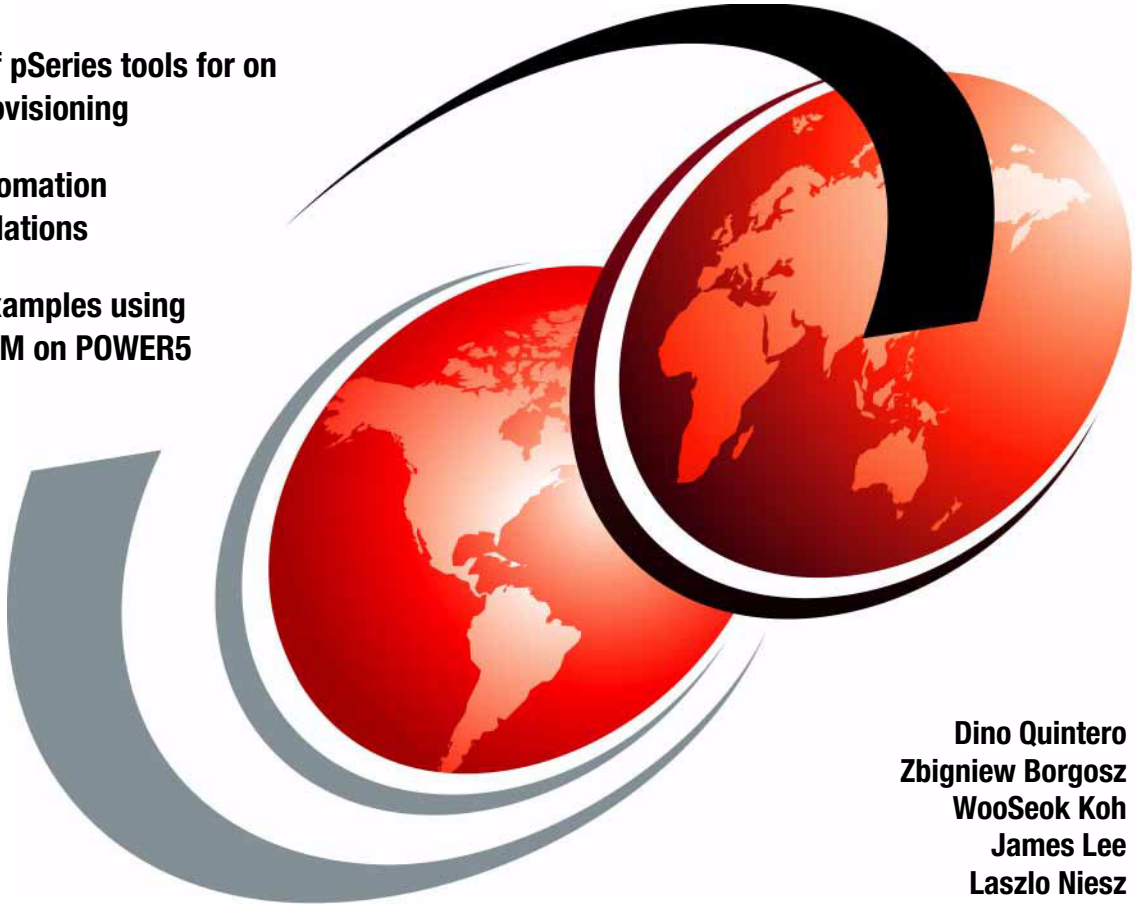
IBM eserver

IBM

Introduction to pSeries Provisioning

- Overview of pSeries tools for on demand provisioning
- pSeries automation recommendations
- Practical examples using CSM and NIM on POWER5



Dino Quintero
Zbigniew Borgosz
WooSeok Koh
James Lee
Laszlo Niesz

Redbooks

ibm.com/redbooks

International Technical Support Organization

Introduction to pSeries Provisioning

November 2004

SG24-6389-00

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

First Edition (November 2004)

This edition applies to AIX 5L Version 5, Release 3, and Cluster Systems Management (CSM) for AIX Version 1, Release 4, Modification 0, Fix 1.

Contents

Figures

Tables

Examples

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:
IBM Director of Licensing, IBM Corporation, North Castle Drive Armonk, NY 10504-1785 U.S.A.

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

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

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

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

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

Trademarks

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

℮server®	DB2 Universal Database™	POWER4™
℮server®	DB2®	POWER5™
Redbooks (logo) ™	Hypervisor™	Redbooks™
^®	HACMP™	RS/6000®
eServer™	IBM®	Tivoli®
iSeries™	LoadLeveler®	TotalStorage®
i5/OS™	Micro-Partitioning™	Virtualization Engine™
pSeries®	NetView®	WebSphere®
AIX 5L™	OS/400®	
AIX®	POWER™	

The following terms are trademarks of other companies:

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

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

Intel, Intel 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, and service names may be trademarks or service marks of others.

Preface

pSeries® provisioning is a term for effectively enabling automation tools to allocate, de-allocate, and re-allocate resources for users, applications, or even functionality within an application, thereby providing the most cost-effective delivery of computing resources to an organization. Provisioning can be a manual process, but there is a point where automation becomes essential. Although automation and integration of the provisioning process is a custom effort for each company, it can involve the scripted provisioning tools and automation becomes more dynamic, and the environment becomes more complex for IT professional. By using provisioning, you can provide users with an environment where resources are dynamically adjusted, given the demands of the business.

This IBM® Redbook summarizes the pSeries provisioning concept. It highlights the IBM on demand business concepts, IBM Tivoli® Provisioning Manager and pSeries automation workflows, pSeries provisioning tools and sample scenarios. It also describes how pSeries provisioning is positioned in an on demand world.

This book describes pSeries provisioning components such as Network Installation Manager (for client installation), RMC (for monitoring), dynamic LPAR (for resource allocation, de-allocation, and re-allocation), HACMP™ (for high availability), AIX® 5L™ (for accounting), CSM (for cluster management), and CUoD (for automated capacity on demand upgrade). In combination with IBM Tivoli Provisioning Manager, they provide tools and workflows for provisioning resources.

