluster using LPARs located within a single IBM System p5 server. We are looking for a cost-effective solution which provides a high degree of availability for its hosted application servers. The principles of this scenario are also applicable if you have two existing application servers that you want to integrate into a highly-available environment.

4.1.1 Physical resources used

- Our test hardware was a single entry level IBM System p5 p550 system having three shared-processor LPARs:
  - One LPAR which hosts the VIOS/IVM.
  - Second and third LPARs run AIX 5L V5.3. They will access external resources through the VIOS.
- Our IBM System p5 p550 system is managed using IVM Version 1.3.0. While our testing was performed with the current version, we also tested an upgrade between versions. Refer to 2.1.3, "Integrated Virtualization Manager (IVM)" on page 20, to see how this affects an installed system.
- Only the VIOS LPAR is connected to external storage by using two Fibre Channel adapters. We used redundant adapters to eliminate a potential single point of failure, while keeping costs at a minimum.
- Also, only the VIOS LPAR is physically connected to the external IP network. We used three network adapters in the following configuration:
  - One network adapter is assigned a fixed IP address and is used for management and testing purposes.
  - The second and the third adapters are used to define an EtherChannel adapter that is bridged to virtual interfaces contained within the virtual partitions. When operating in Network Interface Backup mode, it is recommended to connect Ethernet cables to separate switches. Otherwise, the Ethernet cables must be connected to the same switch and an additional switch port configuration will be required.

4.1.2 IVM setup

In this scenario both client LPARs use services provided by the VIOS/IVM partition. Both LPARs are also defined as nodes of an HACMP cluster. As shown in Figure 4-1 on page 127, each LPAR has a single virtual network adapter named `ent0` that is bridged to the external network via a shared Ethernet adapter configured within the IVM partition. The Shared Ethernet Adapter (SEA) is named `ent8` and is mapped between the virtual trunk Ethernet adapter named `ent3` and the link aggregation (LA) adapter named `ent7`, which is configured to use two physical Ethernet adapters, `ent1` and `ent2`.

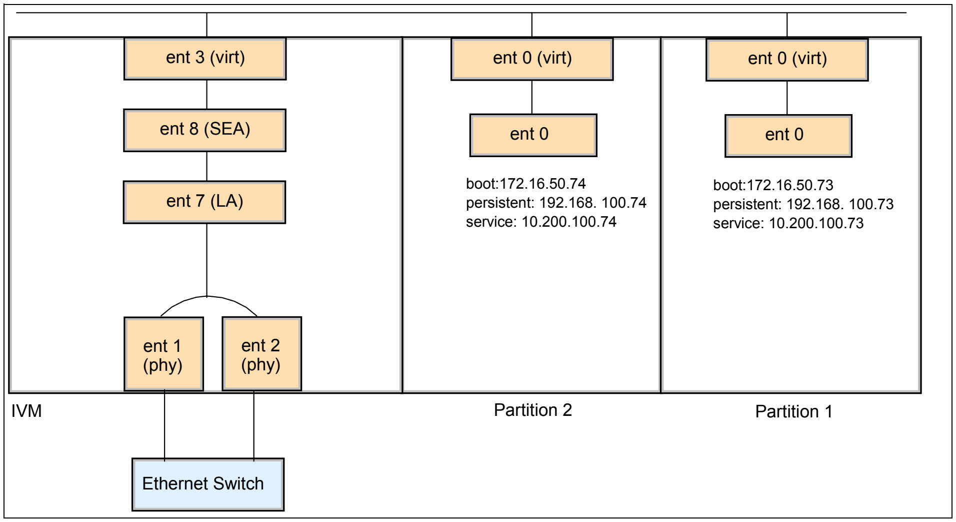


Figure 4-1 Virtual Ethernet and HACMP scenario using one CEC

As shown in Figure 4-2 on page 128, each client partition has two shared disks. These are managed by the IVM LPAR and made available to both client partitions.

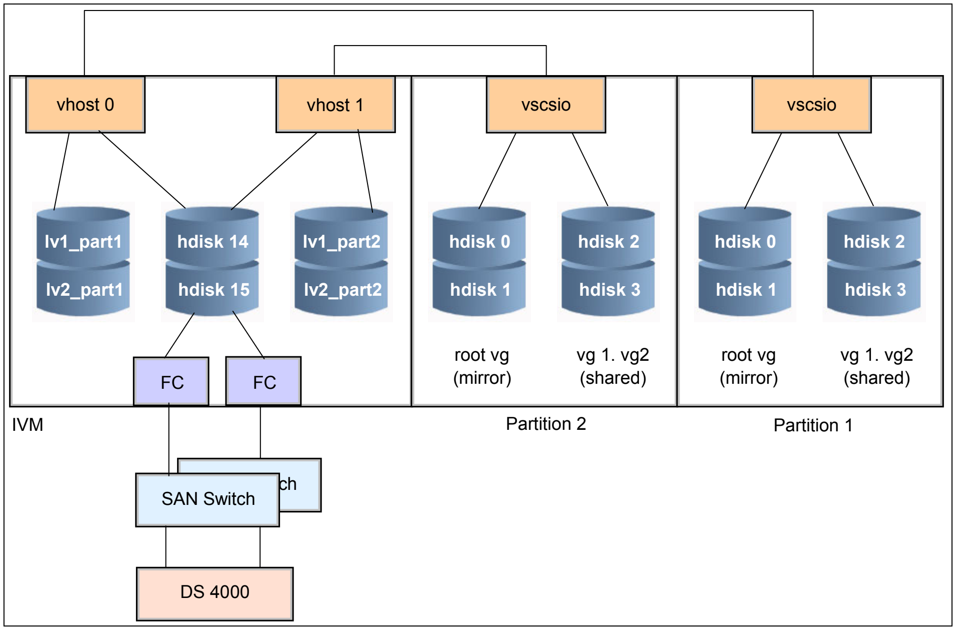


Figure 4-2 Virtual SCSI and HACMP scenario using one CEC

Table 4-1 shows the clients' partitions hdisk mapping in IVM.

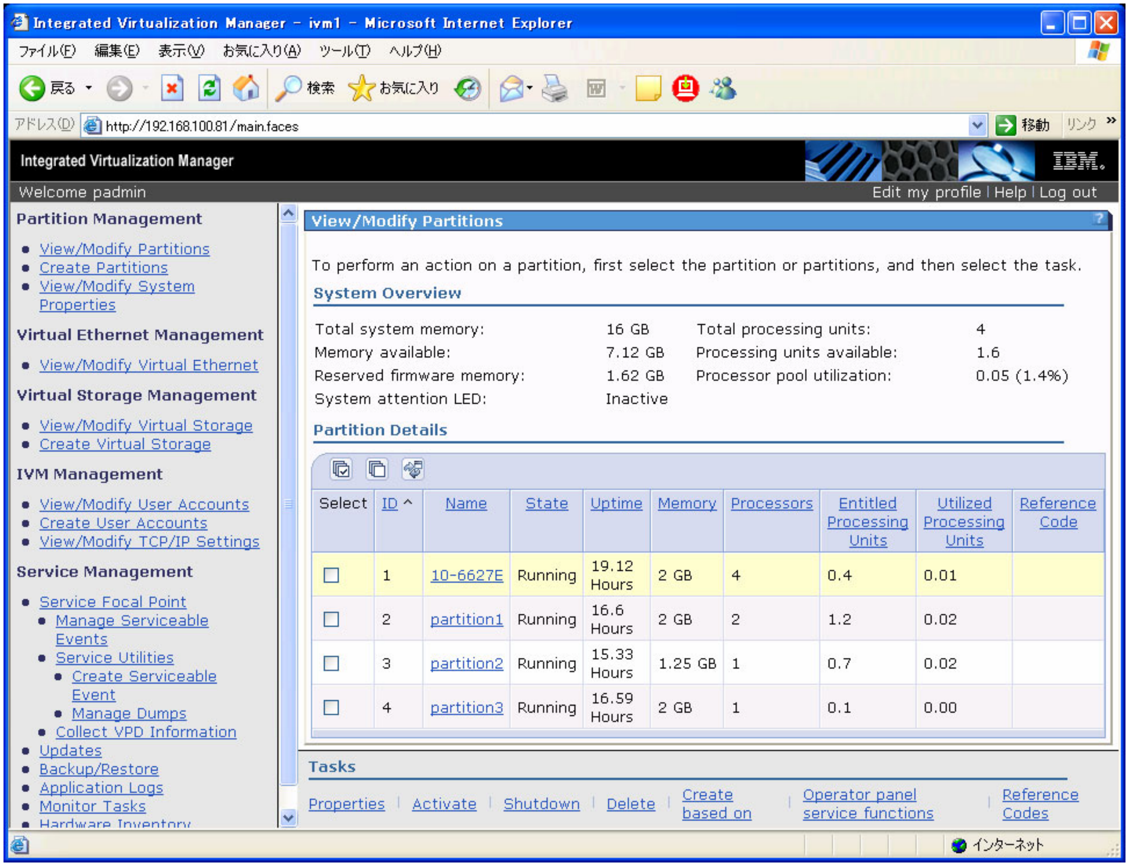
Table 4-1 Virtual SCSI mapping

partition	hdisk	backing device in IVM
partition1	hdisk0	lv1_part1 (Logical Volume created in hdisk1)
	hdisk1	lv2_part1 (Logical Volume created in hdisk5)
	hdisk2	hdisk14 (shared)
	hdisk3	hdisk15 (shared)
partition2	hdisk0	lv1_part2 (Logical Volume created in hdisk2)
	hdisk1	lv2_part2 (Logical Volume created in hdisk6)
	hdisk2	hdisk14 (shared)
	hdisk3	hdisk15 (shared)

4.1.3 Basic LPAR configuration

> **Note:** Prior to IVM install and configuration you must ensure your IBM System p5 meets the prerequisites. System re-configuration or microcode upgrades may be required. For more information refer to the VIOS/IVM release notes on the Virtual I/O Server support site:
>
> `http://techsupport.services.ibm.com/server/vios/home.html`

After we carved our LPARs, the final configuration as seen in the IVM GUI is shown in Figure 4-3. There are actually four LPARs including a VIOS partition, but `partition3` is not used in this scenario.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る
検索
お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動
リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
Create User Accounts
View/Modify TCP/IP Settings
Service Management
Service Focal Point
Manage Serviceable Events
Service Utilities
Create Serviceable Event
Manage Dumps
Collect VPD Information
Updates
Backup/Restore
Application Logs
Monitor Tasks
Hardware Inventory
インターネット

View/Modify Partitions

To perform an action on a partition, first select the partition or partitions, and then select the task.

System Overview

Total system memory:	16 GB	Total processing units:	4
Memory available:	7.12 GB	Processing units available:	1.6
Reserved firmware memory:	1.62 GB	Processor pool utilization:	0.05 (1.4%)
System attention LED:	Inactive		

Partition Details

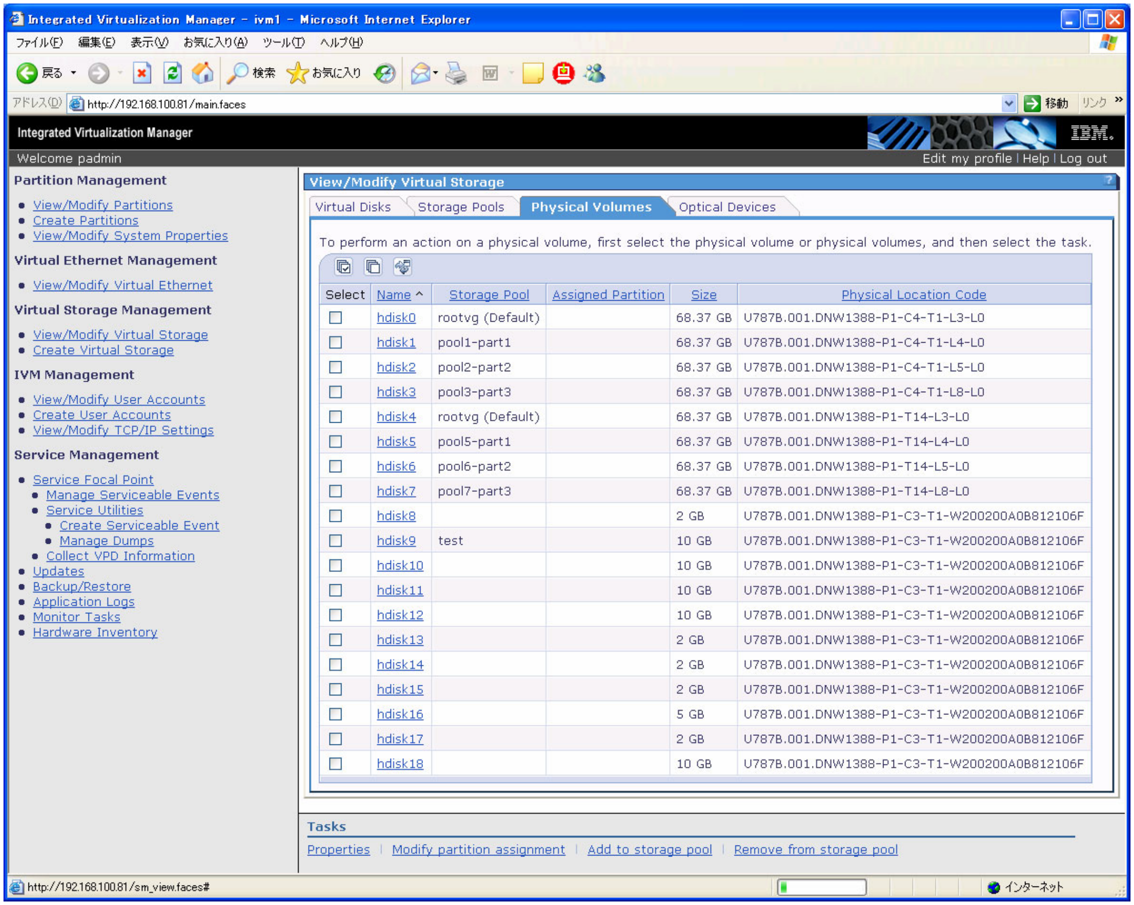
Select	ID ^	Name	State	Uptime	Memory	Processors	Entitled Processing Units	Utilized Processing Units	Reference Code
□	1	10-6627E	Running	19.12 Hours	2 GB	4	0.4	0.01	
□	2	partition1	Running	16.6 Hours	2 GB	2	1.2	0.02	
□	3	partition2	Running	15.33 Hours	1.25 GB	1	0.7	0.02	
□	4	partition3	Running	16.59 Hours	2 GB	1	0.1	0.00	

Tasks

Properties | Activate | Shutdown | Delete | Create based on | Operator panel service functions | Reference Codes

Figure 4-3 LPAR configuration

There are eight (8) internal disks (`hdisk0` to `hdisk7`) and 11 external disks (`hdisk8` to `hdisk18`) defined in IVM, as shown in Figure 4-4. The external disks are LUNs hosted in an IBM System Storage DS4000™.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動 リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
Create User Accounts
View/Modify TCP/IP Settings
Service Management
Service Focal Point
Manage Serviceable Events
Service Utilities
Create Serviceable Event
Manage Dumps
Collect VPD Information
Updates
Backup/Restore
Application Logs
Monitor Tasks
Hardware Inventory
View/Modify Virtual Storage
Virtual Disks
Storage Pools
Physical Volumes
Optical Devices
http://192.168.100.81/sm_view.faces#
インターネット

To perform an action on a physical volume, first select the physical volume or physical volumes, and then select the task.

Select	Name ^	Storage Pool	Assigned Partition	Size	Physical Location Code
□	hdisk0	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L3-L0
□	hdisk1	pool1-part1		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L4-L0
□	hdisk2	pool2-part2		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L5-L0
□	hdisk3	pool3-part3		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L8-L0
□	hdisk4	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-T14-L3-L0
□	hdisk5	pool5-part1		68.37 GB	U787B.001.DNW1388-P1-T14-L4-L0
□	hdisk6	pool6-part2		68.37 GB	U787B.001.DNW1388-P1-T14-L5-L0
□	hdisk7	pool7-part3		68.37 GB	U787B.001.DNW1388-P1-T14-L8-L0
□	hdisk8			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk9	test		10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk10			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk11			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk12			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk13			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk14			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk15			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk16			5 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk17			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk18			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F

Tasks

Properties | Modify partition assignment | Add to storage pool | Remove from storage pool

Figure 4-4 IVM physical volumes

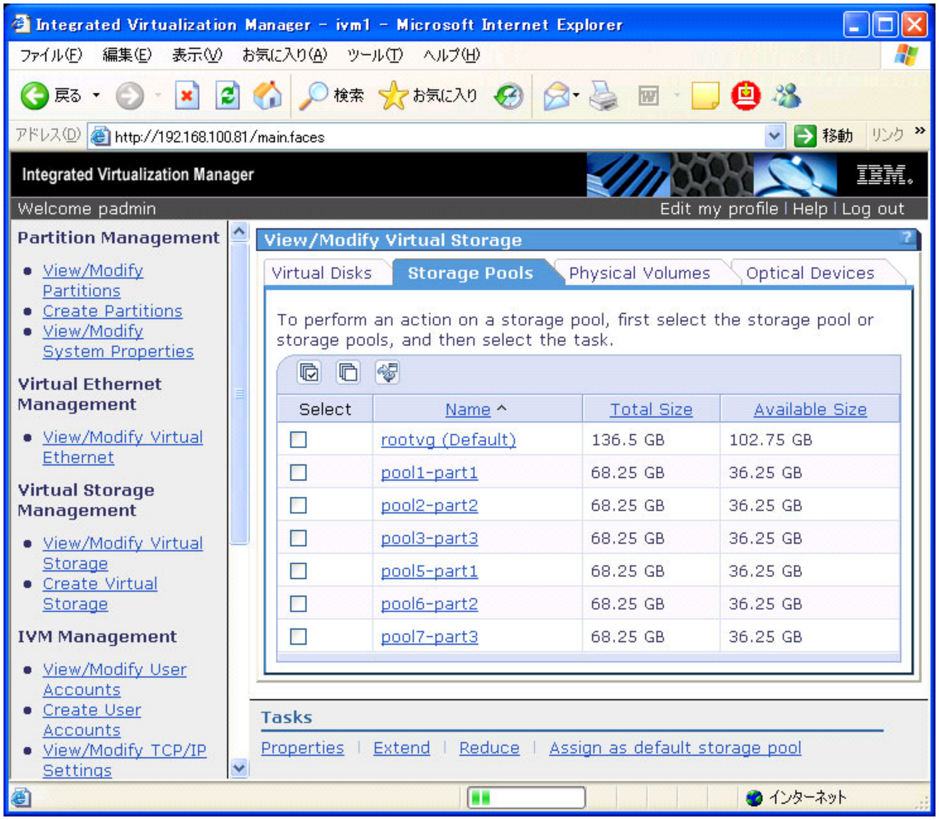
The `hdisk0` and `hdisk4` are included in IVM's `rootvg` volume group which uses LVM mirroring, as shown in Example 4-1.

Example 4-1 rootvg mirroring in IVM partition

```
$ lsvg -lv rootvg
rootvg:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
hd5                 boot       1     2     2    closed/syncd  N/A
hd6                 paging     4     8     2    open/syncd    N/A
paging00            paging     8     16    2    open/syncd    N/A
```

```
hd8                 jfs2log     1     2     2     open/syncd     N/A
hd4                 jfs2        2     4     2     open/syncd     /
hd2                 jfs2        10    20    2     open/syncd     /usr
hd9var              jfs2        5     10    2     open/syncd     /var
hd3                 jfs2        11    22    2     open/syncd     /tmp
hd1                 jfs2        80    160   2     open/syncd     /home
hd10opt             jfs2        1     2     2     open/syncd     /opt
lg_dumplv           sysdump     8     8     1     open/syncd     N/A
fwdump              jfs2        3     6     2     open/syncd     /var/adm/ras/platform
```

All other internal disks (`hdisk1-3`,and `hdisk5-7`) belong to different storage pools. So there are six storage pools, as shown in Figure 4-5.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る 検索 お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動 リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
Create User Accounts
View/Modify TCP/IP Settings
View/Modify Virtual Storage
Virtual Disks
Storage Pools
Physical Volumes
Optical Devices
Tasks
Properties | Extend | Reduce | Assign as default storage pool
インターネット

To perform an action on a storage pool, first select the storage pool or storage pools, and then select the task.