The team that wrote this redbook

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

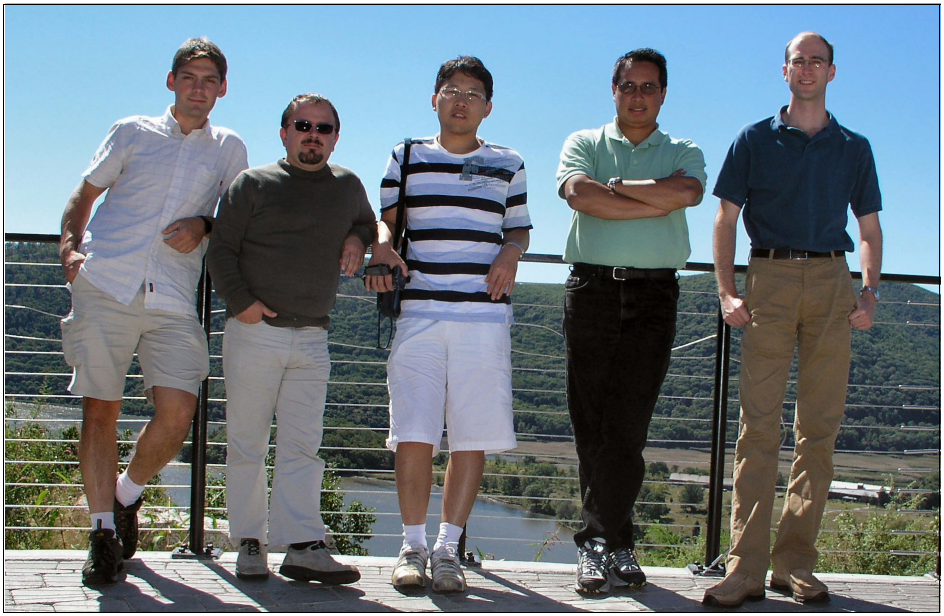
Dino Quintero is a Consulting IT Specialist at the ITSO in Poughkeepsie, New York. Before joining the ITSO, he worked as a Performance Analyst for the Enterprise Systems Group, and as a Disaster Recovery Architect for IBM Global Services. His areas of expertise include disaster recovery and pSeries clustering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

Zbigniew Borgosz is Technology Team Manager and has worked for ComputerLand SA, an IBM Business Partner in Poland since 1998. He is also a System Architect and high-end UNIX® Systems Consultant. His areas of expertise include disaster recovery solutions and advanced UNIX system services (AIX, HP-UX, Solaris™, Linux®). He has co-authored three Redbooks™: *RS/6000 SP Cluster: The Path to Universal Clustering*, SG24-5374; *Managing IBM @server Cluster 1600 Power Recipes for PSSP 3.4*, SG24-6603; and *An Introduction to the New IBM @server pSeries High Performance Switch*, SG24-6978.

WooSeok Koh is an IT Specialist and works at IBM Korea. He has been working with the ITS pSeries Service Team since 2001. He has four years of experience with AIX including RS/6000® and pSeries systems. His main areas of expertise include AIX system maintenance, AIX migration, and the implementation of several cluster solutions.

James Lee is a Product Support Specialist and has worked for IBM UK since 2000. He is a Certified Advanced Technical Expert on pSeries and AIX 5L, and his areas of expertise include AIX and TCP/IP. He has presented material about the new networking features of AIX V5 and Workload Manager. He holds a degree in Philosophy from Cambridge.

Laszlo Niesz is an IT Specialist in IBM Hungary. He has seven years of experience in pSeries clustering. He is a Certified Advanced Technical Expert on AIX. He holds a degree as Computer Programmer from the University of Szeged, Hungary. His areas of expertise include implementation and management of HACMP, and PSSP clusters for Oracle and Tivoli. He has co-authored the redbook *IBM @server Cluster 1600 Managed by PSSP 3.5: What's New*, SG24-6617.



Team members (left to right): Zbigniew Borgosz, Laszlo Niesz, WooSeok Koh, Dino Quintero, James Lee

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

Octavian Lascu, Scott Vetter
International Technical Support Organization, Austin Center

Paul Swiatocha Jr., Shujun Zhou, Les Vigotti, Mike Stancampiano, Mike Schmidt, Ling Gao, Mark Guverich, Elaine Krakower, Yan Koyfman
IBM Poughkeepsie

Kevin Fought, Julie Craft, Ron Goering, J. Scott Sims, Paul Finley
IBM Austin

Become a published author

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

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

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

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

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

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

  ibm.com/redbooks

- Send your comments in an email to:

  redbook@us.ibm.com

- Mail your comments to:

  IBM Corporation, International Technical Support Organization
  Dept. JN9B Building 003 Internal Zip 2834
  11400 Burnet Road
  Austin, Texas 78758-3493

1

Introduction to pSeries provisioning

In this chapter, we start by defining provisioning. We then explain which parts of the provisioning process are covered by this book, and the position of IBM Tivoli Provisioning Manager as the strategic provisioning tool for pSeries.

Additionally, 1.3, “Provisioning on demand” on page 3 looks at provisioning as a whole: what needs to be provisioned; the provisioning process; and the environment required for on demand provisioning.

1.4, “Provisioning and open standards” on page 9, introduces Chapter 8, “pSeries provisioning in an on demand world” on page 157, which looks beyond IBM Tivoli Provisioning Manager to the open source interfaces provided by pSeries for alternative tools.

1.1 What is provisioning?

We define provisioning as the process of providing IT resources to enable business functions to run.

"Provisioning makes the right resources available to the right processes and people at the right time"[1]. In an on demand business, provisioning should have the following characteristics:

- Based on *open* standards
- *Integrated*
- *Autonomic*
- And, the resources should be *virtualized* for maximum flexibility

Chapter 2, "IBM on demand business" on page 11 discusses on demand and its attributes in more detail.

The provisioning goal is to be able to have an application running as quickly as possible from any stage of preparation (from bare metal up) with as little manual intervention as possible.

1.2 What this redbook is all about

pSeries does not provide all the automated tools required to achieve this goal — provisioning. It provides hardware management facilities, such as the HMC (4.1.1, "The Hardware Management Console (HMC)" on page 31), install servers through NIM (4.2.1, "Network Installation Manager (NIM)" on page 37) and clustering technology through CSM and HACMP (4.2.2, "Cluster Systems Management (CSM)" on page 39 and 4.3.5, "HACMP" on page 52). But, there is no single solution for provisioning. Indeed, any single solution is unlikely to be implemented in pSeries alone.

There are two reasons for this: The first is the on demand vision of open standards. Any solution that addressed only pSeries would be unlikely to meet this requirement. More importantly, the increased complexity of IT environments is driving systems management towards integrated, cross-platform, single points of control. From that point of control, we should be able to provision an application on any machine in our environment based on requirements such as performance and availability, irrespective of the hardware on which it is implemented.

The strategic provisioning solution for pSeries is IBM Tivoli Provisioning Manager (ITPM).

[1] http://www.ibm.com/software/info/openenvironment/infrastructure.html

ITPM uses predefined procedures known as workflows to automatically create, install and configure partitions. On pSeries, these partitions can run AIX or Linux. ITPM is described in more detail in Chapter 3, “Tivoli Provisioning Manager” on page 17.

ITPM uses the tools provided by pSeries (the HMC and NIM) to perform these tasks. It therefore relies on a preconfigured environment to provide services so it can provision clients with as little manual intervention as possible. How pSeries can provide this environment, and the place of ITPM within that environment, is the main subject of this redbook and is covered in Chapter 4, “pSeries provisioning tools overview” on page 29, Chapter 5, “General scenario description” on page 61, Chapter 6, “POWER4 provisioning scenario” on page 95 and Chapter 7, “POWER5 provisioning scenario” on page 135.

To explain this further, it is helpful to look at the provisioning process in more detail.

1.3 Provisioning on demand

This section describes provisioning on demand as discussed in these topics:

- 1.3.1, “What do we need to provision?” on page 3.
- 1.3.2, “The provisioning process” on page 5.
- 1.3.3, “The provisioning environment” on page 7.

1.3.1 What do we need to provision?

The first stage in provisioning is to decide what resources are required. This can be viewed as a matrix, mapping customer requirements to the best implementation. An example of such a matrix is given in Figure 1-1 on page 4. In this example, the required response time of our I/O-heavy application affects the number of CPUs, amount of memory and SCSI network and disks, while the operating system backup affects the IP network and tape requirements as well as needing a backup application.

In all such decisions other information becomes pertinent:

- Cost - We want the solution to be as cheap as possible.
- Performance information - To translate requested response times in to required resources.
- Compatibility - Of the hardware and software needed.
- Prerequisites - What middleware and services does our application need?
- Legal requirements - For the storage and security of data, for example.

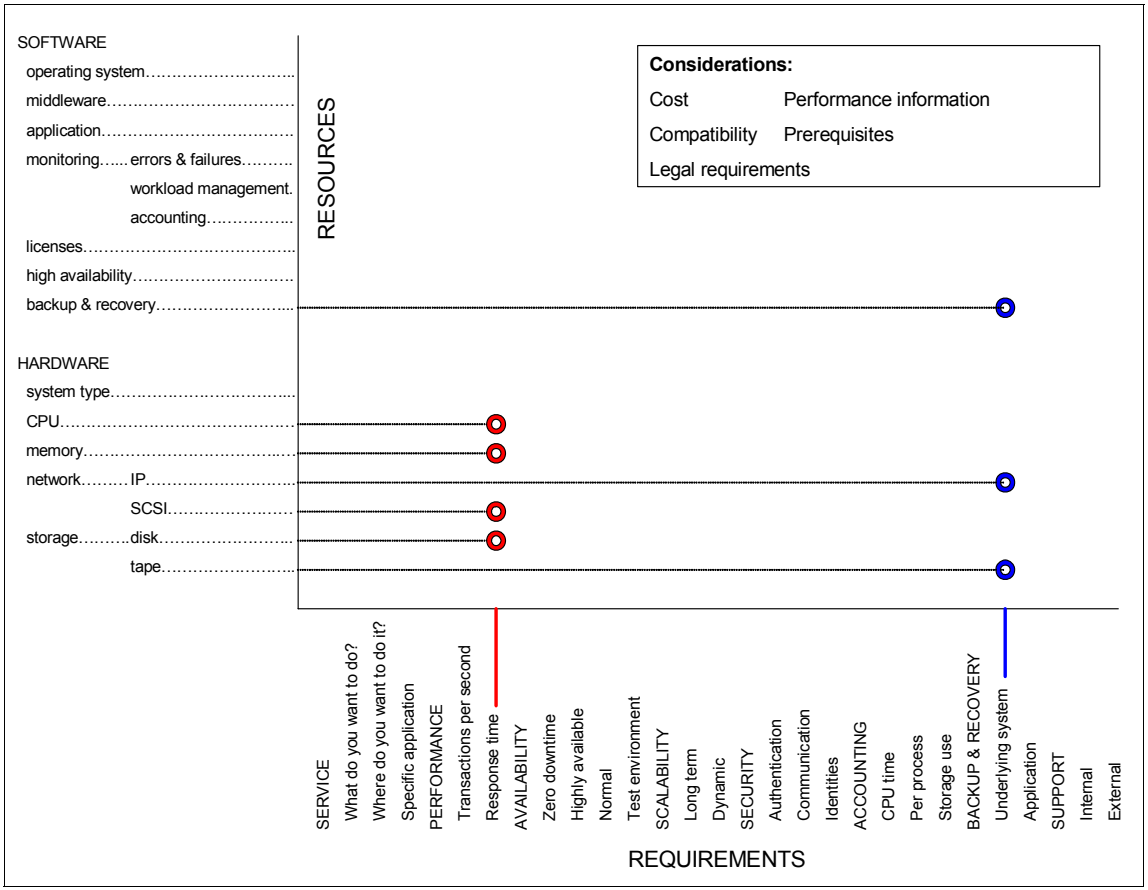


Figure 1-1 The provisioning matrix

Looking in more detail at turning requirements in to a list of resources, we go through the following processes:

- Define hardware, software and network resources for running the application code:

  This step would include gathering requirements for performance, future growth, Service Level Agreements (SLA), virtualization options, storage, versioning and compatibility.

  Ideally, the requests of the application owner would then be turned in to resource requests based on information provided by the provisioning toolset, which itself would be based on performance information, compatibility matrixes and so forth. In practice, this must be done manually.

There are several ways to think about performance requirements. It is possible to ask for a specific processor architecture with given speed, if the application is tested on this type of hardware and the provided performance is well documented. Alternatively the response time is given by the customer for specified actions in the application. Here, the application has to have well defined test results for different processor architectures and environments. In either case, continuous monitoring and dynamic resource allocation once the application is running is an advantage.