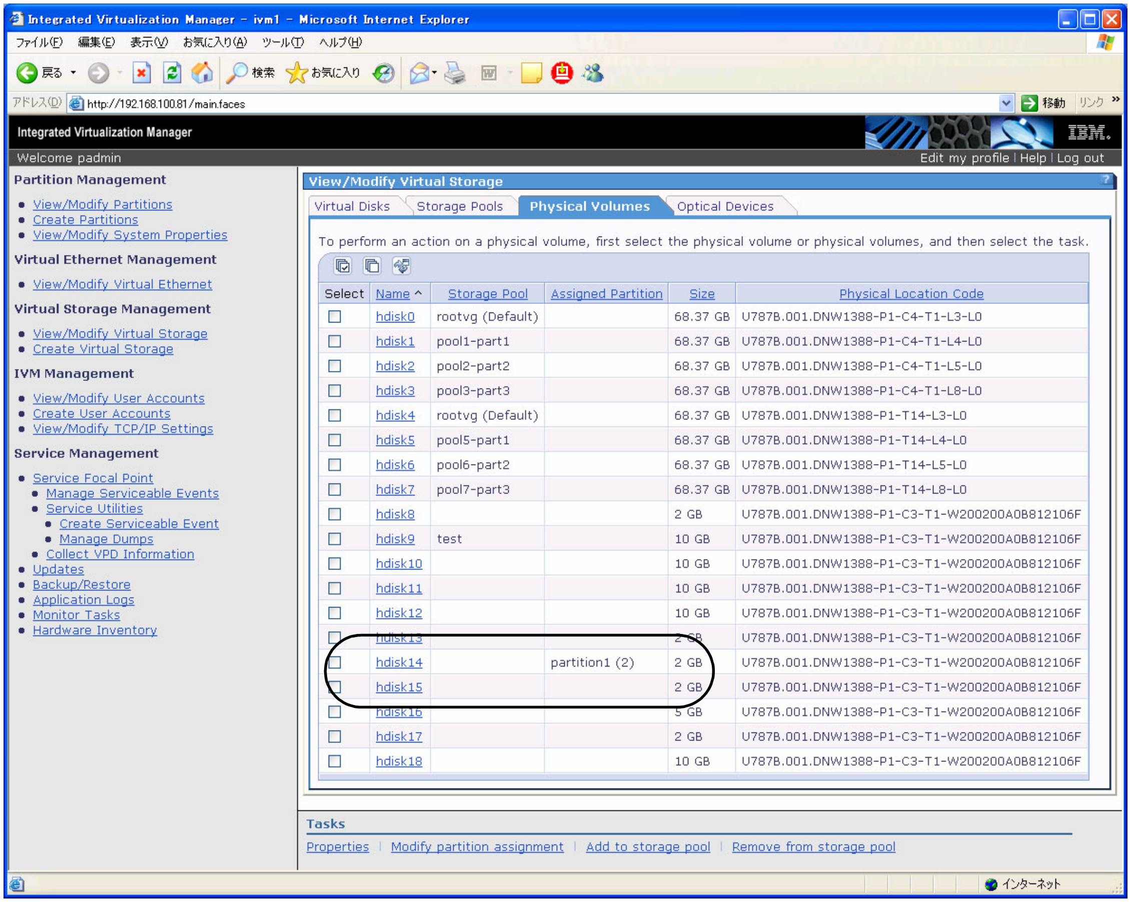
Select	Name ^	Total Size	Available Size
□	rootvg (Default)	136.5 GB	102.75 GB
□	pool1-part1	68.25 GB	36.25 GB
□	pool2-part2	68.25 GB	36.25 GB
□	pool3-part3	68.25 GB	36.25 GB
□	pool5-part1	68.25 GB	36.25 GB
□	pool6-part2	68.25 GB	36.25 GB
□	pool7-part3	68.25 GB	36.25 GB

Figure 4-5 IVM storage pools

Each storage pool has one logical volume. Two logical volumes are exported to each client partition so that each LPAR has two physically separated disks. Each client partition uses LVM mirroring.

4.1.4 Mapping shared disks to multiple partitions

In this scenario, each client partition has two shared disks. To configure these two disks on the IVM server need to be mapped to two partitions. First, we assign one hdisk to one partition. This operation can be performed using either the GUI or the CLI. We assign `hdisk14` to `partition1`, the result is shown in Figure 4-6.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る
検索
お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動
リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
Create User Accounts
View/Modify TCP/IP Settings
Service Management
Service Focal Point
Manage Serviceable Events
Service Utilities
Create Serviceable Event
Manage Dumps
Collect VPD Information
Updates
Backup/Restore
Application Logs
Monitor Tasks
Hardware Inventory
View/Modify Virtual Storage
Virtual Disks
Storage Pools
Physical Volumes
Optical Devices
Tasks
Properties
Modify partition assignment
Add to storage pool
Remove from storage pool
インターネット

To perform an action on a physical volume, first select the physical volume or physical volumes, and then select the task.

Select	Name ^	Storage Pool	Assigned Partition	Size	Physical Location Code
□	hdisk0	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L3-L0
□	hdisk1	pool1-part1		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L4-L0
□	hdisk2	pool2-part2		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L5-L0
□	hdisk3	pool3-part3		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L8-L0
□	hdisk4	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-T14-L3-L0
□	hdisk5	pool5-part1		68.37 GB	U787B.001.DNW1388-P1-T14-L4-L0
□	hdisk6	pool6-part2		68.37 GB	U787B.001.DNW1388-P1-T14-L5-L0
□	hdisk7	pool7-part3		68.37 GB	U787B.001.DNW1388-P1-T14-L8-L0
□	hdisk8			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk9	test		10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk10			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk11			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk12			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk13			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk14		partition1 (2)	2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk15			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk16			5 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk17			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk18			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F

Figure 4-6 Assigning an hdisk to one partition

In Figure 4-6, `hdisk14` is assigned only to one partition. To check device mapping using command line, see Example 4-2.

Example 4-2 Virtual resource mapping in IVM

```
$ lsmap -all
SVSA            Physloc                                              Client Partition
ID
```

```
--------------- -------------------------------------------- -----------------
vhost0          U9113.550.106627E-V1-C11                     0x00000002

VTD                   vtscsi0
LUN                   0x8100000000000000
Backing device        lv1_part1
Physloc

VTD                   vtscsi1
LUN                   0x8200000000000000
Backing device        lv2_part1
Physloc

VTD                   vtscsi6
LUN                   0x8300000000000000
Backing device        hdisk14
Physloc
U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L6000000000000

SVSA            Physloc                                      Client Partition
ID
--------------- -------------------------------------------- ------------------
vhost1          U9113.550.106627E-V1-C13                     0x00000003

VTD                   vtscsi2
LUN                   0x8100000000000000
Backing device        lv1_part2
Physloc

VTD                   vtscsi3
LUN                   0x8200000000000000
Backing device        lv2_part2
Physloc
```

When you want assign one disk to two partitions (disk shared between two different partitions), you have to use the command line interface to assign the disk to the second partition.

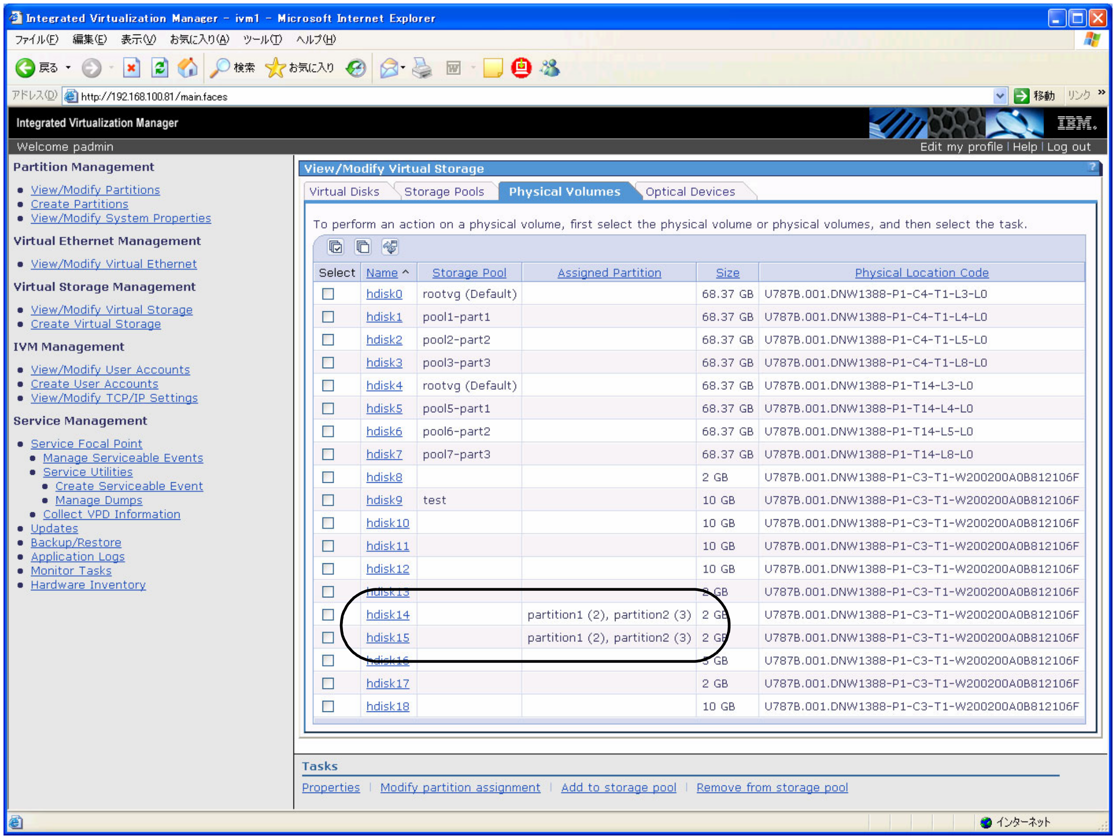
> **Note:** At the time of writing, shared disk configurations will correctly display in the IVM GUI; however they can only be configured from the command line.

As shown in Example 4-2 on page 132, `hdisk14` is mapped to `vhost0`. `vhost0` is connected to `partition1`, whose partition ID is `2`, listed above as “`0x00000002`”. While `vhost1` is connected to `partition2`, whose partition ID is `3`, listed above as “`0x00000003`”. `hdisk14` should also be mapped to `vhost1` so that both `partition1` and `partition2` can share it. We need to create another virtual target device to make hdisk14 connected to another virtual host adapter, vhost1. To connect backing device to another virtual host adapter, you should use mkvdev command. Be sure to execute mkvdev command with “-f” option, otherwise you would get error message as shown in Example 4-3.

Example 4-3 create virtual target device for shared disk

```
$ mkvdev -vdev hdisk14 -vadapter vhost1 -dev vtscsi7
hdisk14 is already being used as a backing device.  Specify the -f flag
to force this device to be used anyway.
$ mkvdev -f -vdev hdisk14 -vadapter vhost1 -dev vtscsi7
vtscsi7 Available
```

Now hdisk14 is ready to be accessed from both partition1 and partition2. In our scenario, hdisk15 also should be mapped to two virtual host adapters by the same procedure as hdisk14; such that partition1 and partition2 can share hisk14 and hdisk15 from IVM partition. The final configuration is shown using the Web interface in Figure 4-7 on page 135.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る
検索
お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動
リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
Create User Accounts
View/Modify TCP/IP Settings
Service Management
Service Focal Point
Manage Serviceable Events
Service Utilities
Create Serviceable Event
Manage Dumps
Collect VPD Information
Updates
Backup/Restore
Application Logs
Monitor Tasks
Hardware Inventory
View/Modify Virtual Storage
Virtual Disks
Storage Pools
Physical Volumes
Optical Devices
Tasks
Properties | Modify partition assignment | Add to storage pool | Remove from storage pool
インターネット

To perform an action on a physical volume, first select the physical volume or physical volumes, and then select the task.

Select	Name ^	Storage Pool	Assigned Partition	Size	Physical Location Code
□	hdisk0	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L3-L0
□	hdisk1	pool1-part1		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L4-L0
□	hdisk2	pool2-part2		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L5-L0
□	hdisk3	pool3-part3		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L8-L0
□	hdisk4	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-T14-L3-L0
□	hdisk5	pool5-part1		68.37 GB	U787B.001.DNW1388-P1-T14-L4-L0
□	hdisk6	pool6-part2		68.37 GB	U787B.001.DNW1388-P1-T14-L5-L0
□	hdisk7	pool7-part3		68.37 GB	U787B.001.DNW1388-P1-T14-L8-L0
□	hdisk8			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk9	test		10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk10			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk11			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk12			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk13			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk14		partition1 (2), partition2 (3)	2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk15		partition1 (2), partition2 (3)	2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk16			5 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk17			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk18			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F

Figure 4-7 Assigning hdisks to multiple partitions

The same information shown using CLI can be observed in Example 4-4.

Example 4-4 mapping after creating second target device

```
$ lsmap -all
SVSA            Physloc                                            Client
Partition ID
--------------- -------------------------------------------- ----------
vhost0          U9113.550.106627E-V1-C11                         0x00000002

...

VTD                   vtscsi6
LUN                   0x8300000000000000
Backing device        hdisk14
```

```
Physloc
U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L6000000000000

VTD                   vtscsi8
LUN                   0x8400000000000000
Backing device        hdisk15
Physloc
U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L7000000000000

SVSA            Physloc                                       Client
Partition ID
--------------- --------------------------------------------- ----------
vhost1          U9113.550.106627E-V1-C13                      0x00000003

...

VTD                   vtscsi7
LUN                   0x8300000000000000
Backing device        hdisk14
Physloc
U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L6000000000000

VTD                   vtscsi9
LUN                   0x8400000000000000
Backing device        hdisk15
Physloc
U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L7000000000000
```

4.1.5 Defining link aggregation

To overcome the bandwidth limitation of a single network adapter or to eliminate the network adapter as a single point of failure, you can define link aggregation between two or more physical adapters within IVM partition.

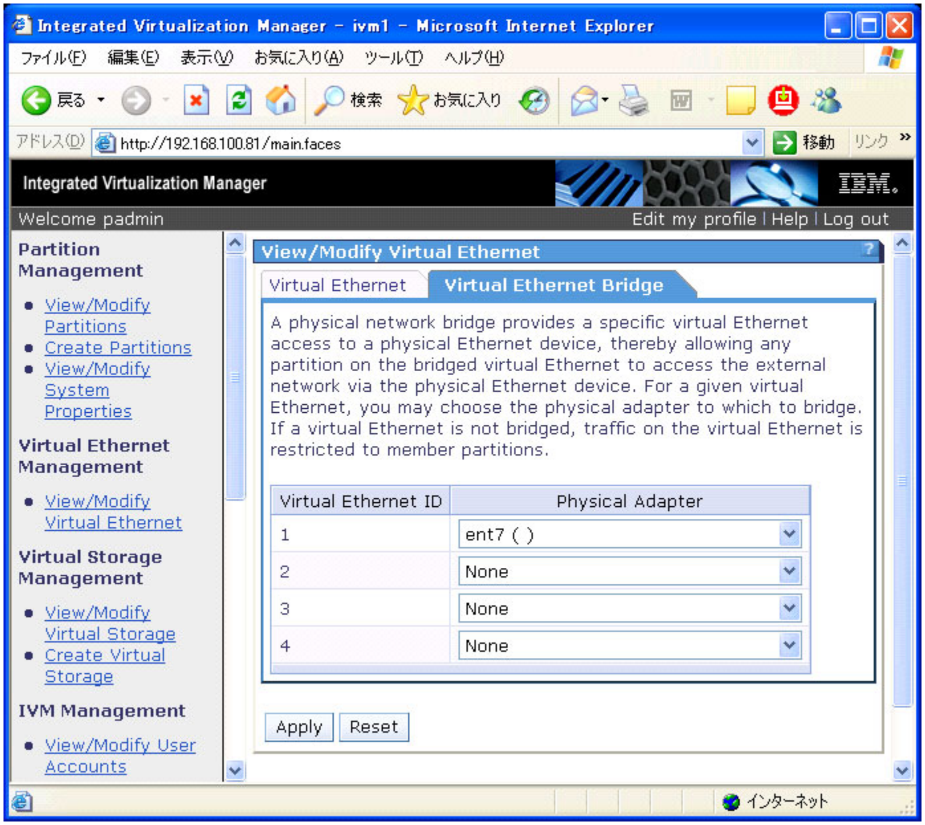
Link aggregation devices can be defined only from the command line interface. In this scenario, a link aggregation adapter is defined using the single-port PCI adapter and one of the two integrated Ethernet ports. The Ethernet device on PCI adapter (`ent2`) is defined as the primary adapter and the integrated Ethernet port (`ent1`) is defined as backup adapter, as shown in Example 4-5 on page 137.

We used the PCI-adapter and one integrated Ethernet adapter because the PCI adapter are hot-pluggable, as opposed to the integrated Ethernet ports which are not. Therefore using both integrated Ethernet adapters, would have introduced an additional single point of failure.

Example 4-5 Creating link aggregation device

```
$ mkvdev -lnagg ent2 -attr backup_adapter=ent1
ent7 Available
$ lsdev | grep ent
ent0            Available  2-Port 10/100/1000 Base-TX PCI-X Adapter (1410890
ent1            Available  2-Port 10/100/1000 Base-TX PCI-X Adapter (1410890
ent2            Available  10/100/1000 Base-TX PCI-X Adapter (14106902)
ent3            Available  Virtual I/O Ethernet Adapter (l-lan)
ent4            Available  Virtual I/O Ethernet Adapter (l-lan)
ent5            Available  Virtual I/O Ethernet Adapter (l-lan)
ent6            Available  Virtual I/O Ethernet Adapter (l-lan)
ent7            Available  EtherChannel / IEEE 802.3ad Link Aggregation
ibmvmc0         Available  Virtual Management Channel
$ lsdev -dev ent7 -attr
attribute       value          description
user_settable

adapter_names   ent2           EtherChannel Adapters                        True
alt_addr        0x000000000000 Alternate EtherChannel Address               True
auto_recovery   yes            Enable automatic recovery after failover     True
backup_adapter  ent1           Adapter used when whole channel fails        True
hash_mode       default        Determines how outgoing adapter is chosen    True
mode            standard       EtherChannel mode of operation               True
netaddr         0              Address to ping                              True
noloss_failover yes            Enable lossless failover after ping failure True
num_retries     3              Times to retry ping before failing           True
retry_time      1              Wait time (in seconds) between pings         True
use_alt_addr    no             Enable Alternate EtherChannel Address        True
use_jumbo_frame no             Enable Gigabit Ethernet Jumbo Frames         True
```

We have now a link aggregation device named `ent7`, as shown in Example 4-5. For each client partition to communicate with external networks, you need to bridge the internal virtual Ethernet interface to an external network interface; this is accomplished by selecting the corresponding virtual Ethernet adapter and link aggregation device, as shown in Figure 4-8 on page 138.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る
検索
お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動
リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
View/Modify Virtual Ethernet
Virtual Ethernet
Virtual Ethernet Bridge
Apply
Reset
インターネット

A physical network bridge provides a specific virtual Ethernet access to a physical Ethernet device, thereby allowing any partition on the bridged virtual Ethernet to access the external network via the physical Ethernet device. For a given virtual Ethernet, you may choose the physical adapter to which to bridge. If a virtual Ethernet is not bridged, traffic on the virtual Ethernet is restricted to member partitions.

Virtual Ethernet ID	Physical Adapter
1	ent7 ( )
2	None
3	None
4	None

Figure 4-8 Virtual Ethernet bridge

In this scenario, both client partitions have one virtual Ethernet adapter whose Virtual Ethernet ID is 1. They can communicate with external network via link aggregation adapter (`ent7`) of the IVM server.

4.1.6 Configuring the HACMP cluster

In this section we show the design and implementation of a very basic HACMP cluster. We kept the configuration very simple because we only wanted to illustrate a small practical example; our configuration demonstrates how to implement an inexpensive cluster that has a satisfactory degree of availability. Its complexity can be increased with additional cluster components, such as nodes, resource groups, volume groups or file systems.

Our example could be well suited for a test or development environment which typically does not require a business-critical level of availability. What we have is a compromise between cost and availability.

Cluster resources

- The cluster comprises 2 nodes, each hosted on a logical partition.
- Each cluster node has one virtual network adapter. Since we have logical partitions with no dedicated physical hardware resources we can use only one virtual interface. All IP labels can be bound to this interface.
- Each cluster node is assigned a resource group.
- Each resource group contains one volume group.
- Each volume group contains a logical volume. On each logical volume there is a single defined filesystem.
- There are 2 application servers defined. Each application server writes to a file located on the filesystem comprised in the same resource group.

As you can see we started with a very simple cluster, which does not require advanced knowledge to be defined and configured. Cluster resources can be added or reconfigured at a later date, as required.

Cluster topology

1. IP networks

   Our cluster contains 2 nodes, named `virt1_node1` and `virt2_node2`.

   Each partition has one virtual Ethernet interface. For each node we defined one boot IP address and one persistent IP address as described in Table 4-2.

Table 4-2 Nodes and IP addresses

Node name	Interface name	IP address
virt1_node1	virt_p1_boot1	172.16.50.73
	virt_p1_persistent	192.168.100.73
virt2_node2	virt_p2_boot1	172.16.50.74
	virt_p2_persistent	192.168.100.74

2. Non-IP networks

   In any HACMP environment it is strongly recommended, apart from IP networks, to define also at least one non-IP network. In this way TCP/IP stack is no longer a single point of failure and therefore availability increases. Non-IP network type can be either `RS-232`, `tmssa`, `tmscsi` or `diskhb`. We defined 2 `diskhb` networks using `hdisk2` and `hdisk3`.

   Disks used for disk heartbeat networks need to be part of an enhanced concurrent volume group. The volume group does not need to be included in an HACMP resource group. However do ensure disk names and their corresponding physical volume IDs (PVIDs) are consistent across cluster nodes.

It is a very good practice to test the disk heartbeat network before adding it to cluster topology as shown in Example 4-6.

Example 4-6 Testing disk heartbeat network that uses hdisk2

```
[virt_p1][/]> dhb_read -p hdisk2 -r
Receive Mode:
Waiting for response . . .
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Magic number = -2023406815
Link operating normally

[virt_p2][/]> dhb_read -p hdisk2 -t
Transmit Mode:
Magic number = -2023406815
Detected remote utility in receive mode.  Waiting for response . . .
Magic number = -2023406815
Magic number = -2023406815
Link operating normally
```

For more information that is pertinent to defining, testing, and troubleshooting disk heart beat networks, refer to the HACMP manuals at:

http://publib.boulder.ibm.com/infocenter/clresctr/vxrx/index.jsp

Service IP labels

We have two service IP labels that are used by resource groups.

For a detailed view of cluster topology we recommend using the **cltopinfo** command as shown in Example 4-7.

Example 4-7 Viewing cluster topology using cltopinfo command

```
[virt_p1][/]> cltopinfo -i
IP Label        Network  Type     Node       Address        If   Netmask
=========       =======  ====     ====       =======       ====  =======
virt1_node1_hdisk2_01 net_diskhb_01 diskhb   virt1_node1 /dev/hdisk2    hdisk2
virt1_node1_hdisk3_01 net_diskhb_02 diskhb   virt1_node1 /dev/hdisk3    hdisk3
virt_p1_serv    net_ether_01 ether    virt1_node1 10.200.100.73       255.255.255.0
virt_p2_serv    net_ether_01 ether    virt1_node1 10.200.100.74       255.255.255.0
virt_p1_boot1   net_ether_01 ether    virt1_node1 172.16.50.73   en0  255.255.255.0
virt2_node2_hdisk2_01 net_diskhb_01 diskhb   virt2_node2 /dev/hdisk2    hdisk2
virt2_node2_hdisk3_01 net_diskhb_02 diskhb   virt2_node2 /dev/hdisk3    hdisk3
virt_p1_serv    net_ether_01 ether    virt2_node2 10.200.100.73       255.255.255.0
virt_p2_serv    net_ether_01 ether    virt2_node2 10.200.100.74       255.255.255.0
virt_p2_boot1   net_ether_01 ether    virt2_node2 172.16.50.74   en0  255.255.255.0
```

Applications servers

We have two application servers named `app_server1` and `app_server2` that access data located on volume group vg1 and `vg2` respectively.