- Define backup and recovery requirements.

  Some parts of the application complex could have dependencies on others, necessitating consistent, timely backups of the various components.

- Define the required security level, authentication and authorization information, user identities and identity mapping data.
- Get authentication and authorization to the provisioning tools to define the resources needed, as well as any licenses required.

We should provide as much information as possible to the provisioning tool, and be aware of any default values for those attributes we do not specify.

1.3.2 The provisioning process

Provisioning turns the client definition provided by the previous section in to reality. It requires a number of steps which are often sequential. In ITPM, each step is known as a transition. The process as a whole (or strictly the instructions to run the process) is called a workflow. A logical view of this process is given in Figure 1-2 on page 6.

Workflows are discussed in more detail in Chapter 3, “Tivoli Provisioning Manager” on page 17.

With current technology, it seems unlikely that manual processes can be entirely eliminated. However, virtualization and careful preparation of the provisioning environment can keep physical changes to a minimum.

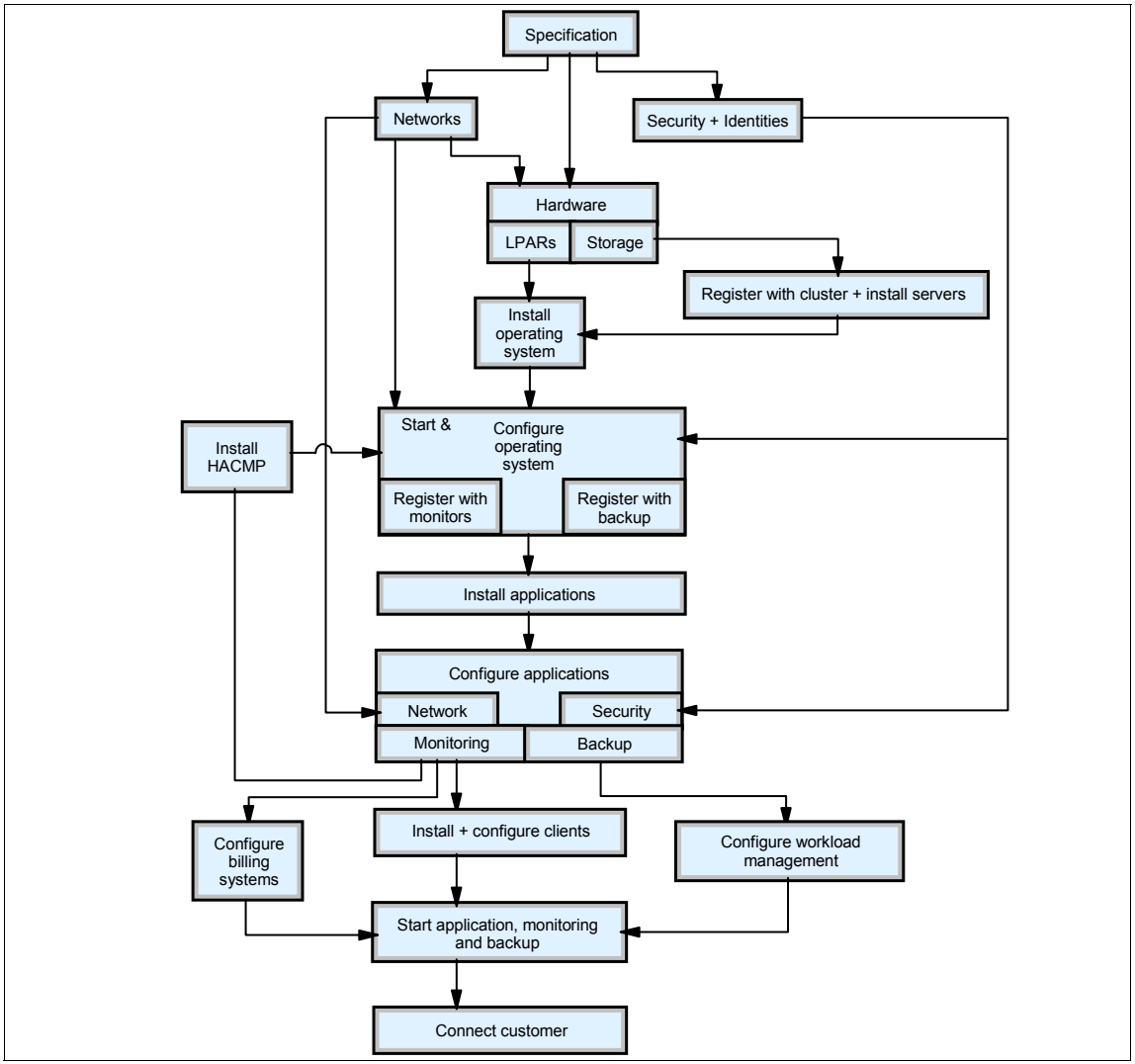


Figure 1-2 The provisioning process

Looking at these steps in more detail, and from a pSeries perspective, we note the following points:

- Network (VLAN) setup using virtualization features:

  Cabling, switch, VLAN and firewall configuration steps could all be needed. Some of these could be simplified by using the Virtual I/O capabilities of pSeries hardware (see Chapter 4, “pSeries provisioning tools overview” on page 29 and Chapter 5, “General scenario description” on page 61).

- LPAR creation:

  Usage of CoD and dynamic LPAR (see Chapter 4, “pSeries provisioning tools overview” on page 29 and Chapter 5, “General scenario description” on page 61) functionality of pSeries servers is suggested.

- Storage setup and allocation:

  Here again a Virtual I/O server could be very useful but it depends on the performance the customer needs. The Virtual I/O server must be set up with a storage pool defined in advance. TotalStorage® tools can simplify LUN allocation from the SAN.

- For NIM installations (which are used by ITPM), or where the machine are part of a cluster, the client must be defined on the server before it can be installed and managed.

- Client network requirements:

  In a multi-layer security environment the server can be in a “build network” until it is fully operational. This is because the production firewall and network environment may not allow some types of communication which are required during installation. If this is the case, additional steps of moving the machine from development to test to production are required. These could use dynamic LPAR to activate different adapters.

- NIM can be configured to run customizing scripts on a client after installation.

- Create and schedule the backup steps needed for each resource defined:

  System backups can be saved into NIM, ready for system recovery if required.

  Application data can be saved with other backup solutions such as Tivoli Storage Manager.

- Start the application:

  Dependencies among the parts of the application environment can require well defined startup and shutdown procedures.

- Start SLA and performance monitoring:

  AIX provides built in accounting and auditing features.

1.3.3 The provisioning environment

The environment must provide a number of resources in order for ITPM to work with pSeries, as shown in Figure 1-3 on page 8.

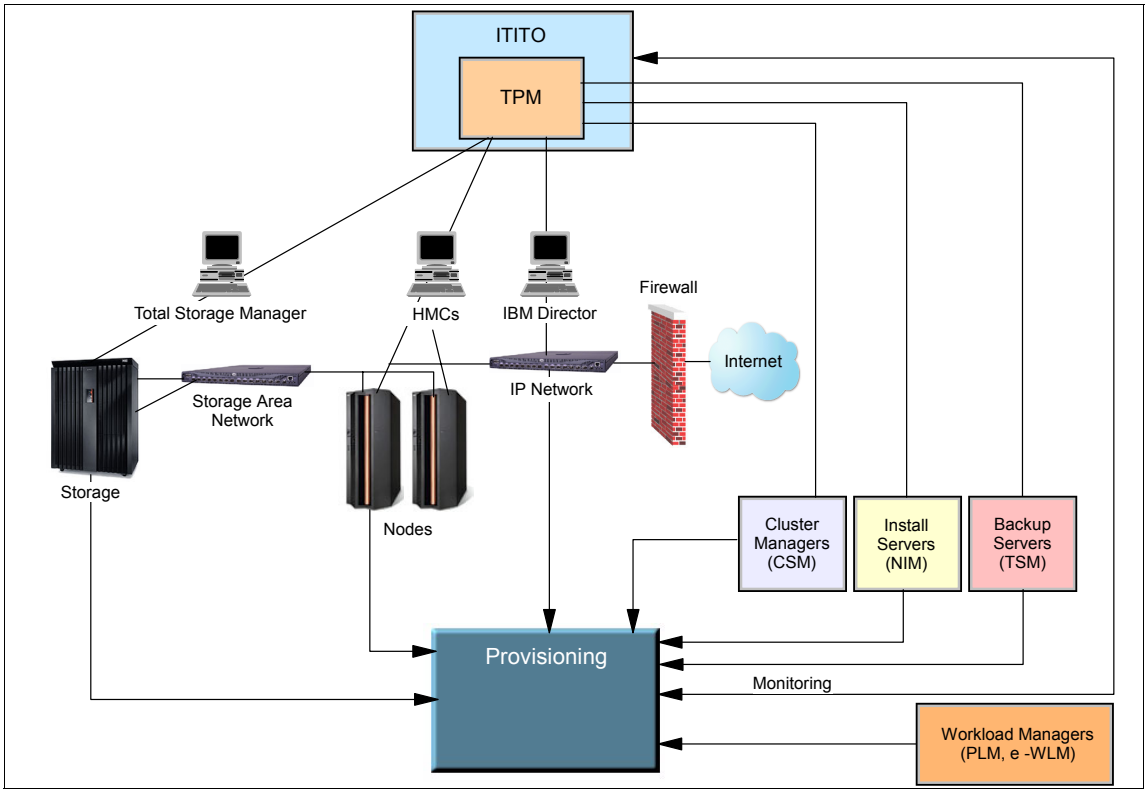


Figure 1-3 The provisioning environment

The creation of the provisioning environment is the main subject of this book, so many of the parts are discussed in detail elsewhere (see Chapter 4, “pSeries provisioning tools overview” on page 29 and Chapter 5, “General scenario description” on page 61). Chapter 6, “POWER4 provisioning scenario” on page 95 and Chapter 7, “POWER5 provisioning scenario” on page 135 provide detailed scenarios where we created a provisioning environment for POWER4™ and POWER5™ systems.

Some minor points can be made here:

- Secure connections must be available from ITPM to each of the service providers.
- Although not shown in the diagram, many environments are partitioned in to development, test and production segments, each with their own security setup.
- The application code must be available to the server responsible for the install (known by ITPM as a “boot server”).

This seems simple, but it is very dependent on the application. Some applications span several CDs and require other middleware to be installed first. A common prerequisite is database management software. Licenses must also be managed.

Predefined application combinations could be tested in advance as part of the provisioning environment setup.

- Switch zoning and the use of virtual LANs can separate the logical layout above from its physical implementation, providing further virtualization of resources. The initial network design should take likely future expansion in to account.
- Because many resources must be allocated before they can be used, the resource allocation and implementation tools must interconnect and respond to changes in each other. The provisioning process should be flexible enough to provide similar solutions (in terms of performance, security and so forth) on different types of hardware (through different workflows).
- Where manual intervention is needed this should be recorded, with the action taken. All the parameters given should be stored separately from the commands. This would allow the integration of these manual steps with the automated provisioning.

Ideally, machines would register themselves as resources in a standard manner. They would then be dynamically added to the server's list of available services, becoming part of a dynamic, extendable environment. The provisioning tools would then provide an up-to-date view of the free and allocated resources. This kind of environment is known as a service oriented architecture, and is discussed in Chapter 8, "pSeries provisioning in an on demand world" on page 157.

1.4 Provisioning and open standards

Today's highly distributed and heterogeneous application environments need a solution for common management interfaces and centralized control points. Several parallel developments are in progress for different applications and resource types and communication between the separate teams is improving over time.

One of the most important steps in this joint effort is the use of standard Web services within GRID computing. Another example is the development and implementation of the common information model for storage resources. After an initial development by Storage Networking Industry Association (SNIA) this was moved under the control of Desktop Management Task Force (DMTF). It has now been expanded for use in the management of a broader range of computer resources.

In Chapter 8, “pSeries provisioning in an on demand world” on page 157 we show ways in which pSeries provisioning can be integrated into this open software based world.

2

IBM on demand business

This chapter provides a brief summary of the IBM on demand business initiatives and how provisioning plays a role in the overall business strategy, and also discusses the on demand business operating environment.

2.1 On demand business operating environment

What is an on demand business operation environment? Is is not a specific set of hardware and software. Rather, it is an environment that supports the needs of the customer allowing it to become and remain responsive, variable, focused, and resilient.