Resource groups

We defined resource groups `rg1` and `rg2` that use the corresponding application servers to access the data; which is located on the associated filesystems within allocated volume groups.

Creation of the LVM-layer objects used by the cluster

We show the step-by-step creation of the LVM-layer objects using the resources defined in previous sections. We want to emphasize which LVM entities are important to HACMP. This section maybe helpful if you have existing LVM-resource utilizing applications server, which you plan to integrate into an HACMP environment.

When starting cluster definition from scratch, we recommend the use of the HACMP Cluster-Single Point of Control (C-SPOC) utility to ensure that definitions for all LVM objects are consistent across all cluster nodes. C-SPOC commands provide a logical interface for managing shared LVM components.

We used the following steps to configure our scenario:

- From the IVM partition, configure `hdisk14` to be seen as `hdisk2` on `partition 1` as shown in Example 4-8 on page 142.

Example 4-8 Configuring hdiks2 on partition 1

```
# uname -n
virt_p1
# lspv
hdisk0          00c6627ef0506e6e                    rootvg          active
hdisk1          00c6627efa5222ce                    destinationvg   active
# cfgmgr
# lspv
hdisk0          00c6627ef0506e6e                    rootvg          active
hdisk1          00c6627efa5222ce                    destinationvg   active
hdisk2          none                                None
```

- Verify that `reserve_policy` of `hdisk2` is set to `single_path` as shown in Example 4-9.

Example 4-9 Verifying that hdisk2 is set to single_path on partition 1

```
# lsattr -El hdisk2
PCM             PCM/friend/vscsi Path Control Module        False
algorithm       fail_over        Algorithm                  True
hcheck_cmd      test_unit_rdy    Health Check Command       True
hcheck_interval 0                Health Check Interval      True
hcheck_mode     nonactive        Health Check Mode          True
max_transfer    0x40000          Maximum TRANSFER Size      True
pvid            none             Physical volume identifier False
queue_depth     3                Queue DEPTH                True
reserve_policy  single_path      Reserve Policy             True
```

- From the IVM partition, configure `hdisk14` to be seen as `hdisk2` on `partition 2` as shown in Example 4-10.

Example 4-10 Configuring hdisk2 on partition 2

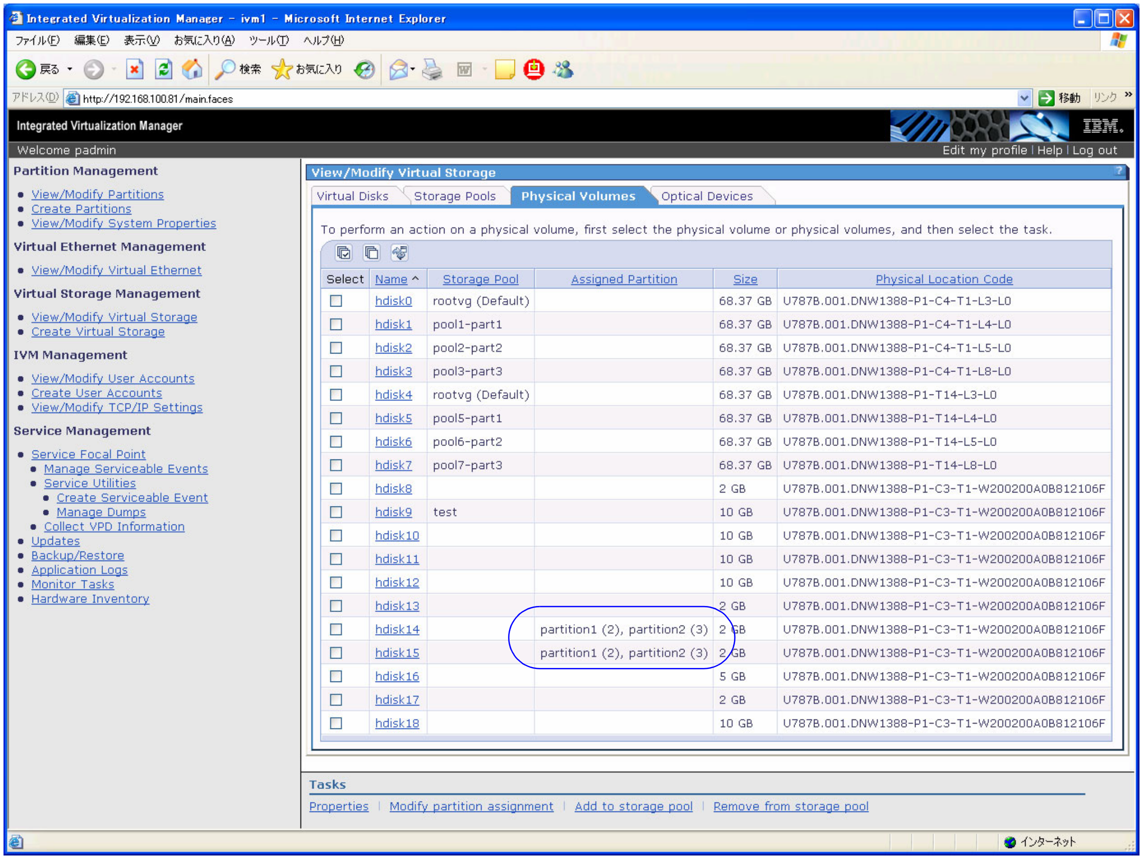
```
# uname -n
virt_p2
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
# cfgmgr
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
hdisk2          none                                None
```

- Verify that `reserve_policy` of `hdisk2` is set to `single_path` as shown in Example 4-11 on page 143.

Example 4-11 Verifying that hdisk2 is set to single_path on partition 2

```
# lsattr -El hdisk2
PCM             PCM/friend/vscsi Path Control Module        False
algorithm       fail_over        Algorithm                  True
hcheck_cmd      test_unit_rdy    Health Check Command       True
hcheck_interval 0                Health Check Interval      True
hcheck_mode     nonactive        Health Check Mode          True
max_transfer    0x40000          Maximum TRANSFER Size      True
pvid            none             Physical volume identifier False
queue_depth     3                Queue DEPTH                True
reserve_policy  single_path      Reserve Policy             True
```

- From the IVM partition, configure `hdisk15` to be seen as `hdisk3` on both `partition 1` and `partition 2`. Verify that `reserve_policy` of `hdisk3` is set to `single_path` on both partitions.
- Note that from IVM we configured `hdisk14` and `hdisk15` to map as the same name, `hdisk2` and `hdisk3` on both LPARs. While this is not mandatory from an HACMP viewpoint, it is good practice, especially with clusters containing large quantities of nodes or disks.
- Verify the assignment of both `hdisk14` and `hdisk15` using IVM GUI as shown in Figure 4-9 on page 144.


Integrated Virtualization Manager - ivm1 - Microsoft Internet Explorer
ファイル(F) 編集(E) 表示(V) お気に入り(A) ツール(T) ヘルプ(H)
戻る
検索
お気に入り
アドレス(D) http://192.168.100.81/main.faces
移動
リンク
Integrated Virtualization Manager
IBM.
Welcome padmin
Edit my profile | Help | Log out
Partition Management
View/Modify Partitions
Create Partitions
View/Modify System Properties
Virtual Ethernet Management
View/Modify Virtual Ethernet
Virtual Storage Management
View/Modify Virtual Storage
Create Virtual Storage
IVM Management
View/Modify User Accounts
Create User Accounts
View/Modify TCP/IP Settings
Service Management
Service Focal Point
Manage Serviceable Events
Service Utilities
Create Serviceable Event
Manage Dumps
Collect VPD Information
Updates
Backup/Restore
Application Logs
Monitor Tasks
Hardware Inventory
View/Modify Virtual Storage
Virtual Disks
Storage Pools
Physical Volumes
Optical Devices
Tasks
Properties | Modify partition assignment | Add to storage pool | Remove from storage pool
インターネット

To perform an action on a physical volume, first select the physical volume or physical volumes, and then select the task.

Select	Name ^	Storage Pool	Assigned Partition	Size	Physical Location Code
□	hdisk0	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L3-L0
□	hdisk1	pool1-part1		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L4-L0
□	hdisk2	pool2-part2		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L5-L0
□	hdisk3	pool3-part3		68.37 GB	U787B.001.DNW1388-P1-C4-T1-L8-L0
□	hdisk4	rootvg (Default)		68.37 GB	U787B.001.DNW1388-P1-T14-L3-L0
□	hdisk5	pool5-part1		68.37 GB	U787B.001.DNW1388-P1-T14-L4-L0
□	hdisk6	pool6-part2		68.37 GB	U787B.001.DNW1388-P1-T14-L5-L0
□	hdisk7	pool7-part3		68.37 GB	U787B.001.DNW1388-P1-T14-L8-L0
□	hdisk8			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk9	test		10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk10			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk11			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk12			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk13			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk14		partition1 (2), partition2 (3)	2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk15		partition1 (2), partition2 (3)	2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk16			5 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk17			2 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F
□	hdisk18			10 GB	U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F

Figure 4-9 Viewing hdisks assignment using IVM Web-based interface

- Verify the assignment of both `hdisk14` and `hdisk15` using IVM command-line interface as shown in Example 4-12.

Example 4-12 Viewing disk assignment using IVM command-line interface

```
$ lsmap -all
SVSA            Physloc                                        Client Partition ID
--------------- ---------------------------------------------- ------------------
vhost0          U9113.550.106627E-V1-C11                       0x00000002

VTD                   vtscsi0
LUN                   0x8100000000000000
Backing device        lv1_part1
Physloc

VTD                   vtscsi1
LUN                   0x8200000000000000
Backing device        lv2_part1
Physloc
```

```
VTD                   vtscsi6
LUN                   0x8300000000000000
Backing device        hdisk14
Physloc               U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L6000000000000

VTD                   vtscsi8
LUN                   0x8400000000000000
Backing device        hdisk15
Physloc               U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L7000000000000

SVSA            Physloc                                      Client Partition ID
--------------- -------------------------------------------- ------------------
vhost1          U9113.550.106627E-V1-C13                     0x00000003

VTD                   vtscsi2
LUN                   0x8100000000000000
Backing device        lv1_part2
Physloc

VTD                   vtscsi3
LUN                   0x8200000000000000
Backing device        lv2_part2
Physloc

VTD                   vtscsi7
LUN                   0x8300000000000000
Backing device        hdisk14
Physloc               U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L6000000000000

VTD                   vtscsi9
LUN                   0x8400000000000000
Backing device        hdisk15
Physloc               U787B.001.DNW1388-P1-C3-T1-W200300A0B812106F-L7000000000000

SVSA            Physloc                                      Client Partition ID
--------------- -------------------------------------------- ------------------
vhost2          U9113.550.106627E-V1-C15                     0x00000004

VTD                   vtscsi4
LUN                   0x8100000000000000
Backing device        lv1_part3
Physloc

VTD                   vtscsi5
LUN                   0x8200000000000000
Backing device        lv2_part3
```

- We made all these verifications because we used command line to define and manage hard disks at IVM layer. It is always good practise to verify storage configuration, before and after you define the new resources.
- Verify that physical volumes are accessible from AIX command line on both partitions using **lspv** command as shown in Example 4-13. Note that `hdisk2` and `hdisk3` are not assigned PVID yet.

Example 4-13 Verifying disk availability and PVIDs on partition 1

```
# uname -n
virt_p1
# lspv
hdisk0          00c6627ef0506e6e           rootvg          active
```

```
hdisk1            00c6627efa5222ce              destinationvg   active
hdisk2            none                          None
hdisk3            none                          None
```

- Verify the VG major numbers available using the **lvlstmajor** command or by using the following command: **ls -l /dev | grep vg**
- Force the creation of volume group vg1 containing `hdisk2` on `partition 1`. Force the creation of volume group vg2 containing `hdisk3` on `partition 1`. It is good practice that every volume group has the same major number across all cluster nodes, especially if you intend to use NFS. Note that `hdisk2` and `hdisk3` have been assigned a PVID and `vg1` and `vg2` have been varied on, as shown in Example 4-14.

Example 4-14 Creating volume groups vg1 and vg2

```
# uname -n
virt_p1
# lsvg
rootvg
destinationvg
# mkvg -y vg1 -V 34 hdisk2
0516-1254 mkvg: Changing the PVID in the ODM.
0516-1398 mkvg: The physical volume hdisk2, appears to belong to
another volume group. Use the force option to add this physical volume
to a volume group.
0516-862 mkvg: Unable to create volume group.
# mkvg -f -y vg1 -V 34 hdisk2
vg1
# mkvg -y vg2 -V35 hdisk3
0516-1254 mkvg: Changing the PVID in the ODM.
0516-1398 mkvg: The physical volume hdisk3, appears to belong to
another volume group. Use the force option to add this physical volume
to a volume group.
0516-862 mkvg: Unable to create volume group.
# mkvg -f -y vg2 -V35 hdisk3
vg2
# lsvg
rootvg
destinationvg
vg1
vg2
# lsvg -o
vg2
vg1
destinationvg
rootvg
# lspv
hdisk0          00c6627ef0506e6e                    rootvg          active
```

```
hdisk1          00c6627efa5222ce                    destinationvg   active
hdisk2          00c6627e142959de                    vg1             active
hdisk3          00c6627e142e634d                    vg2             active
```

- Create logical volume lv1 within volume group vg1.
- Create logical volume lv2 within volume group vg2.
- Create and mount filesystem `fs1` on logical volume `lv1`.
- Create and mount filesystem `fs2` on logical volume `lv2`.
- Create test files named `testfile1` within filesystem `fs1` and `testfile2` within filesystem `fs2`. The result is shown in Example 4-15.

Example 4-15   Creating filesystems and test files on partition 1

```
# lsvg -l vg1
vg1:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
lv1                 jfs2       10    10    1    open/syncd    /fs1
loglv01             jfs2log    1     1     1    open/syncd    N/A
# lsvg -l vg2
vg2:
LV NAME             TYPE       LPs   PPs   PVs  LV STATE      MOUNT POINT
lv2                 jfs2       10    10    1    open/syncd    /fs2
loglv02             jfs2log    1     1     1    open/syncd    N/A

# echo testfile_in_fs1 > /fs1/testfile1
# ls -l /fs1
total 8
drwxr-xr-x   2 root     system          256 Oct  4 11:44 lost+found
-rw-r--r--   1 root     system           16 Oct  4 11:48 testfile1
# echo testfile_in_fs2 > /fs2/testfile2
# ls -l /fs2
total 8
drwxr-xr-x   2 root     system          256 Oct  4 11:46 lost+found
-rw-r--r--   1 root     system           16 Oct  4 11:48 testfile2
```

- Unmount `fs1` and `fs2` and varyoff `vg1` and `vg2` on `partition 1`.
- Log on to `partition 2`.
- On partition 2, although the **`lspv`** command does not display any PVID, the PVIDs do exist on the disks as shown in Example 4-16.

Example 4-16   Displaying PVIDs and disk status on partition 2

```
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
hdisk2          none                                None
```

```
hdisk3          none                                None
# lqueryv -h /dev/hdisk2
```

```
hdisk3          none                                None
# lquerypv -h /dev/hdisk2
00000000   C9C2D4C1 00000000 00000000 00000000  |................|
00000010   00000000 00000000 00000000 00000000  |................|
00000020   00000000 00000000 00000000 00000000  |................|
00000030   00000000 00000000 00000000 00000000  |................|
00000040   00000000 00000000 00000000 00000000  |................|
00000050   00000000 00000000 00000000 00000000  |................|
00000060   00000000 00000000 00000000 00000000  |................|
00000070   00000000 00000000 00000000 00000000  |................|
00000080   00C6627E 142959DE 00000000 00000000  |..b~.)Y.........|
00000090   00000000 00000000 00000000 00000000  |................|
000000A0   00000000 00000000 00000000 00000000  |................|
000000B0   00000000 00000000 00000000 00000000  |................|
000000C0   00000000 00000000 00000000 00000000  |................|
000000D0   00000000 00000000 00000000 00000000  |................|
000000E0   00000000 00000000 00000000 00000000  |................|
000000F0   00000000 00000000 00000000 00000000  |................|
# lquerypv -h /dev/hdisk3
00000000   C9C2D4C1 00000000 00000000 00000000  |................|
00000010   00000000 00000000 00000000 00000000  |................|
00000020   00000000 00000000 00000000 00000000  |................|
00000030   00000000 00000000 00000000 00000000  |................|
00000040   00000000 00000000 00000000 00000000  |................|
00000050   00000000 00000000 00000000 00000000  |................|
00000060   00000000 00000000 00000000 00000000  |................|
00000070   00000000 00000000 00000000 00000000  |................|
00000080   00C6627E 142E634D 00000000 00000000  |..b~..cM........|
00000090   00000000 00000000 00000000 00000000  |................|
000000A0   00000000 00000000 00000000 00000000  |................|
000000B0   00000000 00000000 00000000 00000000  |................|
000000C0   00000000 00000000 00000000 00000000  |................|
000000D0   00000000 00000000 00000000 00000000  |................|
000000E0   00000000 00000000 00000000 00000000  |................|
000000F0   00000000 00000000 00000000 00000000  |................|
```

- Remove the devices hdisk2 and hdisk3 and run **cfgmgr** command. Note that PVIDs are now displayed properly as shown in Example 4-17.

Example 4-17 Listing the PVIDs after having removing hdisks and running cfgmgr

```
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
hdisk2          none                                None
hdisk3          none                                None
# rmdev -dl hdisk2
hdisk2 deleted
# rmdev -dl hdisk3
```

```
hdisk3 deleted
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
# cfgmgr
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
hdisk2          00c6627e142959de                    None
hdisk3          00c6627e142e634d                    None
```

- Import the volume groups on `partition 2` with the same VG major number, mount the filesystems and access data files as shown in Example 4-18. Make sure that when you import a volume group the name of its logical volumes, file systems and file systems logs do not conflict with the already existing ones and remain unchanged.

Example 4-18 Importing volume groups and accessing files

```
# uname -n
virt_p2
# ls -l /dev | grep vg
crw-rw----   1 root     system       10,  0 Sep 27 15:26 IPL_rootvg
crw-------   1 root     system       10,  0 Sep 27 14:28 __vg10
crw-rw----   1 root     system       10,  0 Sep 27 14:28 rootvg
# lsvg
rootvg
# importvg -y vg1 -V 34 hdisk2
vg1
# importvg -y vg2 -V 35 hdisk3
vg2
# lsvg
rootvg
vg1
vg2
# lsvg -o
vg2
vg1
rootvg
# lspv
hdisk0          00c6627ef04fc22d                    rootvg          active
hdisk1          00c6627ec9cf0f77                    rootvg          active
hdisk2          00c6627e142959de                    vg1             active
hdisk3          00c6627e142e634d                    vg2             active
# ls -l /dev | grep vg
crw-rw----   1 root     system       10,  0 Sep 27 15:26 IPL_rootvg
```

```
crw-------   1 root     system       10,  0 Sep 27 14:28 __vg10
crw-------   1 root     system       34,  0 Oct  4 11:56 __vg34
crw-------   1 root     system       35,  0 Oct  4 11:56 __vg35
crw-rw----   1 root     system       10,  0 Sep 27 14:28 rootvg
crw-rw----   1 root     system       34,  0 Oct  4 11:56 vg1
crw-rw----   1 root     system       35,  0 Oct  4 11:56 vg2
# mount /fs1
# mount /fs2
# df
Filesystem    512-blocks      Free %Used    Iused %Iused Mounted on
/dev/hd4          131072    100400   24%     2130    16% /
/dev/hd2         2490368     30840   99%    28736    81% /usr
/dev/hd9var       131072    114976   13%      351     3% /var
/dev/hd3          131072    130160    1%       22     1% /tmp
/dev/hd1          131072    130360    1%        5     1% /home
/proc                  -         -    -         -     -  /proc
/dev/hd10opt      262144     30688   89%     3702    51% /opt
/dev/lv1           81920     81232    1%        5     1% /fs1
/dev/lv2           81920     81232    1%        5     1% /fs2
# ls -l /fs1 /fs2
/fs1:
total 8
drwxr-xr-x   2 root     system          256 Oct  4 11:44 lost+found
-rw-r--r--   1 root     system           16 Oct  4 11:48 testfile1

/fs2:
total 8
drwxr-xr-x   2 root     system          256 Oct  4 11:46 lost+found
-rw-r--r--   1 root     system           16 Oct  4 11:48 testfile2
# cat /fs1/testfile1 /fs2/testfile2
testfile_in_fs1
testfile_in_fs2
```

- Change the volume group vg1 and `vg2` to be enhanced concurrent capable using **`chvg -C`** command; this is required because `hdisk2` and `hdisk3` are used for disk heartbeat networks.

4.1.7 Cluster testing

We recommend the following tests on the cluster:

- Individually test start and stop scripts associated to each application server.
- Test all application monitors that you use and verify if they produce the desired results.

- Verify start and stop cluster services.

It is a very good idea to monitor the status of services and processes that are required for HACMP to function properly. We added some aliases to the root's .profile and used them as shown in Example 4-19.

Example 4-19 Using aliases to monitor HACMP-related processes

```
[virt_p1][/]> cat /.profile|grep lsha
alias lsha='lssrc -a|egrep "svcs|ES|clvm"'
alias lshas='lssrc -ls clstrmgrES | grep "Current state"'

[virt_p1][/]> lsha
 clcomdES         clcomdES         344302       active
 clstrmgrES       cluster          364766       active
 topsvcs          topsvcs          163954       active
 grpsvcs          grpsvcs          249954       active
 emsvcs           emsvcs           217174       active
 emaixos          emsvcs           442404       active
 gsclvmd                           262166       active
 clinfoES         cluster          127136       active
 grpglsm          grpsvcs                       inoperative
[virt_p1][/]> lshas
Current state: ST_STABLE
```

- Bring resource groups online and verify if all resources are correctly acquired. Verify they comply with startup policy and that dependencies, where applicable are correctly handled.
- Bring resource groups offline and verify if all resources are correctly released.
- Move resource groups across cluster nodes and verify that cluster nodes release and acquire resources correctly. Verify that resource group movement complies with fallover policy.
- Carefully plan resource group fallback and failover policies. Consider how nodes leaving or joining the cluster affect resource group behavior. Also consider resource group dependencies.
- Carefully consider if you need to start HACMP services automatically on each cluster node. Consider the following scenario:

  One of the cluster nodes fails during the night. Its resource groups are correctly acquired by the remaining nodes. Next day, you restart the defective node, after resolving the problem. AIX boots, HACMP services are started automatically and the node rejoins the cluster. Depending on their policies, resource groups may fallback to the node that has just rejoined the cluster. This means unplanned downtime.

Ideally you should configure HACMP services startup and resource group policies, such that they do not cause unintended service downtime.

You can query resource group policies and status at any time using **clRGinfo** command as shown in Example 4-20.

Example 4-20 Viewing resource group status and policy using clRGinfo -v

```
virt_p1][/]> clRGinfo -v

Cluster Name: cluster1

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
Node                          State
---------------------------- ---------------
virt1_node1                   ONLINE
virt2_node2                   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
Node                          State
---------------------------- ---------------
virt2_node2                   ONLINE
virt1_node1                   OFFLINE
```

- Basic understanding of cluster events is extremely beneficial. While the logs are verbose, contrary to popular belief it can be relatively easy to observe high-level information. For example it is straightforward to spot information regarding node up and node down cluster events. A detailed description of all cluster logs can be found in HACMP manual available at:
  `http://publib.boulder.ibm.com/infocenter/clresctr/vxrx/index.jsp`
- In Example 4-21 on page 152 we show a few excerpts from /usr/es/adm/cluster.log.

Example 4-21 Excerpts from /usr/es/adm/cluster.log

```
Oct 10 11:51:16 virt_p1 user:notice HACMP for AIX: EVENT START: node_up
virt1_node1
Oct 10 11:52:02 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED: node_up
virt1_node1 0
```

```
Oct 10 12:50:05 virt_p1 user:notice HACMP for AIX: EVENT START:
node_up_remote_complete virt1_node1
Oct 10 12:50:05 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED:
node_up_remote_complete virt1_node1 0

Oct 10 11:52:03 virt_p1 user:notice HACMP for AIX: EVENT START: start_server
app_server1
Oct 10 11:52:03 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED:
start_server app_server1 0

Oct 10 12:49:54 virt_p1 user:notice HACMP for AIX: EVENT START: stop_server
app_server1
Oct 10 12:49:54 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED: stop_server
app_server1 0

Oct 10 12:53:31 virt_p1 user:notice HACMP for AIX: EVENT START:
acquire_service_addr virt_p2_serv

Oct 10 12:53:32 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED:
acquire_service_addr virt_p2_serv 0

Oct 10 12:49:57 virt_p1 user:notice HACMP for AIX: EVENT START:
release_service_addr virt_p1_serv
Oct 10 12:49:58 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED:
release_service_addr virt_p1_serv 0

Oct 10 15:46:24 virt_p1 user:notice HACMP for AIX: EVENT START: network_down
minus 1 net_diskhb_01
Oct 10 15:46:24 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED:
network_down minus 1 net_diskhb_01 0
```

- Test cluster resilience of failure of connections to external storage. In our scenario, we purposely removed one of the Fibre Channel cards.
- Test cluster resilience to the failure of its communication interfaces. In our scenario, the primary Ethernet adapter. The service should be available even in the case of adapter failure. In our case we took the following steps:
  - Unplugged the cable from the primary interface and tested the connection. The service was still available.
  - Plugged the cable back into the adapter. The EtherChannel interface recovered and returned to the main channel.

Errors generated in the AIX-level error log of IVM LPAR are shown in Example 4-22.

Example 4-22 Errors in case of primary Ethernet adapter failure

```
# errpt -a | more
LABEL:          ECH_CHAN_RCVRY
IDENTIFIER:     8650BE3F

Date/Time:       Tue Oct 10 12:33:09 EDT 2006
Sequence Number: 341
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            INFO
Resource Name:   ent7
Resource Class:  adapter
Resource Type:   ibm_ech
Location:

Description
ETHERCHANNEL RECOVERY

Probable Causes
CABLE
SWITCH
ADAPTER

Failure Causes
CABLES AND CONNECTIONS

        Recommended Actions
        CHECK CABLE AND ITS CONNECTIONS
        IF ERROR PERSISTS, REPLACE ADAPTER CARD.

Detail Data
A primary adapter in the EtherChannel recovered: returning to main channel
---------------------------------------------------------------------------
LABEL:          GOENT_RCVRY_EXIT
IDENTIFIER:     F3931284

Date/Time:       Tue Oct 10 12:33:08 EDT 2006
Sequence Number: 340
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            INFO
Resource Name:   ent2
Resource Class:  adapter
Resource Type:   14106902
Location:        U787B.001.DNW1388-P1-C1-T1
VPD:
        Product Specific.(  ).......10/100/1000 Base-TX PCI-X Adapter
        Part Number.................00P6130
        FRU Number..................00P6130
        EC Level....................H12818
        Manufacture ID..............YL1021
        Network Address.............000255D3E152
        ROM Level.(alterable).......GOL021

Description
ETHERNET NETWORK RECOVERY MODE

        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES

Detail Data
FILE NAME
line: 204 file: goent_intr.c
PCI ETHERNET STATISTICS
0000 158C 0063 08D3 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 2400 0000 0000 000A BF37 0000 0000 0000 1DE4 0000 094F 0000 1157
0000 0000 0000 0C3C 0000 0000 0003 C042 0000 0000 0000 0000 0000 0006 0000 048D
0000 0000 0000 0000 0000 0000 0000 0002 0000 0000 0000 0008 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 BB43 00F0 1969 0C00 0000 0000 01A0 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000
DEVICE DRIVER INTERNAL STATE
5555 5555 0000 0000 0000 0000
SOURCE ADDRESS
0002 55D3 E152
---------------------------------------------------------------------------
LABEL:          ECH_CHAN_FAIL
IDENTIFIER:     F3846E13

Date/Time:       Tue Oct 10 12:26:36 EDT 2006
Sequence Number: 338
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            PERM
Resource Name:   ent7
Resource Class:  adapter
Resource Type:   ibm_ech
Location:

Description
ETHERCHANNEL FAILOVER

Probable Causes
CABLE
SWITCH
ADAPTER

Failure Causes
CABLES AND CONNECTIONS

        Recommended Actions
        CHECK CABLE AND ITS CONNECTIONS
        IF ERROR PERSISTS, REPLACE ADAPTER CARD.

Detail Data
All primary EtherChannel adapters failed: switching over to backup adapter
---------------------------------------------------------------------------
LABEL:          GOENT_LINK_DOWN
IDENTIFIER:     EC0BCCD4

Date/Time:       Tue Oct 10 12:26:36 EDT 2006
Sequence Number: 337
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            TEMP
Resource Name:   ent2
Resource Class:  adapter
Resource Type:   14106902
Location:        U787B.001.DNW1388-P1-C1-T1
VPD:
        Product Specific.(  ).......10/100/1000 Base-TX PCI-X Adapter
        Part Number.................00P6130
        FRU Number..................00P6130
        EC Level....................H12818
        Manufacture ID..............YL1021
        Network Address.............000255D3E152
        ROM Level.(alterable).......GOL021

Description
ETHERNET DOWN
```

```
        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES

Detail Data
FILE NAME
line: 149 file: goent_intr.c
PCI ETHERNET STATISTICS
0000 1404 0063 089B 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 2400 0000 0000 000A BF37 0000 0000 0000 1DE4 0000 092B 0000 1120
0000 0000 0000 0C3C 0000 0000 0003 C042 0000 0000 0000 0000 0000 0006 0000 048D
0000 0000 0000 0000 0000 0000 0000 0002 0000 0000 0000 0008 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 BB41 00F0 1969 0C00 0000 0000 01A0 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000
DEVICE DRIVER INTERNAL STATE
2222 2222 0000 0000 0000 0000
SOURCE ADDRESS
0002 55D3 E152
---------------------------------------------------------------------------
LABEL:          SRC_RSTRT
IDENTIFIER:     BA431EB7

Date/Time:       Tue Oct 10 12:16:16 EDT 2006
Sequence Number: 335
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           S
Type:            PERM
Resource Name:   SRC

Description
SOFTWARE PROGRAM ERROR

Probable Causes
APPLICATION PROGRAM

Failure Causes
SOFTWARE PROGRAM

        Recommended Actions
        VERIFY SUBSYSTEM RESTARTED AUTOMATICALLY

Detail Data
SYMPTOM CODE
```

```
256
SOFTWARE ERROR CODE
-9035
        Recommended Actions
        CHECK CABLE AND ITS CONNECTIONS
        IF ERROR PERSISTS, REPLACE ADAPTER CARD.

Detail Data
All primary EtherChannel adapters failed: switching over to backup adapter

---------------------------------------------------------------------------
LABEL:          GOENT_LINK_DOWN
IDENTIFIER:     EC0BCCD4

Date/Time:       Tue Oct 10 12:26:36 EDT 2006
Sequence Number: 337
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            TEMP
Resource Name:   ent2
Resource Class:  adapter
Resource Type:   14106902
Location:        U787B.001.DNW1388-P1-C1-T1
VPD:
        Product Specific.(  ).......10/100/1000 Base-TX PCI-X Adapter
        Part Number.................00P6130
        FRU Number..................00P6130
        EC Level....................H12818
        Manufacture ID..............YL1021
        Network Address.............000255D3E152
        ROM Level.(alterable).......GOL021

Description
ETHERNET DOWN

        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES

Detail Data
FILE NAME
line: 149 file: goent_intr.c
PCI ETHERNET STATISTICS
0000 1404 0063 089B 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 2400 0000 0000 000A BF37 0000 0000 0000 1DE4 0000 092B 0000 1120
```

```
0000 0000 0000 0C3C 0000 0000 0003 C042 0000 0000 0000 0000 0000 0006 0000 048D
0000 0000 0000 0000 0000 0000 0000 0002 0000 0000 0000 0008 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 BB41 00F0 1969 0C00 0000 0000 01A0 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000
DEVICE DRIVER INTERNAL STATE
2222 2222 0000 0000 0000 0000
SOURCE ADDRESS
0002 55D3 E152
```

- Unplugged the cable from the secondary interface and tested the connection. The service was not affected.
- Plugged the cable back into the adapter and verify that secondary connection was available and redundancy restored.

Errors generated in the error log are shown in Example 4-23.

Example 4-23 Errors in case of backup Ethernet adapter failure

```
# errpt -a | more
---------------------------------------------------------------------------
LABEL:          GOENT_RCVRY_EXIT
IDENTIFIER:     F3931284

Date/Time:       Tue Oct 10 12:38:18 EDT 2006
Sequence Number: 344
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            INFO
Resource Name:   ent1
Resource Class:  adapter
Resource Type:   14108902
Location:        U787B.001.DNW1388-P1-T10
VPD:
        Product Specific.(  ).......2-Port 10/100/1000 Base-TX PCI-X
                                    Adapter
        Network Address.............0002552FC827
        ROM Level.(alterable).......DV0210

Description
ETHERNET NETWORK RECOVERY MODE

        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES
```

```
Detail Data
FILE NAME
line: 204 file: goent_intr.c
PCI ETHERNET STATISTICS
0000 16C1 0063 08D3 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 2529 0000 0000 000A 5277 0000 0000 0000 1F39 0000 0A48 0000 175F
0000 0000 0000 013F 0000 0000 0000 6FA3 0000 0000 0000 0000 0000 0000 0000 0002
0000 0000 0000 0000 0000 0000 0000 0002 0000 0000 0000 0002 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 BB47 003C 1969 0C00 0000 0000 01A0 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000
DEVICE DRIVER INTERNAL STATE
5555 5555 0000 0000 0000 0000
SOURCE ADDRESS
0002 55D3 E152
---------------------------------------------------------------------------
LABEL:          GOENT_LINK_DOWN
IDENTIFIER:     EC0BCCD4

Date/Time:       Tue Oct 10 12:37:13 EDT 2006
Sequence Number: 343
Machine Id:      00C6627E4C00
Node Id:         ivm1
Class:           H
Type:            TEMP
Resource Name:   ent1
Resource Class:  adapter
Resource Type:   14108902
Location:        U787B.001.DNW1388-P1-T10
VPD:
        Product Specific.(  ).......2-Port 10/100/1000 Base-TX PCI-X
                                    Adapter
        Network Address.............0002552FC827
        ROM Level.(alterable).......DV0210

Description
ETHERNET DOWN

        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES

Detail Data
FILE NAME
line: 149 file: goent_intr.c
```

```
PCI ETHERNET STATISTICS
0000 1681 0063 089B 0000 0001 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 2529 0000 0000 000A 5277 0000 0000 0000 1F39 0000 0A48 0000 175F
0000 0000 0000 013F 0000 0000 0000 6FA3 0000 0000 0000 0000 0000 0000 0000 0002
0000 0000 0000 0000 0000 0000 0000 0002 0000 0000 0000 0002 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 BB45 003C 1969 0C00 0000 0000 01A0 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000
DEVICE DRIVER INTERNAL STATE
2222 2222 0000 0000 0000 0000
SOURCE ADDRESS
0002 55D3 E152
```

Cluster single points of failure

- IBM System p5 p550 system; an outage of the physical box.
- VIOS LPAR; any outage to the VIOS will affect virtual device availability.
- The switch to which Ethernet cables are connected.

Advantages of our cluster scenario

- You can increase the number of nodes at any time by adding more LPARs and use the system as a testing environment for HACMP.
- You can implement a very complex cluster with a large number of nodes, a lot of resource groups with different policies and complicated dependencies or application servers interactions.
- You can simulate the failure of one or more Single Points Of Failure (SPOFs) either sequential or simultaneously and get a realistic view of the degree of availability achieved.
- Causes of failure for cluster nodes may be classified according to the layer at which they occur. One approach is to divide them in three categories:
  - Failure of a hardware component that renders the system unusable. Hardware components of IBM System p systems are very reliable.
  - Crash of the operating system. AIX is a strong and robust operating system.
  - Failures occurred at the application layer causing a crash of the whole node. This reason accounts for most of the crashes of cluster nodes.

From this perspective, there is benefit for us to draw a comparison between clusters containing distributed physical nodes and cluster having all nodes located within a single physical system as shown in Table 4-3.

Table 4-3 Comparing different type of clusters

Feature	All nodes within the same system	All nodes distributed across different systems
Resilient to hardware failures	no	yes
Resilient to operating system failures	yes	yes
Resilient to application failures	yes	yes

Once we accept the physical system as a single point of failure, we do not need to buy a second system. Moreover, there is no need to buy an HMC as we can use IVM instead. Where given risks can be accepted, this cluster implementation proves cost-effective:

- The amount of memory and CPU resources allocated to each partition can be adjusted according to application requirements.
- The cluster is resilient to failure of one Fibre Channel adapter.
- The cluster is resilient to failure of one physical network adapter.

4.1.8 DLPAR managed by IVM

One of the new features for IVM Version 1.3.0 is the support of DLPAR operations on client LPARs for memory and processors. IVM DLPAR operations can be performed from both the Web GUI and command line interfaces.

Before executing DLPAR operations, you should check the status of your system. You can check following settings of the partitions by using the **`lssyscfg`** command, as shown in Example 4-24 on page 163.

- The state of the RMC connection between IVM and the client partition.
- The IP address used by RMC.
- Whether the partition supports memory DLPAR.
- Whether the partition supports processor DLPAR.

If you need further information, refer to **`lssyscfg`** command reference in Hardware Information Center:

`http://publib.boulder.ibm.com/infocenter/eserver/v1r3s/index.jsp`

Example 4-24 lssyscfg -r lpar

```
$ lssyscfg -r lpar -F
"name,state,rmc_state,rmc_ipaddr,dlpar_mem_capable,dlpar_proc_capable"
--header
name,state,rmc_state,rmc_ipaddr,dlpar_mem_capable,dlpar_proc_capable
10-6627E,Running,active,192.168.100.81,1,1
partition1,Running,active,192.168.100.73,1,1
partition2,Running,active,192.168.100.74,1,1
partition3,Running,active,192.168.100.75,1,1
```

You can also check the following settings of an individual managed node, as shown in Example 4-25.

- Whether the changes of memory take effect immediately on the managed system or not.
- The IP address through which client partitions will communicate with the management partition.
- Whether the lpar_comm_ipaddr is using the default IP address or the IP address the user has set.

Example 4-25 lssyscfg -r sys

```
$ lssyscfg -r sys -F
"name,state,dlpar_mem_capable,lpar_comm_ipaddr,lpar_comm_default"
Server-9113-550-SN106627E,Operating,1,192.168.100.81,1
```

You can check the hardware resources of a managed system by GUI or the **lshwres** command. Output of which is shown in Example 4-26.

Example 4-26 Checking hardware resources by CUI

```
$ lshwres -r mem --level sys
configurable_sys_mem=16384,curr_avail_sys_mem=7680,pend_avail_sys_mem=7680,inst
alled_sys_mem=16384,deconfig_sys_mem=0,sys_firmware_mem=512,mem_region_size=64
$ lshwres -r mem --level lpar
lpar_name=10-6627E,lpar_id=1,curr_min_mem=512,curr_mem=2048,curr_max_mem=2048,p
end_min_mem=512,pend_mem=2048,pend_max_mem=2048,run_min_mem=512,run_mem=2048
lpar_name=partition1,lpar_id=2,curr_min_mem=128,curr_mem=2048,curr_max_mem=2048
,pend_min_mem=128,pend_mem=2048,pend_max_mem=2048,run_min_mem=128,run_mem=2048
lpar_name=partition2,...
...
$ lshwres -r proc --level sys
configurable_sys_proc_units=4.00,curr_avail_sys_proc_units=3.30,pend_avail_sys_
proc_units=3.30,installed_sys_proc_units=4.00,deconfig_sys_proc_units=0.00,min_
proc_units_per_virtual_proc=0.10,max_shared_proc_pools=1
$ lshwres -r proc --level lpar
```

```
lpar_name=10-6627E,lpar_id=1,curr_shared_proc_pool_id=0,curr_proc_mode=shared,c
urr_min_proc_units=0.10,curr_proc_units=0.40,curr_max_proc_units=4.00,curr_min_
procs=1,curr_procs=4,curr_max_procs=4,curr_sharing_mode=uncap,curr_uncap_weight
=128,pend_shared_proc_pool_id=0,pend_proc_mode=shared,pend_min_proc_units=0.10,
pend_proc_units=0.40,pend_max_proc_units=4.00,pend_min_procs=1,pend_procs=4,pen
d_max_procs=4,pend_sharing_mode=uncap,pend_uncap_weight=128,run_proc_units=0.40
,run_procs=4,run_uncap_weight=128
lpar_name=partition1,...
...
$ lshwres -r proc --level pool
shared_proc_pool_id=0,configurable_pool_proc_units=4.00,curr_avail_pool_proc_un
its=3.30,pend_avail_pool_proc_units=3.30
```

You can also check hardware resources in client partition, for example, by using **lparstat** command as shown in Example 4-27.

Example 4-27 Checking hardware resources in client partition

```
> lparstat -i
Node Name                                  : virt_p1
Partition Name                             : partition1
Partition Number                           : 2
Type                                       : Shared-SMT
Mode                                       : Uncapped
Entitled Capacity                          : 0.10
Partition Group-ID                         : 32770
Shared Pool ID                             : 0
Online Virtual CPUs                        : 1
Maximum Virtual CPUs                       : 4
Minimum Virtual CPUs                       : 1
Online Memory                              : 1792 MB
Maximum Memory                             : 2048 MB
Minimum Memory                             : 128 MB
Variable Capacity Weight                   : 128
Minimum Capacity                           : 0.10
Maximum Capacity                           : 4.00
Capacity Increment                         : 0.01
Maximum Physical CPUs in system            : 4
Active Physical CPUs in system             : 4
Active CPUs in Pool                        : 4
Unallocated Capacity                       : 0.00
Physical CPU Percentage                    : 10.00%
Unallocated Weight                         : 0
```

DLPAR operations can be performed from the command line using the **chsyscfg** IVM command. You can change parameters shown in Table 4-4.

Table 4-4 chsyscfg parameters for DLPAR

parameters	description
desired_mem	assigned memory in megabytes
desired_procs	assigned processors. In shared processing mode, this refers to virtual processors.
desired_proc_units	assigned shared processing units
sharing_mode	keep_idle_procs: Valid with dedicated processor mode share_idle_procs: Valid with dedicated processor mode cap: Capped mode. Valid with shared processor mode uncap: Uncapped mode. Valid with shared processor mode
uncap_weight	A weighted average of processing priority when in uncapped sharing mode. The smaller the value, the lower the weight. Valid values are: 0 - 255

When you execute the **chsyscfg** command, you can indicate the target value using ‘=’, but you can also indicate a value to increase or decrease by using ‘+=’ or ‘-=’. For example, to increase the amount of memory with 256 MB, execute **chsyscfg** command with `desired_mem` parameter in IVM partition as shown in Example 4-28.

Example 4-28 DLPAR operation for increasing memory

```
$ chsyscfg -r prof -i "lpar_name=partition1,desired_mem+=256"
```

To action an DLPAR operation to set the processing units to 0.5, execute the **chsyscfg** IVM command with `desired_proc_units` parameter as shown in Example 4-29.