An on demand business operating environment unlocks the value within the IT infrastructure to be applied to solving business problems. It is an integrated platform, based on open standards that enable rapid deployment and integration of business applications and processes. Combined with an environment that allows true virtualization and automation of its infrastructure, it enables delivery of IT capability on demand.

An on demand business operating environment must be:

- Flexible
- Self-managing
- Scalable
- Economical
- Resilient
- Based on open standards

The value of the operating environment is in the ability to dynamically link business processes and policies with the allocation of IT resources using offerings across all of these categories. In the operating environment, resources are allocated and managed without intervention, enabling resources to be used efficiently based on business requirements. Having a flexible, dynamic business processes increases the ability to grow and manage change within the business as shown in Figure 2-1 on page 13.

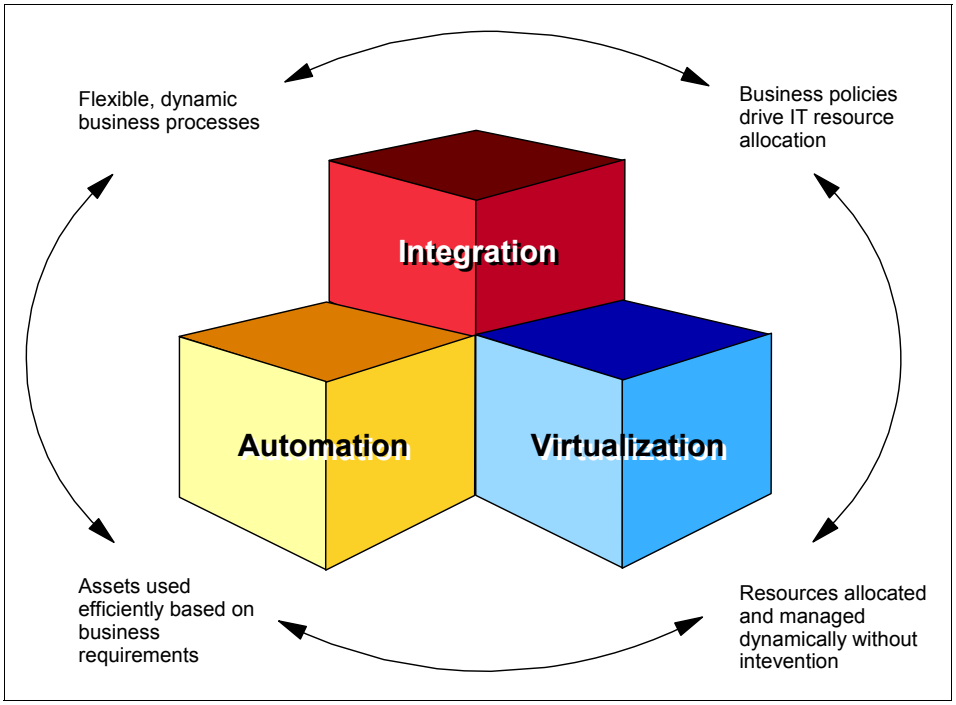


Figure 2-1 IBM on demand business environment overview

Figure 2-2 on page 14 provides an overview of the key components of an on demand operating environment.

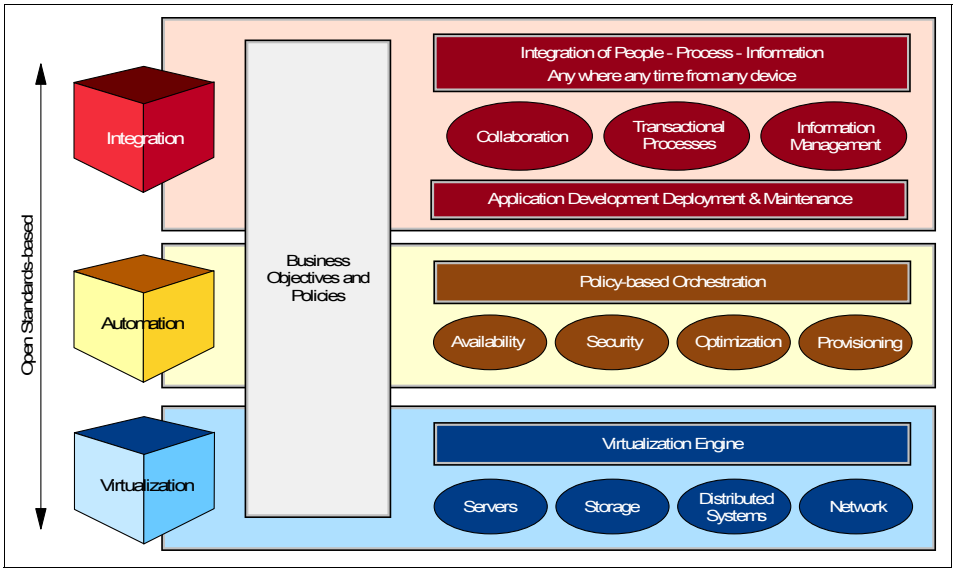


Figure 2-2 Components of an on demand business environment

Figure 2-2 and Figure 2-3 on page 15 show that virtualization and provisioning are key elements of the automation component of an IBM on demand business operating environment. IBM has products to support these key elements. However, IBM Tivoli Provisioning Manager is the strategic product enabled to provide the necessary integration of all the components for an on demand business environment.

Horizontal business processes are unique to each business. Integrating and automating these processes cannot be done by purchasing packaged applications. It is a custom effort for each individual company.