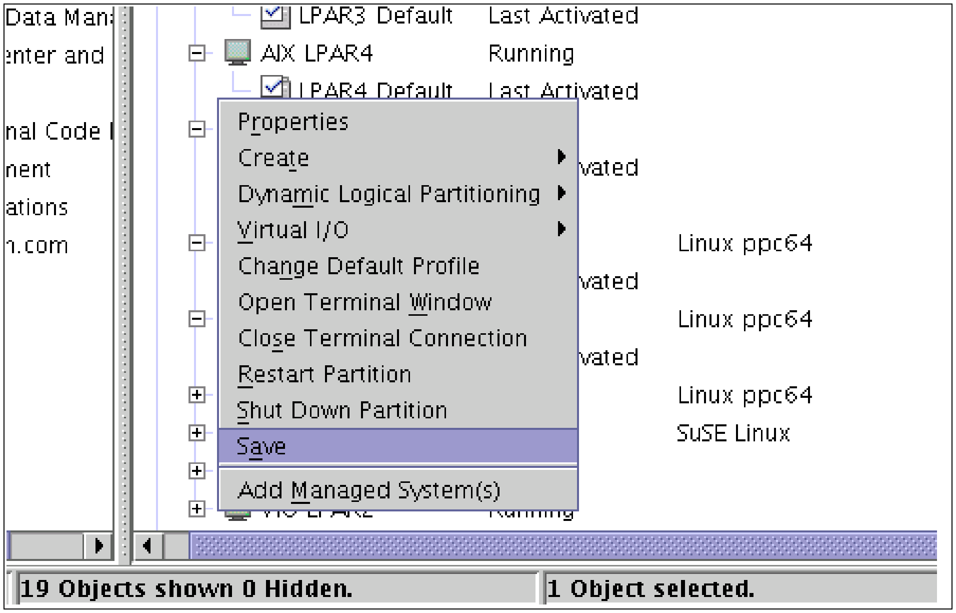
Example 4-29 DLPAR operation of setting processing units

```
$ chsyscfg -r prof -i "lpar_name=partition1,desired_proc_units=0.5"
```

> **Note:** Do not set sharing_mode and uncap_weight parameters in the same command; this may provide unexpected results. It is recommended to action such a reconfiguration in two separate steps. Setting sharing_mode to ‘cap’ results in uncap_weight being set to ‘0’, because POWER Hypervisor™ requires the weight to be ‘0’ before setting the sharing mode to capped. An Uncapped weight of ‘0’ has the same effect as Capped.

With regard to DLPAR, we found one significant difference between IVM-managed and HMC-managed systems. As previously mentioned, IVM only supports one profile for a given LPAR. Performing a DLPAR operation to add or remove resource actually commits the changes to that profile. Whereas a similar operation on an HMC-managed LPAR will not update the LPAR profile; it just makes the resource changes to the active instance. If for example, you added two extra CPUs via DLPAR to an HMC-managed LPAR, a logical LPAR shutdown and activation would revert the configuration back to that of the LPAR profile. that is, it would not be re-allocated the two additional CPUs.

On an IBM System p5 HMC, there is a “Save” option within WebSM; found on the context menu accessed from right-clicking the LPAR. Figure 4-10 illustrates this menu option. Selecting this option will prompt you for a new LPAR profile name. The current LPAR configuration will be saved into that profile name.



19 Objects shown 0 Hidden.

1 Object selected.

Figure 4-10 Save LPAR menu option

The same result can be achieved by using the **mksyscfg** HMC command with the **-o** parameter.

Note: Unfortunately neither the “Save” option nor the additional **mksyscfg** parameter are available on IBM System p4 HMCs.

4.1.9 Using HACMP to migrate resources between LPARs

Since our system is not managed by an HMC we need to find another way to dynamically assign or move resources from one partition to another. In our scenarios, we further customized cluster behavior to automatically allocate CPU and memory resources to logical partitions depending on cluster-specific events.

We had to configure client LPARs to be able to communicate with the VIOS LPAR without requiring manual password authentication every time a command is executed. We also needed this communication channel to be secure in order to prevent unwanted interferences with LPAR operations.

We installed and configured SSH on the client LPARs. Then we generated public and private keys; exchanging the public keys between VIOS and LPARs.

> **Note:** Either installing or upgrading to VIOS Version 1.3.0 will install and configure SSH/SSL on your VIOS LPAR. There is no need to install it manually. If you previously installed SSH/SSL then the upgrade to Version 1.3.0 will update your SSH installation to the supported configuration.

Scenario 1

First of all we considered the case in which equal resources are required by each LPAR when a resource group is acquired and application server is started. Therefore the same amount of CPU and memory are required on each LPAR for each resource group it acquires. We thought of this scenario as “partition-centric.”

For instance, two resource groups each containing a small scale applications; each would only require 256 MB of memory to run at an acceptable level of performance.

We created two scripts which are called every time a resource group is activated on a given LPAR. We called these scripts `dec-mem.sh` and `inc-mem.sh` and saved them in the local root directory as shown in Example 4-30.

Example 4-30 Scripts used to add or subtract memory on partition 1

```
root@virt_p1:/# cat dec-mem.sh
#!/bin/ksh
ssh -l padmin ivm1 'echo ". ./.profile;chsyscfg -r prof -i
lpar_name='$PARTNAME',desired_mem-=256"|ksh -s'

root@virt_p1:/# more inc-mem.sh
#!/bin/ksh
ssh -l padmin ivm1 'echo ". ./.profile;chsyscfg -r prof -i
lpar_name='$PARTNAME',desired_proc_units+=256"|ksh -s'
```

We have done the same thing for the second partition. As you can see in the example, the script contains the name of the partition.

Then we created two customized cluster events as shown in Example 4-31. One event is used to increase the amount of memory, the other is used to decrease the amount of memory. The definition of these events is consistent across all cluster nodes. These customized events call the scripts we created. Note that scripts are local to every node, they have the same name and are located in the same location. But you actually can change the content of the script if you decide to modify the amount of memory added on each partition or you decide to add CPU resources. You have to keep in mind that these scripts are *local* and you can implement any particular customization specific to the local system.

Example 4-31 Creating customized cluster events

```
Add a Custom Cluster Event

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

                                                        [Entry Fields]
* Cluster Event Name                                 [dec-mem]
* Cluster Event Description                          [decrease memory
256MB]
* Cluster Event Script Filename                      [/dec-mem.sh]



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

-----------------------------------------------------------------------
-----
                           Add a Custom Cluster Event

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

                                                        [Entry Fields]
* Cluster Event Name                                 [inc-mem]
* Cluster Event Description                          [increase memory
256MB]
* Cluster Event Script Filename                      [/inc-mem.sh]



F1=Help             F2=Refresh          F3=Cancel           F4=List
```

```
F5=Reset              F6=Command            F7=Edit               F8=Image
F9=Shell              F10=Exit              Enter=Do
```

Then we defined the `inc-mem` as a pre-event script of the `start_server` predefined cluster event. This way, the amount of 256 MB memory always is allocated to the partition just before the application server is started. So every time the partition acquires a resource group that contains an application server it will get 256 MB of additional memory before starting the application server. This is based on the assumption that the system has that amount of memory un-allocated.

We defined the `dec-mem` as a post-event script of the `stop_server` predefined cluster event. This way, the amount of 256 MB memory always is deallocated from the partition after the application server is stopped. So every time the partition releases a resource group that contains an application server it also releases 256 MB of its running memory after the stop application server completes. This memory block of 256 MB returns to the memory pool that has not been allocated.

Definition of the pre-event and post-event commands are shown in Example 4-32.

Example 4-32 Defining pre-event and post-event commands

```
Change/Show Cluster Events

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

                                                        [Entry Fields]

  Event Name                                          start_server

  Description                                         Script run to start a>

* Event Command                                      [/usr/es/sbin/cluster/>

  Notify Command                                     []
  Pre-event Command                                  [inc-mem]              +
  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



                           Change/Show Cluster Events

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

                                                        [Entry Fields]

  Event Name                                          stop_server

  Description                                         Script run to stop ap>

* Event Command                                      [/usr/es/sbin/cluster/>

  Notify Command                                     []
  Pre-event Command                                  []                       +
  Post-event Command                                 [dec-mem]                +
  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
```

When we started and stopped the cluster the following entries were generated in /usr/es/adm/cluster.log as shown in Example 4-33 on page 170.

Example 4-33   Excerpts from /usr/es/adm/cluster.log

```
Oct 17 12:02:59 virt_p1 user:notice HACMP for AIX: EVENT START: start_server
app_server1
Oct 17 12:02:59 virt_p1 user:notice HACMP for AIX: PRE EVENT COMMAND: inc-mem
app_server1
Oct 17 12:03:00 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED: inc-mem
app_server1 0
Oct 17 12:03:00 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED:
start_server app_server1 0
.......................................lines omitted ................
Oct 17 12:09:21 virt_p1 user:notice HACMP for AIX: EVENT START: stop_server
app_server1
Oct 17 12:09:21 virt_p1 user:notice HACMP for AIX: POST EVENT COMMAND: dec-mem
app_server1
Oct 17 12:09:22 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED: dec-mem
app_server1 0
```

```
Oct 17 12:09:22 virt_p1 user:notice HACMP for AIX: EVENT COMPLETED: stop_server
app_server1 0
```

Scenario 2

The second scenario we imagined was to modify the behavior of logical partitions to be “resource-group” centric.

For instance two resource groups that require different amount of memory or CPU power in order to run an application at an acceptable level of performance. We imagined that resource group `rg1` needs 0.1 extra CPU units and resource group `rg2` needs 0.2 extra CPU units and modified the start and stop application servers accordingly.

We modified the start application server script for resource group `rg1` to contain.

```
. /hacmp.env
```

(at the beginning of the script)

And:

```
ssh -l padmin ivm1 'echo ". ./.profile;chsyscfg -r prof -i
lpar_name='$PARTNAME',desired_proc_units+=0.1"|ksh -s'
```

(as the first line of the script)

We ensured that the partition receives the necessary CPU resources before it starts the application server. We made the assumption there are still enough CPU resources un-allocated on the system.

We modified the stop application server script of resource group `rg1` to contain

```
. /hacmp.env
```

(at the beginning of the script)

And:

```
ssh -l padmin ivm1 'echo ". ./.profile;chsyscfg -r prof -i
lpar_name='$PARTNAME',desired_proc_units-=0.1"|ksh -s'
```

(as the last line of the script)

We ensured that the partition releases the additional CPU resources that were required for the application server to run properly. The released resources would return to the pool of unassigned resources.

Because when configuring HACMP with CUoD or DLPAR, the LPAR names (as defined on HMC) must match the HACMP node names and the AIX hostnames, we had to find a work-around for this issue.

The start and stop application servers are associated to the resource group. The scripts are executed each time the resource group is brought online or offline, on every cluster node that acquires or releases the resource group. So the scripts need to have a unique form that would allow them to run on every cluster node.

Therefore, we decided to use a variable that points to the partition name. We defined on each system a profile file that defines and exports a variable that contains the name of the partition. The name of the variable is identical on all cluster nodes. However, its value is dependent on the name of the partition.

An example of this file shown in Example 4-34.

Example 4-34 Profile file on partition 1

```
root@virt_p1:/# more /hacmp.env
export PARTNAME=partition1
```

We verified how our partitions acquired and released CPU and memory using the following scenario:

- In their initial state of both partitions have 0.50 CPU and 1 GB of RAM as shown in Example 4-35.

Example 4-35 Partitions initial status

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.50
Online Memory                              : 1024 MB

root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
Entitled Capacity                          : 0.50
Online Memory                              : 1024 MB
```

- Start HACMP services on both partitions. Each partition acquires its resource group as shown in Example 4-36.

Example 4-36 Partition resources after starting HACMP services

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.60
Online Memory                              : 1280 MB
```

```
root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
Entitled Capacity                          : 0.70
Online Memory                              : 1280 MB
```

- Migrate resource group `rg1` to second partition. The results are shown in Example 4-37.

Example 4-37 Resources on each partition after migrating first resource group on the second node

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.50
Online Memory                              : 1024 MB

root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
Entitled Capacity                          : 0.80
Online Memory                              : 1536 MB
```

- Move resource group rg1 to back to first partition. The results are shown in Example 4-38 on page 173.

Example 4-38 Resources on each partition after moving back first resource group on the first node

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.60
Online Memory                              : 1280 MB

root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
Entitled Capacity                          : 0.70
Online Memory                              : 1280 MB
```

- Migrate resource group `rg2` to first partition. The results are shown in Example 4-39.

Example 4-39 Resources on each partition after migrating second resource group on the first node

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.80
Online Memory                              : 1536 MB

root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
Entitled Capacity                          : 0.50
Online Memory                              : 1024 MB
```

- Move resource group `rg2` to back to the second partition. The results are shown in Example 4-40.

Example 4-40 Resources on each partition after moving back the second resource group on the second node

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.60
Online Memory                              : 1280 MB

root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
Entitled Capacity                          : 0.70
Online Memory                              : 1280 MB
```

- Stop cluster services on both partitions. The results are shown in Example 4-41.

Example 4-41 Resource status on both partitions after stopping HACMP services

```
root@virt_p1:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition1
Entitled Capacity                          : 0.50
Online Memory                              : 1024 MB

root@virt_p2:/# lparstat -i|egrep “Partition Name|Entitled
Capacity|Online Memory”
Partition Name                             : partition2
```

```
Entitled Capacity                          : 0.50
Online Memory                              : 1024 MB
```

4.2 IBM System p5 scenario using two CECs

In the previous section we illustrated effective ways to use a single CEC with elements of Virtualization and HACMP, providing a trade off between application availability and overall cost.

In this section we build on the virtualization concepts and apply them to a traditional HACMP cluster; where cluster nodes are defined on LPARs located on different physical systems. Both cluster nodes use VIOS for both storage and network resources.

We are particularly interested to evaluate how cluster availability is influenced by VIOS. We also will evaluate the behavior of the Shared Ethernet Adapter (SEA) in an HACMP environment.

Our first cluster node relies on the services provided by a single VIOS. The VIO server itself therefore a single point of failure.

The second cluster node uses the services of two VIO servers to eliminate a VIOS as a single point of failure.

The system topology, showing the arrangement of the two CECs, is illustrated in Figure 4-11 on page 180.

4.2.1 Configuring virtualized storage for LPARs

The first node is defined on an LPAR named `partition3` hosted in an IVM-managed IBM System p5 p550 system. This LPAR has an AIX hostname of `virt_p3`. This system is connected to external storage via two Fibre Channel Adapters. Each adapter is connected to a different SAN switch. No physical storage adapters are allocated to `partition3`; it therefore uses external storage virtualized from the VIOS LPAR. This LPAR also provides IVM management for the physical system.

On our IVM LPAR we defined two hard disks named `hdisk11` and `hdisk12` and changed the reserve policy as shown in Example 4-42.

Example 4-42   Disk definition on IVM partition

```
$ lsmap -all
...
```

```
SVSA            Physloc                                      Client Partition ID
--------------- -------------------------------------------- ------------------
vhost2          U9113.550.106627E-V1-C15                     0x00000004


VTD                   vtscsi10
LUN                   0x8300000000000000
Backing device        hdisk11
Physloc               U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F-L3000000000000

VTD                   vtscsi11
LUN                   0x8400000000000000
Backing device        hdisk12
Physloc               U787B.001.DNW1388-P1-C3-T1-W200200A0B812106F-L4000000000000

$ lsdev -dev hdisk11 -attr
attribute      value                            description                           user_settable

PR_key_value   none                             Persistant Reserve Key Value          True
cache_method   fast_write                       Write Caching method                  False
ieee_volname   600A0B800012106E0000004D451B9589 IEEE Unique volume name               False
lun_id         0x0003000000000000               Logical Unit Number                   False
max_transfer   0x100000                         Maximum TRANSFER Size                 True
prefetch_mult  1                                Multiple of blocks to prefetch on read False
pvid           none                             Physical volume identifier            False
q_type         simple                           Queuing Type                          False
queue_depth    10                               Queue Depth                           True
raid_level     5                                RAID Level                            False
reassign_to    120                              Reassign Timeout value                True
reserve_policy no_reserve                       Reserve Policy                        True
rw_timeout     30                               Read/Write Timeout value              True
scsi_id        0x651d00                         SCSI ID                               False
size           10240                            Size in Mbytes                        False
write_cache    yes                              Write Caching enabled                 False
$ lsdev -dev hdisk12 -attr
attribute      value                            description                           user_settable

PR_key_value   none                             Persistant Reserve Key Value          True
cache_method   fast_write                       Write Caching method                  False
ieee_volname   600A0B8000120F330000005F451B94A4 IEEE Unique volume name               False
lun_id         0x0004000000000000               Logical Unit Number                   False
max_transfer   0x100000                         Maximum TRANSFER Size                 True
prefetch_mult  1                                Multiple of blocks to prefetch on read False
pvid           none                             Physical volume identifier            False
q_type         simple                           Queuing Type                          False
queue_depth    10                               Queue Depth                           True
raid_level     5                                RAID Level                            False
reassign_to    120                              Reassign Timeout value                True
reserve_policy no_reserve                       Reserve Policy                        True
rw_timeout     30                               Read/Write Timeout value              True
scsi_id        0x651d00                         SCSI ID                               False
size           10240                            Size in Mbytes                        False
write_cache    yes                              Write Caching enabled                 False
```

On the first cluster node `partition3`, storage appears at the AIX level as shown in Example 4-43.

Example 4-43 Disk configuration on the first cluster node

```
root@virt_p3:/# cfgmgr
root@virt_p3:/# lspv
hdisk0          00c6627ef0aa61cf                    rootvg
active
```

```
hdisk1          none                                    None
hdisk2          none                                    None
hdisk3          none                                    None
```

The second cluster node `partition6` is contained within an HMC-managed IBM System p5 p570 system. The corresponding AIX hostname for `partition6` is `virt_p6`. On the system we have defined 2 VIOS, each having assigned 2 Fibre Channel Adapters. Each Fibre Channel adapter is connected to a different SAN switch. `partition6` has no adapters and uses both VIOS to access data located on external storage.

On VIO `Server1` we defined two hard disks named `hdisk5` and `hdisk6` and changed the reserve policy as shown in Example 4-44

Example 4-44   Defining and configuring hdisk5 and hdisk6 on VIO Server 1

```
$ mkvdev -vdev hdisk5 -vadapter vhost0 -dev virt_p6_shared5
virt_p6_shared5 Available
$ mkvdev -vdev hdisk6 -vadapter vhost0 -dev virt_p6_shared6
virt_p6_shared6 Available

$ chdev -dev hdisk5 -attr reserve_policy=no_reserve
hdisk5 changed
$ chdev -dev hdisk6 -attr reserve_policy=no_reserve
hdisk6 changed

$ lsmap -all | more
SVSA            Physloc                                      Client Partition ID
--------------- -------------------------------------------- ------------------
vhost0          U9117.570.10C5D5C-V8-C10                     0x00000006

VTD                   virt_p6_shared5
LUN                   0x8300000000000000
Backing device        hdisk5
Physloc               U7879.001.DQDKZNP-P1-C2-T1-W200300A0B812106F-L3000000000000

VTD                   virt_p6_shared6
LUN                   0x8400000000000000
Backing device        hdisk6
Physloc               U7879.001.DQDKZNP-P1-C2-T1-W200300A0B812106F-L4000000000000

$ lsdev -dev hdisk5 -attr
attribute      value                            description                        user_settable

PR_key_value   none                             Persistant Reserve Key Value       True
cache_method   fast_write                       Write Caching method               False
ieee_volname   600A0B800012106E0000004D451B9589 IEEE Unique volume name            False
lun_id         0x0003000000000000               Logical Unit Number                False
max_transfer   0x100000                         Maximum TRANSFER Size              True
prefetch_mult  1                                Multiple of blocks to prefetch on read False
pvid           none                             Physical volume identifier         False
```

```
q_type         simple                           Queuing Type                       False
queue_depth    10                               Queue Depth                        True
raid_level     5                                RAID Level                         False
reassign_to    120                              Reassign Timeout value             True
reserve_policy no_reserve                       Reserve Policy                     True
rw_timeout     30                               Read/Write Timeout value           True
scsi_id        0x651d00                         SCSI ID                            False
size           10240                            Size in Mbytes                     False
write_cache    yes                              Write Caching enabled              False
$ lsdev -dev hdisk6 -attr
attribute      value                            description                        user_settable

PR_key_value   none                             Persistant Reserve Key Value       True
cache_method   fast_write                       Write Caching method               False
ieee_volname   600A0B8000120F330000005F451B94A4 IEEE Unique volume name            False
lun_id         0x0004000000000000               Logical Unit Number                False
max_transfer   0x100000                         Maximum TRANSFER Size              True
prefetch_mult  1                                Multiple of blocks to prefetch on read False
pvid           none                             Physical volume identifier         False
q_type         simple                           Queuing Type                       False
queue_depth    10                               Queue Depth                        True
raid_level     5                                RAID Level                         False
reassign_to    120                              Reassign Timeout value             True
reserve_policy no_reserve                       Reserve Policy                     True
rw_timeout     30                               Read/Write Timeout value           True
scsi_id        0x651d00                         SCSI ID                            False
size           10240                            Size in Mbytes                     False
write_cache    yes                              Write Caching enabled              False
```

We repeated the same operations for VIO Server2 and obtained the configuration shown in Example 4-45.

Example 4-45 Configuration of hdisk5 and hdisk6 on VIO Server 2

```
$ lsmap -all | more
SVSA            Physloc                                      Client Partition ID
--------------- -------------------------------------------- ------------------
vhost0          U9117.570.10C5D5C-V9-C20                     0x00000006

VTD                   virt_p6_shared5
LUN                   0x8200000000000000
Backing device        hdisk5
Physloc               U7879.001.DQDKZNP-P1-C6-T1-W200300A0B812106F-L3000000000000

VTD                   virt_p6_shared6
LUN                   0x8300000000000000
Backing device        hdisk6
Physloc               U7879.001.DQDKZNP-P1-C6-T1-W200300A0B812106F-L4000000000000

$ lsdev -dev hdisk5 -attr
attribute      value                            description                        user_settable

PR_key_value   none                             Persistant Reserve Key Value       True
```

```
cache_method   fast_write                       Write Caching method               False
ieee_volname   600A0B800012106E0000004D451B9589 IEEE Unique volume name            False
lun_id         0x0003000000000000               Logical Unit Number                False
max_transfer   0x100000                         Maximum TRANSFER Size              True
prefetch_mult  1                                Multiple of blocks to prefetch on read False
pvid           none                             Physical volume identifier         False
q_type         simple                           Queuing Type                       False
queue_depth    10                               Queue Depth                        True
raid_level     5                                RAID Level                         False
reassign_to    120                              Reassign Timeout value             True
reserve_policy no_reserve                       Reserve Policy                     True
rw_timeout     30                               Read/Write Timeout value           True
scsi_id        0x651d00                         SCSI ID                            False
size           10240                            Size in Mbytes                     False
write_cache    yes                              Write Caching enabled              False
$ lsdev -dev hdisk6 -attr
attribute      value                            description                        user_settable

PR_key_value   none                             Persistant Reserve Key Value       True
cache_method   fast_write                       Write Caching method               False
ieee_volname   600A0B8000120F330000005F451B94A4 IEEE Unique volume name            False
lun_id         0x0004000000000000               Logical Unit Number                False
max_transfer   0x100000                         Maximum TRANSFER Size              True
prefetch_mult  1                                Multiple of blocks to prefetch on read False
pvid           none                             Physical volume identifier         False
q_type         simple                           Queuing Type                       False
queue_depth    10                               Queue Depth                        True
raid_level     5                                RAID Level                         False
reassign_to    120                              Reassign Timeout value             True
reserve_policy no_reserve                       Reserve Policy                     True
rw_timeout     30                               Read/Write Timeout value           True
scsi_id        0x651d00                         SCSI ID                            False
size           10240                            Size in Mbytes                     False
write_cache    yes                              Write Caching enabled              False
```

On `partition6`, at the AIX level we have the configuration shown in Example 4-46.

Example 4-46 Disk configuration on the second cluster node

```
root@virt_p6:/# cfgmgr
root@virt_p6:/# lspath
Enabled hdisk0 vscsi0
Enabled hdisk1 vscsi0
Enabled hdisk1 vscsi1
Enabled hdisk2 vscsi0
Enabled hdisk3 vscsi0
Enabled hdisk3 vscsi1
Enabled hdisk2 vscsi1
root@virt_p6:/# lspv
```

```
hdisk0          00cc5d5c15b0578f                    None
hdisk1          00cc5d5c3e364341                    rootvg          active
hdisk2          none                                None
hdisk3          none                                None
```

The overall configuration for access to external storage is shown in Figure 4-11

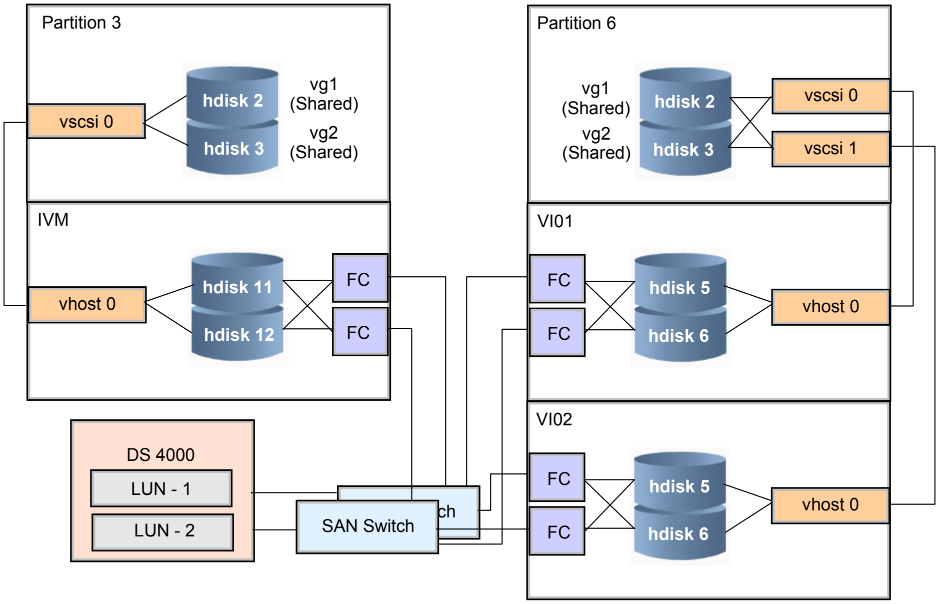


Figure 4-11   Storage access topology

4.2.2 Configuring cluster resource groups

On each cluster node we have two disks available: `hdisk2` and `hdisk3`.

`hdisk2` is used to define volume group vg1. `vg1` is assigned to resource group `rg1`. `rg1` is acquired by first cluster node.

`hdisk3` is used to define volume group vg2. `vg2` is assigned to resource group `rg2`. `rg2` is acquired by the second cluster node.

We maintained the policy of referring to map the same hard drive or LUN to the same hdisk at the AIX level on cluster nodes.

4.2.3 Configuring LPARs for IP network access

The first cluster node uses one virtual Ethernet interface named ent0 which is bridged to virtual interface ent3 defined in the VIOS LPAR. We use one Shared Ethernet Adapter (SEA ent8) which is mapped to virtual interface EtherChannel adapter ent7. The ent7 interface is a link aggregation of two physical adapters (ent1 and ent2) and is mapped to virtual interface ent3.

The second cluster node uses one virtual Ethernet interface ent0 which is bridged to virtual Ethernet interface ent2 of each VIO server. On each VIOS partition virtual Ethernet interface ent2 is part of an SEA named ent5. On each server ent5 interface is mapped to physical Ethernet interface ent0.

All physical interfaces are connected to the same Ethernet switch. Obviously for a production implementation, if possible we would recommend using separate switches.

The topology used for IP data network is shown in Figure 4-12.

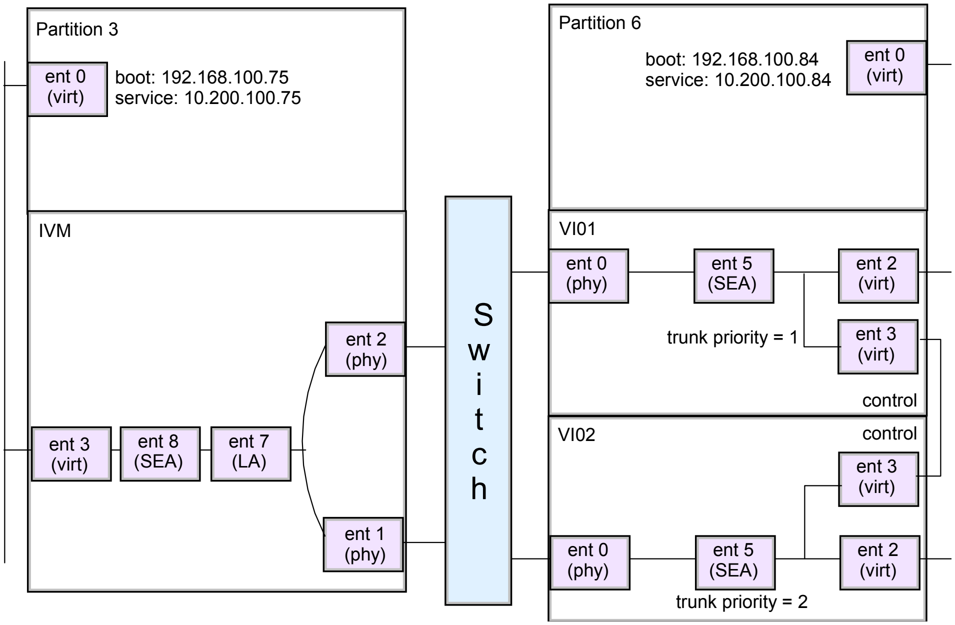


Figure 4-12 IP network topology

4.2.4 Configuring cluster topology

Each partition has one virtual Ethernet interface. For each node we define one boot IP address and one service IP address as described in Table 4-2 on page 139.

Table 4-5 Nodes and IP addresses

Node name	Interface name	IP address
virt_p3	virt_p3	192.168.100.75
	virt_p3s	10.200.100.75
virt_p6	virt_p6	192.168.100.84
	virt_p6s	10.200.100.84

4.2.5 Cluster testing

Failure of one Ethernet adapter from one VIO server

- We unplugged the cable from interface `ent0` of `VIO Server 1.` The physical interface `ent0` is flagged as down and SEA ent5 is disabled as shown in Example 4-47.

Example 4-47 SEA status on VIO Server 1

```
$ entstat -all ent5 | grep Priority
    Priority: 1
  Priority: 1  Active: False
```

- The error messages generated on `VIO Server 1` are shown in Example 4-48. The interface failed at 10:51:20.

Example 4-48 Error messages generated on VIO Server 1

```
$ errlog | more
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
0B41DD00   1025105106 I H ent5           ADAPTER FAILURE
EC0BCCD4   1025105106 T H ent0           ETHERNET DOWN
$ errlog -ls | more
---------------------------------------------------------------------------
LABEL:          SEA_ADAP_FAIL
IDENTIFIER:     0B41DD00

Date/Time:       Wed Oct 25 10:51:20 CDT 2006
Sequence Number: 199
Machine Id:      00CC5D5C4C00
```

```
Node Id:         virt_vios1
Class:           H
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
ADAPTER FAILURE

Probable Causes
ADAPTER FAILURE

Failure Causes
ADAPTER FAILURE

        Recommended Actions
        SWITCHING TO LIMBO STATE

Detail Data
Switching SEA to limbo state until our physical adapter comes back up
ent0
---------------------------------------------------------------------------
LABEL:          GOENT_LINK_DOWN
IDENTIFIER:     EC0BCCD4

Date/Time:       Wed Oct 25 10:51:20 CDT 2006
Sequence Number: 198
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios1
Class:           H
Type:            TEMP
Resource Name:   ent0
Resource Class:  adapter
Resource Type:   14108902
Location:        U7879.001.DQDKZNV-P1-T6
VPD:
        Product Specific.(  ).......2-Port 10/100/1000 Base-TX PCI-X
                                    Adapter
        Network Address.............001125E71FCC
        ROM Level.(alterable).......DV0210

Description
ETHERNET DOWN
```

```
        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES

Detail Data
FILE NAME
line: 149 file: goent_intr.c
PCI ETHERNET STATISTICS
0006 CF34 0063 089B 0000 0003 0000 0003 0000 0000 0000 0000 0000 0000 0000 0001
0000 0000 0063 70C9 0000 0000 7D0D 2A95 0000 0000 0059 5080 0003 2E1A 0007 9A0B
0000 0000 0059 8D1F 0000 0001 0010 AA4F 0000 0000 0000 0000 0000 001E 0000 081F
0000 0000 0000 0000 0000 0000 0000 002D 0000 0001 0000 002C 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 BB40 18F0 0068 0C00 0000 0000 01A0 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000
DEVICE DRIVER INTERNAL STATE
2222 2222 0000 0000 0000 0000
SOURCE ADDRESS
0011 25E7 1FCC
```

- On VIO Server 2 the SEA is active as shown in Example 4-49.

Example 4-49 SEA status on VIO Server 2

```
$ entstat -all ent5 | grep Priority
    Priority: 2
  Priority: 2  Active: True
```

- The errors messages generated on VIO Server 2 are shown in Example 4-50. They are logged one second after ent0 interface failure.

Example 4-50 Error messages generated on VIO Server 2

```
$ errlog | more
E136EAFA   1025105106 I H ent5           BECOME PRIMARY

$ errlog -ls | more
---------------------------------------------------------------------------
LABEL:          SEAHA_PRIMARY
IDENTIFIER:     E136EAFA

Date/Time:       Wed Oct 25 10:51:21 CDT 2006
Sequence Number: 97
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios2
Class:           H
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
BECOME PRIMARY

Probable Causes
BECOME PRIMARY

Failure Causes
BECOME PRIMARY

        Recommended Actions
        BECOME PRIMARY

Detail Data
Become the Primary SEA
```

- At the AIX and HACMP level on cluster nodes there are no errors related to this adapter failure. Therefore the design of our system has proved its resilience - there was no application impact.
- We plugged the cable back into interface `ent0` of `VIO Server 1.` The physical interface `ent0` is up and SEA ent5 is reactivated as shown in Example 4-51.

Example 4-51 SEA ent5 of VIO Server 1 is reactivated

```
$ entstat -all ent5 | grep Priority
    Priority: 1
  Priority: 1  Active: True
```

- SEA `ent5` automatically becomes primary again as shown in Example 4-52.

Example 4-52 SEA ent5 of VIO Server 1 becomes primary

```
$ errlog | more
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
E136EAFA   1025110306 I H ent5           BECOME PRIMARY

$ errlog -ls | more
---------------------------------------------------------------------------
LABEL:          SEAHA_PRIMARY
IDENTIFIER:     E136EAFA

Date/Time:       Wed Oct 25 11:03:56 CDT 2006
Sequence Number: 201
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios1
Class:           H
```

```
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
BECOME PRIMARY

Probable Causes
BECOME PRIMARY

Failure Causes
BECOME PRIMARY

        Recommended Actions
        BECOME PRIMARY

Detail Data
Become the Primary SEA
```

- SEA adapter from VIO Server 2 becomes secondary as shown in Example 4-53.

Example 4-53 SEA ent5 status on VIO Server 2

```
$ entstat -all ent5 | grep Priority
    Priority: 2
  Priority: 2  Active: False
```

- Corresponding error messages are logged in VIO Server 2 as shown in Example 4-54.

Example 4-54 Error messages generated on VIO Server 2

```
$ errlog | more
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
40D97644   1025110306 I H ent5           BECOME BACKUP

$ errlog -ls | more
---------------------------------------------------------------------------
LABEL:          SEAHA_BACKUP
IDENTIFIER:     40D97644

Date/Time:       Wed Oct 25 11:03:56 CDT 2006
Sequence Number: 98
```

```
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios2
Class:           H
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
BECOME BACKUP

Probable Causes
BECOME BACKUP

Failure Causes
BECOME BACKUP

        Recommended Actions
        BECOME BACKUP

Detail Data
Become the Backup SEA
```

- At the AIX and HACMP level on cluster nodes there are no errors related to this failure.

Failure of a single VIOS

- To simulate a VIOS failure, we halted the partition hosting `VIO Server 1`.
- SEA ent5 from the VIO Server2 becomes active as shown in Example 4-55.

Example 4-55   SEA ent5 status on VIO Server 2 after VIO Server 1 crashed

```
$ entstat -all ent5 | grep Priority
    Priority: 2
  Priority: 2  Active: True
```

- SEA ent5 becomes primary and corresponding error messages are logged in VIO Server 2 as shown in Example 4-56.

Example 4-56   Error messages from VIO Server 2

```
$ errlog | more
E136EAFA   1025111006 I H ent5           BECOME PRIMARY
$ errlog -ls | more
---------------------------------------------------------------------------
LABEL:          SEAHA_PRIMARY
IDENTIFIER:     E136EAFA
```

```
Date/Time:       Wed Oct 25 11:10:47 CDT 2006
Sequence Number: 99
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios2
Class:           H
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
BECOME PRIMARY

Probable Causes
BECOME PRIMARY

Failure Causes
BECOME PRIMARY

        Recommended Actions
        BECOME PRIMARY

Detail Data
Become the Primary SEA
```

- `hdisk1` from the second cluster nod loses one path due to `VIO Server 1 failure` and error messages are written in the log as shown in Example 4-57.

Example 4-57   hdisk1 lost one path due to VIO Server 1 failure

```
root@virt_p6:/# errpt | more
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
DE3B8540   1025131206 P H hdisk1         PATH HAS FAILED

root@virt_p6:/# errpt -a | more
---------------------------------------------------------------------------
LABEL:          SC_DISK_ERR7
IDENTIFIER:     DE3B8540

Date/Time:       Wed Oct 25 13:12:07 2006
Sequence Number: 115
Machine Id:      00CC5D5C4C00
Node Id:         virt_p6
Class:           H
Type:            PERM
Resource Name:   hdisk1
Resource Class:  disk
Resource Type:   vdisk
Location:        U9117.570.10C5D5C-V6-C10-T1-L820000000000

Description
PATH HAS FAILED

Probable Causes
ADAPTER HARDWARE OR CABLE
DASD DEVICE

Failure Causes
UNDETERMINED

        Recommended Actions
        PERFORM PROBLEM DETERMINATION PROCEDURES
        CHECK PATH

Detail Data
PATH ID
0
SENSE DATA
0A00 2A00 00A3 C218 0000 0804 0000 0000 0000 0000 0000 0000 0200 0B00 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000 0000
```

- At the AIX and HACMP level on cluster nodes there are no errors related to this failure.
- We restart VIO Server 1. SEA ent5 adapter becomes active again as shown in Example 4-58.

Example 4-58 SEA ent5 adapter from VIO Server 1 becomes active again

```
$ entstat -all ent5 | grep Priority
    Priority: 1
  Priority: 1  Active: True
```

- SEA `ent5` becomes primary and corresponding error messages are logged on `VIO Server 1` as shown in Example 4-59.

Example 4-59 Error logs in VIO Server 1

```
$ errlog | more

IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
136EAFA   1025112106 I H ent5          BECOME PRIMARY
---------------------------------------------------------------------------
```

```
$ errlog -ls | more
LABEL:          SEAHA_PRIMARY
IDENTIFIER:     E136EAFA

Date/Time:       Wed Oct 25 11:21:04 CDT 2006
Sequence Number: 204
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios1
Class:           H
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
BECOME PRIMARY

Probable Causes
BECOME PRIMARY

Failure Causes
BECOME PRIMARY

        Recommended Actions
        BECOME PRIMARY

Detail Data
Become the Primary SEA
```

- SEA ent5 from VIO Server 2 becomes secondary backup and corresponding error messages are logged on `VIO Server 2` as shown in Example 4-60.

Example 4-60 SEA ent5 from VIO Server 2 becomes backup

```
$ errlog | more
IDENTIFIER TIMESTAMP  T C RESOURCE_NAME  DESCRIPTION
40D97644   1025112106 I H ent5           BECOME BACKUP

$ errlog -ls | more
---------------------------------------------------------------------------
LABEL:          SEAHA_BACKUP
IDENTIFIER:     40D97644

Date/Time:       Wed Oct 25 11:21:04 CDT 2006
Sequence Number: 100
Machine Id:      00CC5D5C4C00
Node Id:         virt_vios2
Class:           H
Type:            INFO
Resource Name:   ent5
Resource Class:  adapter
Resource Type:   sea
Location:

Description
BECOME BACKUP

Probable Causes
BECOME BACKUP

Failure Causes
BECOME BACKUP

        Recommended Actions
        BECOME BACKUP

Detail Data
Become the Backup SEA
```

- After `VIO Server 1` is restarted paths corresponding to `vscsi0` and `hdisk2` and `hdisk3` are failed as you can see in Example 4-61.

Example 4-61 Paths to hdisk2, hdisk3 and vscsi0 are failed

```
root@virt_p6:/var/ha/log# lspath
Enabled hdisk0 vscsi0
Enabled hdisk1 vscsi0
Enabled hdisk1 vscsi1
Failed  hdisk2 vscsi0
Failed  hdisk3 vscsi0
Enabled hdisk3 vscsi1
Enabled hdisk2 vscsi1
```

- This can be rectified via smitty MPIO →MPIO Path Management →Enable Paths →Enable Selected or All Paths to restore all path connectivity. Now all paths are enabled as you can see in Example 4-62.

Example 4-62 Paths to hdisk2, hdisk3 and vscsi0 are enabled

```
root@virt_p6:/var/ha/log# lspath
Enabled hdisk0 vscsi0
Enabled hdisk1 vscsi0
Enabled hdisk1 vscsi1
Enabled hdisk2 vscsi0
Enabled hdisk3 vscsi0
Enabled hdisk3 vscsi1
Enabled hdisk2 vscsi1
```

Failure of one cluster node

- We simulated a node failure by stopping the first cluster node using the command **`halt`**.

- On the second node we monitored messages logged by topology services in files created for each of the networks defined between cluster nodes. These are shown in Example 4-63.