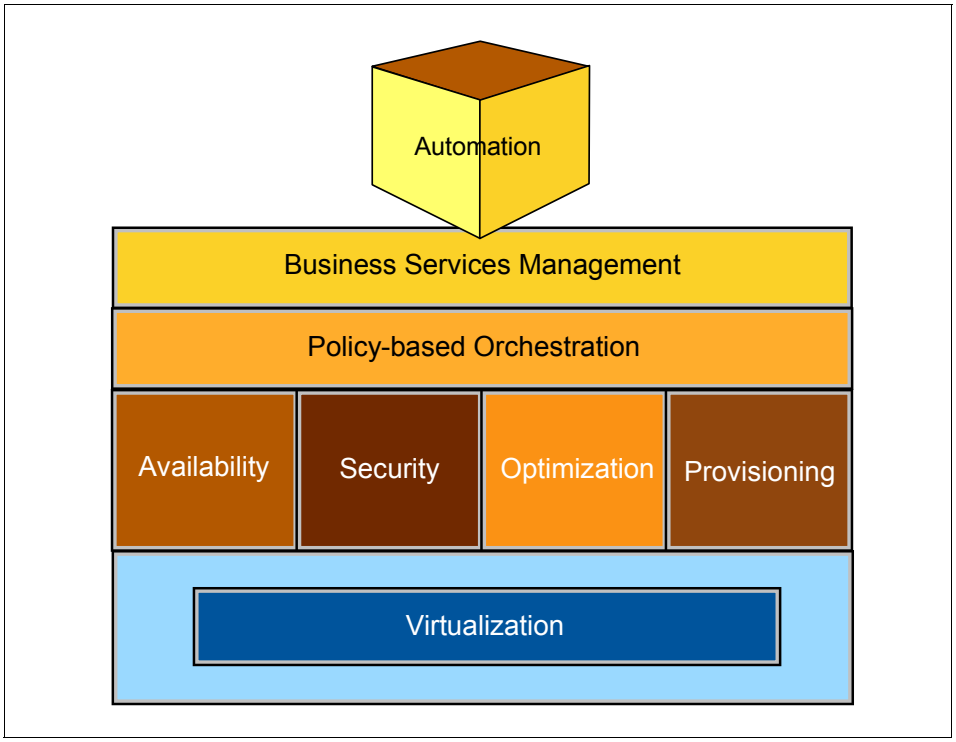


Figure 2-3 IBM automation components

pSeries provides additional provisioning tools designed to complement IBM Tivoli Provisioning Manager. These tools are broken down into two areas: hardware provisioning tools (POWER5, POWER4, Hardware Management Console (HMC), dynamic Logical Partitioning (dynamic LPAR), micro-partitioning, virtual I/O, etc.), and software provisioning tools (AIX 5L V5.3, Network Installation Manager (NIM), Cluster Systems Manager (CSM), WorkLoad Manager (WLM), Partition Load Manager (PLM), Virtualization Engine™, etc.).

Although pSeries provides tools for provisioning to work with IBM Tivoli Provisioning Manager, ultimately it is how these tools are used as building blocks in an integrated environment that is important. Refer to Chapter 6, “POWER4 provisioning scenario” on page 95, and Chapter 7, “POWER5 provisioning scenario” on page 135 for sample scenarios on how the pSeries provisioning tools are used as building blocks for an on demand business environment.

For detailed information about the pSeries provisioning tools, refer to 4.1, “Hardware provisioning tools” on page 30, and 4.2, “Software provisioning tools” on page 35.

Chapter 3, “Tivoli Provisioning Manager” on page 17, provides more details on how this component is positioned to provide on demand business automation.

3

Tivoli Provisioning Manager

The IBM @server® pSeries and AIX 5L provide the basic building blocks for an on demand provisioning environment. The missing element is to implement the connections between these blocks and automate the provisioning process. After the required environment is built, we start the day to day administration to keep it available by providing optimal performance with the lowest non utilized resources possible.

To achieve this level of automation IBM provides IBM Tivoli Intelligent ThinkDynamic Orchestrator (ITITO), and IBM Tivoli Provisioning Manager (ITPM).

In this chapter, we describe some aspects of IBM Tivoli Provisioning Manager within these topics:

3.1 High level architecture

IBM Tivoli Intelligent ThinkDynamic Orchestrator (ITITO) includes IBM Tivoli Provisioning Manager (ITPM), a standalone product that can be purchased separately, based on your data center needs. Tivoli Provisioning Manager provides core automated deployment capability, while Tivoli Intelligent ThinkDynamic Orchestrator adds policy-based decision-making capabilities.

See Figure 3-1 for the high level architecture of ITITO.

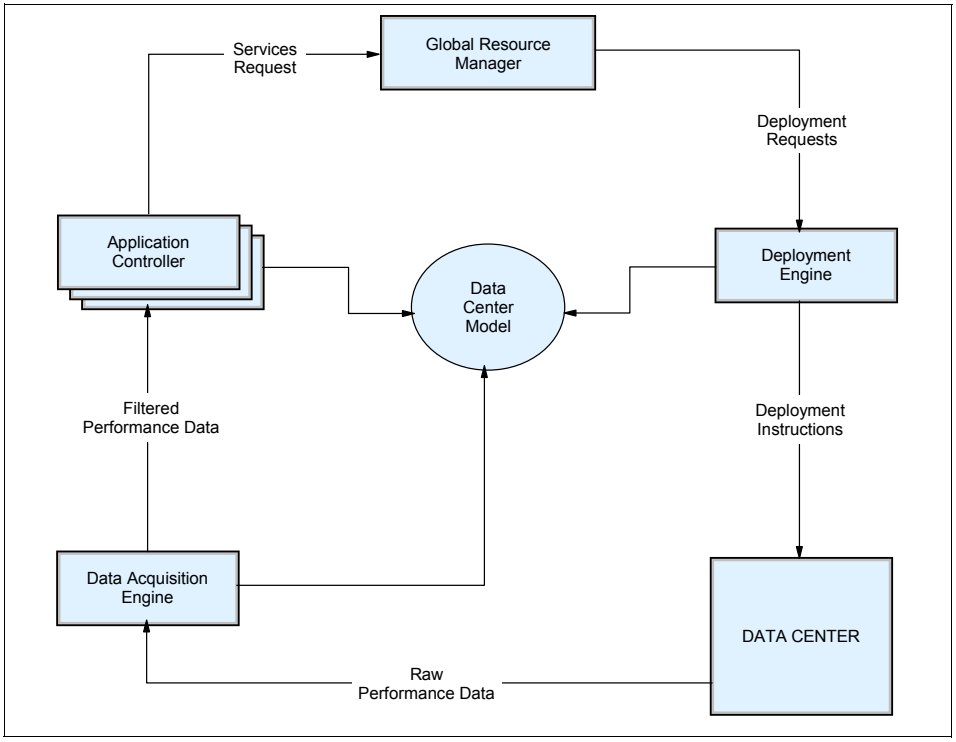


Figure 3-1 ITITO Architecture

For better understanding, we describe the terms used in the context of IBM Tivoli Intelligent ThinkDynamic Orchestrator, and IBM Tivoli Provisioning Manager:

- Data acquisition engine

  Collects and preprocess performance metric data from the managed environment and in doing so, it participates in the information gathering task. It works via drivers which are specific for each resource type.