Example 4-63 RSCT error messages after one node failure

```
root@virt_p6:/var/ha/log# cat nim.topsvcs.en0.cluster2
10/25 13:56:53.395: Received a SEND MSG command. Dst: 192.168.100.75.
10/25 13:57:13.397: Received a SEND MSG command. Dst: 192.168.100.75.
10/25 13:57:29.898: Heartbeat was NOT received. Missed HBs: 1. Limit: 10
10/25 13:57:31.898: Heartbeat was NOT received. Missed HBs: 2. Limit: 10
10/25 13:57:31.899: Starting sending ICMP ECHOs.
10/25 13:57:31.899: Invoking netmon to find status of local adapter.
10/25 13:57:32.700: netmon response: Adapter is up
10/25 13:57:33.399: Received a SEND MSG command. Dst: 192.168.100.75.
10/25 13:57:33.899: Heartbeat was NOT received. Missed HBs: 3. Limit: 10
10/25 13:57:33.899: Invoking netmon to find status of local adapter.
10/25 13:57:34.699: netmon response: Adapter is up
10/25 13:57:35.899: Heartbeat was NOT received. Missed HBs: 4. Limit: 10
10/25 13:57:35.899: Invoking netmon to find status of local adapter.
10/25 13:57:36.699: netmon response: Adapter is up
10/25 13:57:37.899: Heartbeat was NOT received. Missed HBs: 5. Limit: 10
10/25 13:57:37.899: Invoking netmon to find status of local adapter.
10/25 13:57:38.700: netmon response: Adapter is up
10/25 13:57:39.899: Heartbeat was NOT received. Missed HBs: 6. Limit: 10
10/25 13:57:39.899: Invoking netmon to find status of local adapter.
10/25 13:57:40.700: netmon response: Adapter is up
10/25 13:57:41.899: Heartbeat was NOT received. Missed HBs: 7. Limit: 10
10/25 13:57:41.899: Invoking netmon to find status of local adapter.
10/25 13:57:42.700: netmon response: Adapter is up
10/25 13:57:43.900: Heartbeat was NOT received. Missed HBs: 8. Limit: 10
10/25 13:57:43.900: Invoking netmon to find status of local adapter.
10/25 13:57:44.700: netmon response: Adapter is up
10/25 13:57:45.900: Heartbeat was NOT received. Missed HBs: 9. Limit: 10
10/25 13:57:45.900: Invoking netmon to find status of local adapter.
10/25 13:57:46.700: netmon response: Adapter is up
10/25 13:57:47.900: Heartbeat was NOT received. Missed HBs: 10. Limit: 10
10/25 13:57:47.900: Invoking netmon to find status of local adapter.
10/25 13:57:47.900: Local adapter is up: issuing notification for remote
adapter
10/25 13:57:47.900: Adapter status successfully sent.
10/25 13:57:47.900: Error sending to 192.168.100.75: Bad file number.
10/25 13:57:47.900: Error sending to 192.168.100.75: Bad file number.
10/25 13:57:47.902: Dispatching netmon request while another in progress.
10/25 13:57:47.902: Received a STOP HB command.
10/25 13:57:47.902: Received a STOP MONITOR command.
10/25 13:57:47.902: Dispatching netmon request while another in progress.
10/25 13:57:48.701: netmon response: Adapter is up
10/25 13:57:48.701: Adapter status successfully sent.
```

```
10/25 13:57:52.904: Received a SEND MSG command. Dst: 192.168.100.255.
10/25 13:57:57.915: Received a SEND MSG command. Dst: 192.168.100.75.
10/25 13:58:02.914: Received a SEND MSG command. Dst: 192.168.100.75.

root@virt_p6:/var/ha/log# cat nim.topsvcs.rhdisk2.cluster2
10/25 13:56:29.831: Received a SEND MSG command. Dst: .
10/25 13:56:48.395: Received a SEND MSG command. Dst: .
10/25 13:57:06.615: Received a SEND MSG command. Dst: .
10/25 13:57:28.398: Received a SEND MSG command. Dst: .
10/25 13:57:33.755: Heartbeat was NOT received. Missed HBs: 1. Limit: 4
10/25 13:57:37.755: Heartbeat was NOT received. Missed HBs: 2. Limit: 4
10/25 13:57:41.756: Heartbeat was NOT received. Missed HBs: 3. Limit: 4
10/25 13:57:45.756: Heartbeat was NOT received. Missed HBs: 4. Limit: 4
10/25 13:57:45.756: Local adapter is up: issuing notification for remote
adapter
10/25 13:57:45.756: Adapter status successfully sent.
10/25 13:57:45.757: Received a STOP HB command.
10/25 13:57:45.757: Received a STOP MONITOR command.
10/25 13:57:46.515: Received a SEND MSG command. Dst: .
10/25 13:57:55.764: Received a SEND MSG command. Dst: .
10/25 13:57:57.915: Received a SEND MSG command. Dst: .
10/25 13:58:05.764: Received a SEND MSG command. Dst: .
10/25 13:58:07.126: writePacket(): Unable to write for too long
10/25 13:58:07.926: Received a SEND MSG command. Dst: .
10/25 13:58:15.764: Received a SEND MSG command. Dst: .
10/25 13:58:17.935: Received a SEND MSG command. Dst: .
10/25 13:58:25.764: Received a SEND MSG command. Dst: .
10/25 13:58:27.936: Received a SEND MSG command. Dst: .
10/25 13:58:35.769: Received a SEND MSG command. Dst: .
10/25 13:58:37.937: Received a SEND MSG command. Dst: .
10/25 13:58:42.883: writePacket(): Unable to write for too long
10/25 13:58:45.038: writePacket(): Unable to write for too long
10/25 13:58:45.769: Received a SEND MSG command. Dst: .
10/25 13:58:47.193: writePacket(): Unable to write for too long
10/25 13:58:49.349: writePacket(): Unable to write for too long
10/25 13:58:50.406: Received a SEND MSG command. Dst: .
10/25 13:58:51.506: writePacket(): Unable to write for too long
10/25 13:58:53.662: writePacket(): Unable to write for too long
10/25 13:58:55.774: Received a SEND MSG command. Dst: .
10/25 13:58:57.918: writePacket(): Unable to write for too long
10/25 13:59:00.406: Received a SEND MSG command. Dst: .
10/25 13:59:02.175: writePacket(): Unable to write for too long
10/25 13:59:02.225: 8 failed writes in a row - clearing send queue.
10/25 13:59:05.774: Received a SEND MSG command. Dst: .
10/25 13:59:10.407: Received a SEND MSG command. Dst: .
10/25 13:59:15.774: Received a SEND MSG command. Dst: .
10/25 13:59:20.408: Received a SEND MSG command. Dst: .
10/25 13:59:25.784: Received a SEND MSG command. Dst: .
10/25 13:59:30.409: Received a SEND MSG command. Dst: .
```

```
10/25 13:59:35.791: Received a SEND MSG command. Dst: .
10/25 13:59:40.410: Received a SEND MSG command. Dst: .
10/25 13:59:41.480: writePacket(): Unable to write for too long
10/25 13:59:45.736: writePacket(): Unable to write for too long
10/25 13:59:45.791: Received a SEND MSG command. Dst: .
10/25 13:59:50.411: Received a SEND MSG command. Dst: .
10/25 13:59:52.094: writePacket(): Unable to write for too long
10/25 13:59:55.791: Received a SEND MSG command. Dst: .
10/25 13:59:56.350: writePacket(): Unable to write for too long
10/25 14:00:00.411: Received a SEND MSG command. Dst: .
10/25 14:00:00.606: writePacket(): Unable to write for too long
10/25 14:00:04.863: writePacket(): Unable to write for too long
10/25 14:00:05.794: Received a SEND MSG command. Dst: .
10/25 14:00:10.412: Received a SEND MSG command. Dst: .
10/25 14:00:11.219: writePacket(): Unable to write for too long
10/25 14:00:15.475: writePacket(): Unable to write for too long
10/25 14:00:15.804: Received a SEND MSG command. Dst: .
10/25 14:00:20.413: Received a SEND MSG command. Dst: .
10/25 14:00:21.830: writePacket(): Unable to write for too long
10/25 14:00:21.881: 8 failed writes in a row - clearing send queue.
10/25 14:00:25.804: Received a SEND MSG command. Dst: .
10/25 14:00:30.414: Received a SEND MSG command. Dst: .
10/25 14:00:30.515: dhb_lost_handshake_fct(): Restarting handshaking
10/25 14:00:30.516: initHS(): Wrote initial handshake
10/25 14:00:35.804: Received a SEND MSG command. Dst: .


root@virt_p6:/var/ha/log# cat nim.topsvcs.rhdisk3.cluster2
10/25 13:56:29.831: Received a SEND MSG command. Dst: .
10/25 13:56:48.395: Received a SEND MSG command. Dst: .
10/25 13:57:06.615: Received a SEND MSG command. Dst: .
10/25 13:57:28.399: Received a SEND MSG command. Dst: .
10/25 13:57:33.505: Heartbeat was NOT received. Missed HBs: 1. Limit: 4
10/25 13:57:37.505: Heartbeat was NOT received. Missed HBs: 2. Limit: 4
10/25 13:57:41.506: Heartbeat was NOT received. Missed HBs: 3. Limit: 4
10/25 13:57:45.506: Heartbeat was NOT received. Missed HBs: 4. Limit: 4
10/25 13:57:45.506: Local adapter is up: issuing notification for remote
adapter
10/25 13:57:45.506: Adapter status successfully sent.
10/25 13:57:45.507: Received a STOP HB command.
10/25 13:57:45.507: Received a STOP MONITOR command.
10/25 13:57:46.515: Received a SEND MSG command. Dst: .
10/25 13:57:55.514: Received a SEND MSG command. Dst: .
10/25 13:57:57.915: Received a SEND MSG command. Dst: .
10/25 13:58:05.524: Received a SEND MSG command. Dst: .
10/25 13:58:06.802: writePacket(): Unable to write for too long
10/25 13:58:07.926: Received a SEND MSG command. Dst: .
10/25 13:58:15.524: Received a SEND MSG command. Dst: .
10/25 13:58:17.935: Received a SEND MSG command. Dst: .
```

```
10/25 13:58:25.524: Received a SEND MSG command. Dst: .
10/25 13:58:27.936: Received a SEND MSG command. Dst: .
10/25 13:58:35.524: Received a SEND MSG command. Dst: .
10/25 13:58:37.937: Received a SEND MSG command. Dst: .
10/25 13:58:42.570: writePacket(): Unable to write for too long
10/25 13:58:44.755: writePacket(): Unable to write for too long
10/25 13:58:45.527: Received a SEND MSG command. Dst: .
10/25 13:58:46.911: writePacket(): Unable to write for too long
10/25 13:58:49.067: writePacket(): Unable to write for too long
10/25 13:58:50.406: Received a SEND MSG command. Dst: .
10/25 13:58:51.223: writePacket(): Unable to write for too long
10/25 13:58:53.378: writePacket(): Unable to write for too long
10/25 13:58:55.531: Received a SEND MSG command. Dst: .
10/25 13:58:57.634: writePacket(): Unable to write for too long
10/25 13:59:00.407: Received a SEND MSG command. Dst: .
10/25 13:59:01.890: writePacket(): Unable to write for too long
10/25 13:59:01.940: 8 failed writes in a row - clearing send queue.
10/25 13:59:05.531: Received a SEND MSG command. Dst: .
10/25 13:59:10.407: Received a SEND MSG command. Dst: .
10/25 13:59:15.531: Received a SEND MSG command. Dst: .
10/25 13:59:20.408: Received a SEND MSG command. Dst: .
10/25 13:59:25.534: Received a SEND MSG command. Dst: .
10/25 13:59:30.409: Received a SEND MSG command. Dst: .
10/25 13:59:35.534: Received a SEND MSG command. Dst: .
10/25 13:59:40.410: Received a SEND MSG command. Dst: .
10/25 13:59:41.238: writePacket(): Unable to write for too long
10/25 13:59:45.518: writePacket(): Unable to write for too long
10/25 13:59:45.536: Received a SEND MSG command. Dst: .
10/25 13:59:49.774: writePacket(): Unable to write for too long
10/25 13:59:50.411: Received a SEND MSG command. Dst: .
10/25 13:59:55.536: Received a SEND MSG command. Dst: .
10/25 13:59:56.130: writePacket(): Unable to write for too long
10/25 14:00:00.386: writePacket(): Unable to write for too long
10/25 14:00:00.411: Received a SEND MSG command. Dst: .
10/25 14:00:04.641: writePacket(): Unable to write for too long
10/25 14:00:05.536: Received a SEND MSG command. Dst: .
10/25 14:00:10.412: Received a SEND MSG command. Dst: .
10/25 14:00:10.997: writePacket(): Unable to write for too long
10/25 14:00:15.254: writePacket(): Unable to write for too long
10/25 14:00:15.544: Received a SEND MSG command. Dst: .
10/25 14:00:20.413: Received a SEND MSG command. Dst: .
10/25 14:00:21.609: writePacket(): Unable to write for too long
10/25 14:00:21.659: 8 failed writes in a row - clearing send queue.
10/25 14:00:25.544: Received a SEND MSG command. Dst: .
10/25 14:00:30.414: Received a SEND MSG command. Dst: .
10/25 14:00:30.515: dhb_lost_handshake_fct(): Restarting handshaking
10/25 14:00:30.516: initHS(): Wrote initial handshake
10/25 14:00:35.544: Received a SEND MSG command. Dst: .
```

```
10/25 14:00:40.414: Received a SEND MSG command. Dst: .
```

A

Sample scripts

This appendix lists the various scripts we used to test the scenarios in this book.

A.1 DCEM backup in CSM cluster

A.1.1 The script used with DCEM

This script is used in 3.3.3, “Using DCEM as a backup strategy in an CSM cluster” on page 88. The script mounts the remote NFS directory and runs the **mksysb** command. Return codes are given special consideration.

Example: A-1 AIX operating system backup script /usr/local/bin/aixos_backup.sh

```
#!/bin/ksh
##*
##* Tested on: AIX 5.2,AIX 5.3
##*
#-- Description:
#-- ============
#-- Saves the AIX OS (mksysb) image to the NFS directory
##* *******************************************************************

#---------------------------------------------------------------
#--     Initialisation
#---------------------------------------------------------------
#--  Assign the parameters of environment and data file
#---------------------------------------------------------------
INIFILE=/ap/os/etc/aixos_backup.ini     #-- environment file in which
are parameters about
                         #-- destination of VG image, etc stored

HOSTNAME=`hostname`

#----------------------------------------------------------------------
#----------------------------------------------------------------------
#-- Definition of local functions
#----------------------------------------------------------------------

#----------------------------------------------------------------------
#-- UN-mount the NFS remote directory and check return code
#-- The mount name is used (see /etc/filesystems for defined names)
#-- for the mount and umount operations.
#----------------------------------------------------------------------
cleanup()
{
   echo "Unmounting the NFS directory: $MOUNTNAME"#-- Unmounting NFS...
   umount -t $MOUNTNAME         #-- the umount operation
   if [ $? -ne 0 ]              #-- return code check
```

```
     then
      echo "Unmount the NFS directory: $MOUNTNAME failed"#-- Unmounting
NFS...
     return
   fi
    echo "Unmount of the $MOUNTNAME finished OK"
}
#-----------------------------------------------------------------------

#-----------------------------------------------------------------------
#-- Issues messages and exits
#-----------------------------------------------------------------------
exit_with_error()
{
   echo "The mksysb on $HOSTNAME failed"
   exit $@
}

#----------------------------------------------------------------
#--  Check for environment file, where is stored:
#--    - destination file system (usualy mounted by NFS)
#--    - mount name of the NFS file system
#--    - VG name to backup
#----------------------------------------------------------------
load_file_profile $INIFILE
if [ $? -ne 0 ]
   then
    exit_with_error 1
fi

#----------------------------------------------------------------
#--  Load the environment
#----------------------------------------------------------------
IMGNAME=${HOSTNAME}.bos                       # Name of the Image
IMGFULLNAME=${VGDEST}/${IMGNAME}             # Full name of the Image
IMGCOPYFULLNAME=${VGDEST}/copy/${IMGNAME}            # Full name of
the Image

#----------------------------------------------------------------
#-- Mount the remote NFS directory where the VG image is
#-- temporarily stored
#----------------------------------------------------------------

#-- Message about mounting of the NFS directory
echo "Mounting the NFS filesystem $MOUNTNAME"
```

```
  mount -t $MOUNTNAME   #-- Mount of the remote directory
  if [ $? -ne 0 ]       #-- and check if this was successfull
    then
    echo"Mounting the NFS directory $MOUNTNAME failed, exiting"
    cleanup
    exit_with_error 2
  fi

#-- Message about successfull NFS mount
 echo "NFS Mount $MOUNTNAME finished OK"
#---------------------------------------------------------------

#---------------------------------------------------------------
#-- Remove the old incomplete image, if there is one
#---------------------------------------------------------------
  if [ -f $IMGFULLNAME ]
    then
    echo "Removing the old incomplete image $IMGFULLNAME"
    rm -f $IMGFULLNAME > /dev/null 2>&1
  fi
#---------------------------------------------------------------
#---------------------------------------------------------------
#-- Rename the old image, if there is one
#---------------------------------------------------------------
  if [ -f $IMGCOPYFULLNAME ]
    then
    mv $IMGCOPYFULLNAME ${VGDEST}/arch/
  fi
#---------------------------------------------------------------

#---------------------------------------------------------------
#-- Run the AIX mksysb command to backup VG image
#---------------------------------------------------------------
  echo "Running mksysb on $HOSTNAME, image path: $IMGFULLNAME"
  mksysb -i -m -X $IMGFULLNAME >> $LOGFILE 2>&1
  rc=$?
  if [ $rc -ne 0 ]
    then
    rm $IMGFULLNAME > /dev/null 2>&1    #-- Normaly when mksysb is not
successfull
    cleanup                             #-- this is  not needed
    exit_with_error 1
  fi
```

```
#---------------------------------------------------------------
echo "Mksysb backup on $HOSTNAME finished OK"
#---------------------------------------------------------------

#---------------------------------------------------------------
#-- move a image file to copy directory
#---------------------------------------------------------------
  mv $IMGFULLNAME ${VGDEST}/copy/
  rcmv=$?
  if [ $rcmv -ne 0 ]
    then
    echo "Image move $IMGFULLNAME to ${VGDEST} failed on $HOSTNAME"
    else
    echo "Image move $IMGFULLNAME to ${VGDEST} on $HOSTNAME finished
OK"
  fi
#---------------------------------------------------------------

#---------------------------------------------------------------
#-- To reach this point, all steps have to be successfull,
#-- Unmount the NFS is needed
#---------------------------------------------------------------
  cleanup
#---------------------------------------------------------------

#----------------------------------------------------------------------
#-- Write termination messagess
#----------------------------------------------------------------------
echo "OS Backup finished (`date`) with rc:$rc"
exit $rc
```

A.1.2 DCEM script initialization (.ini) file

This is the ini file used in conjunction with the DCEM script for passing certain parameters in 3.3.3, “Using DCEM as a backup strategy in an CSM cluster” on page 88.

Example: A-2 AIX operating system backup /usr/local/etc/aixos_backup.ini

```
DEST=/nfs/aiximage
MOUNTNAME=aiximg
```

A.2 FC adapter inventory script

This script is used in 3.3.14, “Hardware and software inventory and configuration” on page 113.

Example: A-3 FC inventory script /usr/local/bin/get_fc_wwn.sh

```
#!/bin/sh

for i in `lsdev -Cc adapter | grep fc | awk '{print $1}'`
  do
  lscfg -vl $i | grep -E "fcs|Network"
done
```

A.3 HMC Version 3.7 monitoring script

This script is used in 3.3.5, “Monitoring SFP hardware events via HMC” on page 98.

Example: A-4 HMC Version 3.7 monitoring script /usr/local/bin/hmc4_monitor.sh

```
#!/bin/ksh

#-- This is tested with HMC v 3.7.1
#-- it will not work with HMC v 5.x.x

HMCs="hmcitso"#-- list of HMCs to track
TIME_BACK_TRACK=10#-- time to look for the events in seconds
TMPFILE="/tmp/hmc4_monitor.$$"


#-- Process each HMC and find it's CECs
for hmc in $HMCs
  do
  CECs=`ssh hscroot@${hmc} lssyscfg -r sys --all -F name`
  if [ $? -ne 0 ]
    then
    echo "Error inquiring managed systems in script $0, check commands
syntax"
    exit 1
  fi

  for CEC in $CECs#-- For each CEC enquire the events
    do
```

```
     ssh hscroot@${hmc} lssvcevents -t hardware -m $CEC -s ALL -i
$TIME_BACK_TRACK  -F status name created_time description > $TMPFILE
2>&1
      BODY=`cat $TMPFILE | grep -v 'No results were found.'`
      if [ ! -z "$BODY" ]
        then#-- there are events, report them
   echo "There are new open SFP events on HMC:$hmc CEC:$CEC:$BODY"
      fi

  done
done

rm -f TMPFILE#-- cleanup
```

A.4 HMC Version 5.2.1 monitor script

This script is used in 3.3.5, “Monitoring SFP hardware events via HMC” on page 98.

Example: A-5 HMC Version 5.2.1 monitoring script /usr/local/bin/hmc5_monitor.sh

```
#!/bin/ksh

#-- This is tested with HMC v 5.2.1
#-- it will not work with HMC v 3.x.x

HMCs="hmcp570"#-- HMCs to connect to
TIME_BACK_TRACK=10#-- time to look for the events in seconds
TMPFILE="/tmp/hmc5_monitor.$$"

#-- Process each HMC and find it's CECs
for hmc in $HMCs
  do
  CECs=`ssh hscroot@${hmc} lssyscfg -r sys -F name`
  if [ $? -ne 0 ]
    then
    exit 1
  fi

  for CEC in $CECs#-- For each CEC enquire the events
    do
     ssh hscroot@${hmc} lssvcevents -t hardware -i $TIME_BACK_TRACK -F
"status sys_name created_time text" > $TMPFILE  2>&1
      if [ $? -ne 0 ]
```

```
        then #-- Announce the bad command line syntax
   echo "The ssh command to $hmc did not complete \
successfully, check the command line options in $0"
      fi
      BODY=`cat $TMPFILE | grep -v 'No results were found.' | grep
"Open"`
      if [ ! -z "$BODY" ]
        then#-- If there are events, report them
   echo "There are new open SFP events on HMC:$hmc CEC:$CEC:$BODY"
      fi

  done
done

rm -f $TMPFILE #-- cleanup
```

A.5 VIO Version 1.3 monitor script

This script is used in 3.3.6, “Monitoring of the VIOS error log” on page 101. The current issue of this script is that if someone occasionally clears the error log, first message following the deletion is not detected.

Example: A-6 VIO_errorlog.pl script to monitor error log on VIO servers

```
#!/usr/bin/perl -w

##* ******************************************************************
##* $Author: tcepelk $
##*
##* File Onwership:     root:system
##* File mode:          rwxr-xr-x
##*
##* Tested on: AIX 5.3
#-- Description:
#-- ============
#-- This script checks the new errors (from ioscli errlog command) since last run
#-- of the script
#-- /usr/ios/utils/VIO_errorlog.pl
##* ******************************************************************


 $rc=0;                               #-- General assumption
```

```
##--Help variables----------------------------------------------------
 $last_record="/tmp/errlog_last_record";#-- Here is stored timestamp of last script
run
 $hostname=`ioscli hostname`;
 chomp( $hostname );
#----------------------------------------------------------------------
 if( -f $last_record ){
   $test_content=`grep -E '[A-Z]' $last_record`;
   if (not($test_content =~ /[A-Z]/ )){
#     print "File does not containe any record\n";
      &inicialize_tmp_file();
      exit $rc; #-- This happen at the first command run and with no error log
messages
      }
   open ( LAST_RECORD, "<$last_record");
   $last_record_line = <LAST_RECORD>;
   chomp( $last_record_line );
   ($timestamp,$message_code) = split (/:/,$last_record_line);
   close LAST_RECORD;
   }
 else{
   &inicialize_tmp_file();
   }

 $position=`ioscli errlog|grep -v "IDENTIFIER TIMESTAMP"|grep -nE '$message_code
[[:space:]] $timestamp'|head -1|awk -F: '{print \$1}'`;
 $rc=$?;
 chomp($position);

 if (! $position){
#  print "Error messages not found\n";
   &inicialize_tmp_file();
   exit 1;
   }

 $count=$position - 1;

 if ($count > 0){
   $last_message=`ioscli errlog | head -2 | tail -1`;
   $last_message =~ s/\s+/_/g;
   print "In the error log on $hostname is: $count new records\n";
   print "Last Error log message:$last_message\n";

   #-- Writing last status to the temporary file
   &inicialize_tmp_file();
   }
 else{
#  print "No new messages\n";
   }

 sub inicialize_tmp_file(){
   $timestamp = `ioscli errlog|grep -v "IDENTIFIER TIMESTAMP"|head -1|awk '{print 
\$2}'`;
        $message_code = `ioscli errlog|grep -v "IDENTIFIER TIMESTAMP"|head -1|awk 
'{print \$1}'`;
        chomp($timestamp);
        chomp($message_code);
        open ( LAST_RECORD, ">$last_record");
        printf LAST_RECORD "$timestamp:$message_code";
        close LAST_RECORD;
   }

#-----------------------------------------------------------------------
#-- Write termination messagess
#-----------------------------------------------------------------------
# print "end of the processing\n";
 exit $rc;
```

A.6 Nmon performance collection scripts

There few steps to get data to the RRDTool database.

1. Install nmon software.
2. Get the nmon text file with appropriate values with command, such as Example A-7. File disks.txt contains information about Volume Groups to hdisk mapping so that we collect summarized data for the VGs. -f flag means saving the output into a file. Its standard form is <hostname>_YYYYMMDD_HHMM.nmon, but you can choose the own one using "-F <output-file-name>". For for more info type: **nmon -h**

Example: A-7 Collect the nmon data on a node

```
# nmon -f -s 600 -c 139 -T -g ./disks.txt -W -A
```

3. use nmon2rrd, which prepares the sets of RRDtool commands for creating and updating own database as in Example 3 on page 207. rrd_* files are generated options for rrdtool command. You can run this commands as "rrdtool <text-of-each-line>" or edit "rrdtool" on beginning of each line of the file for batch execution. (In that case do not forget to change the file mode to

executable). Import nmon data into RRDTool database by nmon2rrd command

```
#nmon2rrd -f sample_060601_0000.nmon.csv -d output_dir
# ls -l
-rw-r--r--   1 root     system         3063 Oct 16 09:57 index.html
-rw-r--r--   1 root     system         9135 Oct 16 09:57 rrd_create
-rw-r--r--   1 root     system        36514 Oct 16 09:57 rrd_graph
-rw-r--r--   1 root     system        85062 Oct 16 09:57 rrd_top
-rw-r--r--   1 root     system       255649 Oct 16 09:57 rrd_update
```

4. It is necessary to proceed the files in correct order. rrd_create file generates empty databases (*.rrd files) for all items from *.csv input file for appropriate time range. By editing this file, it's possible to edit options to customize characteristics of output charts, see Example A-8. Here the --start option (1160863207) means start time in seconds since January 1st 1970 UTC, --step option (600) means size of step in second and the last value (140) means number of steps. If you need to prepare the database at once for further updates, it's necessary to edit number of steps, because it's not possible to additionally expand database. If we want to create the database for one year period in the example above, we will change the last value from 140 to 52560 (1 year = 600*52560 seconds).

Example: A-8   nmon .csv input file

```
# head -1 rrd_create
rrdtool create cpu_all.rrd --start 1160863207 --step 600
DS:User:GAUGE:1200:U:U DS:Sys:GAUGE:1200:U:U DS:Wait:GAUGE:1200:U:U
DS:Idle:GAUGE:1200:U:U  RRA:AVERAGE:0.5:1:140
```

5. Into a once created database, you can add new data using rrd_update and rrd_top. Data must be newer than the latest old updates in DB. It doesn't matter, if there is a hole in the data round. It causes only empty white interval in the chart.
6. After generating charts with rrd_graph is the file index.html ready to use.

External links

Nmon, free tool for performance monitoring. Allows on-screen displaying or saving into csv textfile with comma separated values. Available for AIX and Linux: http://www-941.ibm.com/collaboration/wiki/display/WikiPtype/nmon

nmon2rrd, converts nmon CSV files to RRD databases:
http://users.ca.astound.net/baspence/AIXtip/nmon2rrdv1.htm

RRDTool, Round Robin Database Tool. Stores and displays time-series data:
http://oss.oetiker.ch/rrdtool/

A.7 Inventory scripts

Here is a brief description how to use CVS to store inventory data using Tivoli Enterprise data collections features. Note that script can be adjusted for use with dcp commands.

1. Install the CVS software.
2. Run task on specified endpoint which acts as a CVS repository. The task reads central configuration file (see Example A-10). The configuration file contains all endpoints (or profile manager containing endpoints) we want to get files from and the list of files we want to store.

Example: A-9 Task for getting the inventory data

```
PMNGR:physic_ep.sap.nd.aix.SLA.pm
/etc/hosts
/etc/passwd
/etc/environment
/etc/filesystems
/etc/group
/etc/inittab
/etc/hosts.allow
/etc/hosts.deny
/etc/hosts.equiv
/etc/inetd.conf
/etc/services
/etc/ldap.conf
/etc/ntp.conf
/etc/resolv.conf
/etc/sudoers
/tmp/sla/inv/\*
EP_NAME:smbcmnga_sap_1
/cfmroot/ap/os/etc/FS_check.cfg._\*
/cfmroot/ap/os/etc/sap/\*.cfg\*
```

3. Files are stored at the tar archive on the endpoint (EP_NAME or the member of profile manager PMNGR). See the commands to use in Example A-10.

Example: A-10 Commands to prepare files on the target endpoints

```
wadminep $ENDPOINT send_file $SEND_TARRING_SCRIPT
wadminep $ENDPOINT send_file $SEND_LIST_OF_FILES_TO_STORE
wadminep $ENDPOINT exec_process $RUN_TARRING_SCRIPT
```

4. This "local" tar archive is than moved to the target (CVS) endpoint into temporary folders hierarchy by Tivoli configuration manager command **wspmvdata** see Example A-11.

Example: A-11 Inventory data movement into the CVS server

```
wspmvdata -s @$ENDPOINT -t @$TARGET_EP -P sp:PATH_TO_LOCAL_ARCHIVE -P
tp:TARGETPATH -r spre:PATH_TO_TARRING_SCRIPT -r
tpost:PATH_TO_UNTARRING_SCRIPT THE_NAME_OF_LOCAL_ARCHIVE
```

5. Then the tar archive is decompressed to original CVS structure and committed to the repository by script see Example A-12.

Example: A-12 Inventory script using Tivoli CLI; stores data into the CVS system

```
#!/bin/ksh
##---------------------------------------------------------------------
#-- Variables definitions
##---------------------------------------------------------------------
general_config="$LCF_DATDIR/data_moving/files_to_download.txt"; #--
file with all eps/files definitions
compressing_script='tar_files_to_store.sh'; #-- path to script, which
tars desired files
remote_working_dir='/ap/os/etc/data_moving';
TARGETPATH="$LCF_DATDIR/data_moving/content";
post_script="$LCF_DATDIR/data_moving/untar_stored_archives.sh";
TARGET_EP='EP_NAME_TO_STORE_TARS';

##---------------------------------------------------------------------
#-- CVS
##---------------------------------------------------------------------
SCR_DIR="$TARGETPATH";
CVS_DIR="/tmp/"`basename $0`".$$.d";
LOGFILE="/tmp/"`basename $0`"$$.log";
export CVSROOT="/ap/os/cvs/cvsroot";

TO_REMOVE="/tmp/"`basename $0`"*";

rm -rf $TO_REMOVE;


if ! [ -d $TARGETPATH ];then
    mkdir -p $TARGETPATH;
    fi



##-----------------------------------------------------------------------
#-- Each line, except commented (#) lines
#-- Creating individual config files (parsing from general config file)
##-----------------------------------------------------------------------
list_of_individual_files='temporary_list';
cat /dev/null > $list_of_individual_files;

rm $LCF_DATDIR/data_moving/*.files.txt

for line_config in `cat $general_config|grep -vE "^#"`
do

    if [[ $line_config = EP_NAME:* ]];then

        ep_name=${line_config#EP_NAME:};        #-- gathering the
endpoint name from config
        individual_config_file="${ep_name}.files.txt";
        echo creating $individual_config_file ...;

        echo $individual_config_file > $list_of_individual_files;

        if [ -e $LCF_DATDIR/data_moving/$individual_config_file ];then
            rm $LCF_DATDIR/data_moving/$individual_config_file;
            fi

    elif [[ $line_config = PMNGR:* ]];then
        profile_manager=${line_config#PMNGR:};
        cat /dev/null > $list_of_individual_files;
        for ep_name in `wgetsub @$profile_manager`
        do
            individual_config_file="${ep_name}.files.txt";
            echo $individual_config_file >> $list_of_individual_files
            done

    else
        file_name=`echo $line_config|sed 's/\\\//g'`;
        for individual_config_file in `cat $list_of_individual_files`
        do
            printf "$file_name\n" >>
$LCF_DATDIR/data_moving/$individual_config_file;
            done
        fi
    done
##---------------------------------------------------------------------


for line_config in `cat $general_config|grep -vE "^#"`
do
    if [[ $line_config = EP_NAME:* ]];then
        ep_name=${line_config#EP_NAME:};        #-- gathering the
endpoint name from config
        individual_config_file="${ep_name}.files.txt";
        wadminep $ep_name exec_process mkdir -p $remote_working_dir #--
preparing remote working directory
        wadminep $ep_name send_file
$LCF_DATDIR/data_moving/$individual_config_file
$remote_working_dir/$individual_config_file #-- sending individual
config file
        wadminep $ep_name send_file
$LCF_DATDIR/data_moving/$compressing_script
$remote_working_dir/$compressing_script #-- sending the tar script
        wadminep $ep_name exec_process chmod +x
$remote_working_dir/$compressing_script #-- setting executing
permissions


##---------------------------------------------------------------------
        #-- Executing the distribution (spawning the target)

##---------------------------------------------------------------------
        wspmvdata -s @$ep_name -t @$TARGET_EP -P sp:$remote_working_dir
-P tp:${TARGETPATH}/${ep_name} -r
spre:${remote_working_dir}/${compressing_script} -r tpost:$post_script
downloaded_files.tar

        fi

    if [[ $line_config = PMNGR:* ]];then
        profile_manager=${line_config#PMNGR:};
        for ep_name in `wgetsub @$profile_manager`
        do
            individual_config_file="${ep_name}.files.txt";
            wadminep $ep_name exec_process mkdir -p $remote_working_dir
#-- preparing remote working directory
            wadminep $ep_name send_file
$LCF_DATDIR/data_moving/$individual_config_file
```

```
$remote_working_dir/$individual_config_file #-- sending individual
config file
            wadminep $ep_name send_file
$LCF_DATDIR/data_moving/$compressing_script
$remote_working_dir/$compressing_script #-- sending the tar script
            wadminep $ep_name exec_process chmod +x
$remote_working_dir/$compressing_script #-- setting executing
permissions


##---------------------------------------------------------------------
            #-- Executing the distribution (spawning the target)

##---------------------------------------------------------------------
            wspmvdata -s @$ep_name -t @$TARGET_EP -P
sp:$remote_working_dir -P tp:${TARGETPATH}/${ep_name} -r
spre:${remote_working_dir}/${compressing_script} -r tpost:$post_script
downloaded_files.tar
            done
        fi

    done

##---------------------------------------------------------------------
#-- Commiting the spawning result to CVS repository
#-- Possibility to commit older versions of files (from previous runs)
due to
#-- delay of Tivoli distribution.
##---------------------------------------------------------------------
if [ -e $CVS_DIR ] ; then
  rm -rf $CVS_DIR
fi
mkdir -p $CVS_DIR


##---------------------------------------------------------------------
#-- 1. cvs checkout
##---------------------------------------------------------------------
cd $CVS_DIR
cvs checkout SLA/scans >> $LOGFILE 2>&1
rc=$?
if [ $rc -ne 0 ] ; then
  exit $rc
fi
```

```
##----------------------------------------------------------------------
#-- 2. passing SCR_DIR - if any folder or file is missing in CVS, then
addition
##----------------------------------------------------------------------

cd $SCR_DIR
for endpoint in `/bin/find .|awk -F/ '{print $2}'|sort|uniq|grep -vE
"^$"`
do
        cd $endpoint
        /bin/find . |/bin/grep ./ | while read PATH
        do
                if ! [ -e
$CVS_DIR/SLA/scans/$endpoint/config_files/$PATH ];then
                        if [ -d $SCR_DIR/$endpoint/$PATH ];then
                                /usr/bin/mkdir -p
$CVS_DIR/SLA/scans/$endpoint/config_files/$PATH
                        else
                                /usr/bin/touch
$CVS_DIR/SLA/scans/$endpoint/config_files/$PATH
                                fi
                        cd $CVS_DIR
                        /usr/bin/cvs add SLA/scans/$endpoint
                        /usr/bin/cvs add
SLA/scans/$endpoint/config_files/
                        /usr/bin/cvs add
SLA/scans/$endpoint/config_files/$PATH >> $LOGFILE
                        fi
                if [ -f $SCR_DIR/$endpoint/$PATH ];then
                        /usr/bin/cp -p $SCR_DIR/$endpoint/$PATH
$CVS_DIR/SLA/scans/$endpoint/config_files/$PATH
                        cd $CVS_DIR
                        /usr/bin/cvs add
SLA/scans/$endpoint/config_files/$PATH >> $LOGFILE
                        fi
                /usr/bin/cvs update
SLA/scans/$endpoint/config_files/$PATH >> $LOGFILE
                done
        cd $SCR_DIR
        done
##----------------------------------------------------------------------
#-- 3. cvs commit
##----------------------------------------------------------------------
```

```
cd $CVS_DIR
timestamp=`/usr/bin/date`
timestamp_1=`/usr/bin/date +"%Y_%m_%d"`
timestamp_2=`echo d_$timestamp_1`
cd $CVS_DIR/SLA/scans
/bin/chmod -R a+r ./*

for file in `/bin/find ./*/config_files/*|/bin/grep -v CVS`
do
    if [ -f $file ];then

        /usr/bin/su cepelka -c "/bin/cvs commit -m \"$timestamp\"
$file"
        if [[ $? -eq 0 ]];then
            /bin/cvs -r tag $timestamp_2 $file
            fi
        fi
    done

##---------------------------------------------------------------------
#-- 4. cvs removing old tags
##---------------------------------------------------------------------
cvs status -v >/tmp/cvs_status 2>&1
tags_count=`/bin/cat /tmp/cvs_status|/bin/grep \"revision: \"|/bin/awk
'{print $1}'|/bin/sort|/bin/uniq|/bin/wc -l`
if [ $tags_count -gt 20 ];then
    tag_to_remove=`cat /tmp/cvs_status|grep \"revision: \"|awk '{print
$1}'|sort|uniq|head -1`
    cvs rtag -d $tag_to_remove SLA
    fi
```

Abbreviations and acronyms

AIX	Advanced Interactive Executive
AMD	Advanced Micro Devices
APV	Advanced POWER Virtualization
ARP	Address Resolution Protocol
BOS	Base Operating System
CCMDB	Change and Configuration Management Database
CEC	Central Electronic Complex
CFM	Configuration File Manager
CLI	Command Line Interface
CPU	Central Processing Unit
CRH	Cluster Ready Hardware
CRHS	Cluster Ready Hardware Server
CSM	Cluster Systems Management
C-SPOC	Cluster Single Point Of Control (HACMP)
CVS	Concurrent Versions System
CWS	Control Workstation
DASD	Direct Access Storage Device
DB	Database
DCEM	Distributed Command Execution Manager
DHCP	Dynamic Host Configuration Protocol
DIMM	Dual-in-line Memory Module
DLPAR	Dynamic Logical Partition
FC	Fibre Channel
FRU	Filed Replaceable Unit
FSP	Frame Service Processor
GLVM	Geographical Logical Volume Manager
GPFS	General Parallel File System™
GPL	GNU Public License
GUI	Graphical User Interface
HA	High Availability
HACMP	High Availability Cluster Multi-Processing
HACMP/XD	HACMP Extended Distance
HMC	Hardware Management Console
HPC	High Performance Computing
HPS	High Performance Switch
HTML	HyperText Markup Language
HW	Hardware
IBM	International Business Machines Corporation
ICMP	Internet Control Messaging Protocol
ID	Identity
IP	Internet Protocol
IT	Information Technology
ITSO	International Technical Support Organization
IVM	Integrated Virtualization Manager
LAA	Locally Administered Address
LAN	Local Area Network
LPAR	Logical Partition
LPP	Licensed Program Product
LUN	Logical Unit Number
LV	Logical Volume
LVM	Logical Volume Manager

MAC	Media Access Control
MPIO	Multi-Path I/O
NFS	Network File System
NIM	Network Installation Manager
ODM	Object Data Manager
OS	Operating System
PCI	Peripheral Component Interconnect
PCI-X	PCI - Extended
PCM	Path Control Module
PLM	Partition Load Manager
POWER	Performance Optimization With Enhanced RISC
PPRC	Peer-to-Peer Remote Copy
PSSP	Parallel Systems Support Program
PV	Physical Volume
PVID	Physical Volume ID
RAID	Redundant Array of Independent Disks
RAM	Random Access Memory
RAS	Reliability Availability Serviceability
RMC	Resource Monitoring and Control
ROI	Return Of Investment
ROM	Read-only Memory
RPD	RSCT Peer Domain
RPM	Red Hat Package Manager
RSCT	Reliable Scalable Clustering Technology
SAN	Storage Area Network
SCSI	Small Computer System Interface
SFP	Service Focal Point
SLA	Service Level Agreement
SMIT	System Management Interface Tool
SMS	System Management Services
SMT	Symmetric Multi-Threading
SNMP	Simple Network Management Protocol
SPOF	Single Point Of Failure
SPOT	Shared Product Object Tree
SRC	System resource Controller
SSA	Serial Storage Architecture
SSH	Secure Shell
SSL	Secure Socket Layer
SUMA	Service Update Management Assistant
TCO	Total Cost of Ownership
TCP/IP	Transmission Control Protocol/ Internet Protocol
TEC	Tivoli Enterprise Console®
TSA	Tivoli Systems Automation
TSM	Tivoli Storage Manager
UDP	Universal Datagram Protocol
UPS	Uninterruptible Power Supply
VG	Volume Group
VIO	Virtual I/O
VIOS	Virtual I/O Server
VLAN	Virtual LAN
VPD	Vital Product Data
VSCSI	Virtual SCSI
VTERM	Virtual terminal
WAN	Wide Area Network
WAS	WebSphere Application Server
WLM	Workload Manager
WWPN	World-Wide Port Number

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks

For information about ordering these publications, see "How to get IBM Redbooks" on page 218. Note that some of the documents referenced here may be available in softcopy only.

- *Advanced POWER Virtualization on IBM eServer p5 Servers: Architecture and Performance Considerations*, SG24-5768
- *Cluster Systems Management Cookbook for pSeries*, SG24-6859
- *CSM Guide for the PSSP System Administrator*, SG24-6953
- *Transition from PSSP to Cluster Systems Management (CSM)*, SG24-6967
- *An Introduction to the New IBM eServer pSeries High Performance Switch*, SG24-6978
- *Partitioning Implementations for IBM eServer p5 Servers*, SG24-7039
- *IBM System p5 Approaches to 24x7 Availability Including AIX 5L*, SG24-7196
- *IBM BladeCenter JS21: The POWER of Blade Innovation*, SG24-7273
- *NIM from A to Z in AIX 5L*, SG24-7296
- *Advanced POWER Virtualization on IBM System p5: Introduction and Configuration*, SG24-7940
- *Integrated Virtualization Manager on IBM System p5*, REDP-4061
- *IBM BladeCenter JS21 Technical Overview and Introduction*, REDP-4130
- *IBM System p Advanced POWER Virtualization Best Practices*, REDP-4194
- *IBM Director on System p5*, REDP-4219
- *IBM System p5 570 Technical Overview and Introduction*, REDP-9117

Other publications

These publications are also relevant as further information sources:

- *RSCT Administration Guide*, SA22-7889
- *RSCT for AIX 5L Technical Reference*, SA22-7890
- *IBM Cluster Systems Management for AIX 5L and Linux 1.6 Planning and Installation Guide*, SA23-1344
- *IBM Cluster Systems Management for AIX 5L and Linux V1.6 Command and Technical Reference*, SA23-1345
- *HACMP for AIX 5L V5.4 Planning and Installation Guide*, SC23-4861
- *HACMP for AIX 5L V5.4 Adminstration Guide*, SC23-4862

Online resources

These Web sites are also relevant as further information sources:

- Virtual I/O Server (VIOS)

  `http://techsupport.services.ibm.com/server/vios/home.html`
- IBM Support: Fix Central

  `http://www-912.ibm.com/eserver/support/fixes/fcgui.jsp/`
- IBM AIX 5L Library

  `http://www-03.ibm.com/servers/aix/library/`
- IBM Cluster Information Center (CSM, PSSP, HACMP, RSCT documentation)

  `http://publib.boulder.ibm.com/infocenter/clresctr/vxrx/index.jsp`

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

IBM Redbooks

Virtualization and Clustering Best Practices Using IBM System p Servers

(0.2”spine)
0.17”<->0.473”
90<->249 pages

Virtualization and Clustering Best Practices Using IBM System p Servers

Latest clustering enhancements revealed

Virtualization solutions

Sample scenarios included

This IBM Redbooks publication highlights and demonstrates, through practical examples, clustering technologies, and principles that involve IBM System p servers and various IBM management software. Different viewpoints are exposed and analyzed, to help reveal areas of importance that may be exploited in reducing the total cost of ownership of your IT environment.

This book will help you install, tailor, and configure IBM System p5 and exploit its advanced features, such as Advanced POWER Virtualization, which provides new ways to get more out of your machine and therefore more from your investment.

This book will not tell you which server to buy, or which technology is best for you, or mandate the “right” way to deploy and administer your environment. This book shows you available methods, potential options, and different viewpoints. It also shows you how to get more from your system, and that to get more is not always as complicated as you might expect.

This book shows you that many “right” answers may exist for given problems, and also illustrates points to consider when evaluating potential solutions. We demonstrate what you need to understand, appreciate, and consider when making a choice or when to change a decision. Making the wrong choice may prove more expensive to manage, than the apparently expensive choices you make when purchasing elements for your IT environment.

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

SG24-7349-00 ISBN 0738486175