- Application controller

  It determines the resource requirements based on real-time performance data and predictions defined for an application environments.

- Global resource manager

  Receives requirements from all application controllers and manages the overall optimization of the data center by:

  - Making optimal resource allocation decisions
  - Ensuring a stable control over the application infrastructure

  In the case of new server allocation it decides the resources the server will use.

- Data center model

  The data center model is a representation of all of the physical and logical assets under ITPM and ITITO management.

- Deployment engine

  Responsible for managing the workflows which automate the allocation and configuration of assets.

The deployment engine component manages the provisioning process, and it is the most important part of the Tivoli Provisioning Manager product.

Figure 3-2 on page 20 shows the architecture of deployment engine.

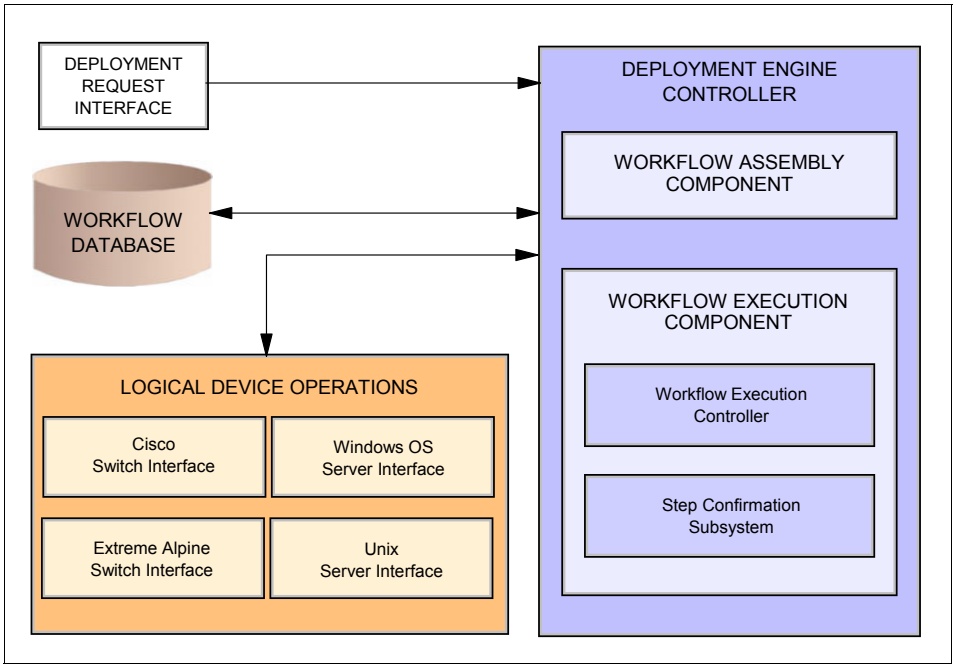


Figure 3-2 Deployment engine architecture

The deployment engine provides an interface via the assembly and execution of the workflows are initiated. It provides a controlled execution of the workflow steps which enables parallelism, and can be based on defined conditions.

This next list provides information about how the workflows build up from smaller elements:

- Workflow

  A workflow is a sequentially executed series of steps that performs a particular task. A step in a workflow is called a transition. Each workflow has a single compensating workflow that is executed if any transition fails.

- Workflow or automation package

  Also referred to as TC driver, device driver, or simply driver, a workflow package is a collection (or a container) of commands, shell scripts, workflows, logical operations, and Java™ plug-ins that applies to the operation of one specific type of software component or a physical device. It contains a group of tasks that corresponds to a physical or logical device. These tasks typically implement logical operations. A device could be a specific piece of hardware, an operating system, a service, or a cluster.

These resources are packaged into a single file with the *.tcdriver* extension. The IBM Automation Package for AIX NIM installation is an example of available workflow packages.

- Transition

  A transition is a step in a workflow. This could be another workflow, a logical operation, a simple command, or a Java plug-in.

- Logical operation

  A logical operation is a task that is abstracted from its actual implementation. An example in a data center is the task of adding or changing an IP address on a server. A logical operation makes no assumptions about the implementation such that the complexity of adding or changing an IP address is hidden from the operator or workflow. For example, in Tivoli Provisioning Manager and Tivoli Intelligent Orchestrator, a single logical operation adds or changes the server IP address on Linux, AIX, OS/400®, zVM, Windows®, HP-UX or Solaris. Logical operations are implemented via Enterprise Java Beans (EJB™).

- Simple command

  A simple command is a wrapper for a Java plug-in. It describes the plug-in's input and output requirements via input and output variables. A simple command performs an action in the data center, such as installing an application on a server, saving the configuration of a switch, and so on.

- Java plug-in

  A Java plug-in is the Java class that contains the interface or protocol code that interacts with the devices in the data center.

- Service access point (SAP)

  A Service access point (SAP) is a definition of the protocol and credentials used by or associated with an asset. An asset can have more than one SAP.

Each managed object in a ITPM environment has to have a customized Data Center Model (DCM) type. This customization has to be done to represent the unique attributes and operations a specific managed object provides. The adaptation is done by adding variables to DCM types, and developing workflows to implement their operations.

3.2 Workflows

The workflows are controlled by the deployment engine. Java plug-ins provide the interface for interaction with data center objects. Every action that is

performed in the data center is implemented by the deployment engine using a Java plug-in.

Java plug-ins are implemented by Enterprise Java Beans (EJB). This means that the versatility of the deployment engine is limited to the versatility of the (external) commands available through the EJB implementation.

Note: It is possible to program new Java plug-ins to enhance functionality.

For detailed information about ITITO and ITPM see the official documentation and also the redbook: *Provisioning On Demand Introducing IBM Tivoli Intelligent ThinkDynamic Orchestrator,* SG24-8888.

ITPM comes with predefined best practices for standard products from all major infrastructure vendors, and provides an environment where administrators can create new workflows to automate provisioning steps. The workflow transitions are using the existing product provisioning capabilities they are executed on.

The on demand automation catalog has a repository of many workflow packages what can be used as is or as a base for future development of customized workflows. You can check the catalog at:

`http://www-18.lotus.com/wps/portal/automation`

Figure 3-3 on page 23 shows the search result for AIX workflows on the IBM on demand automation catalog Web site.

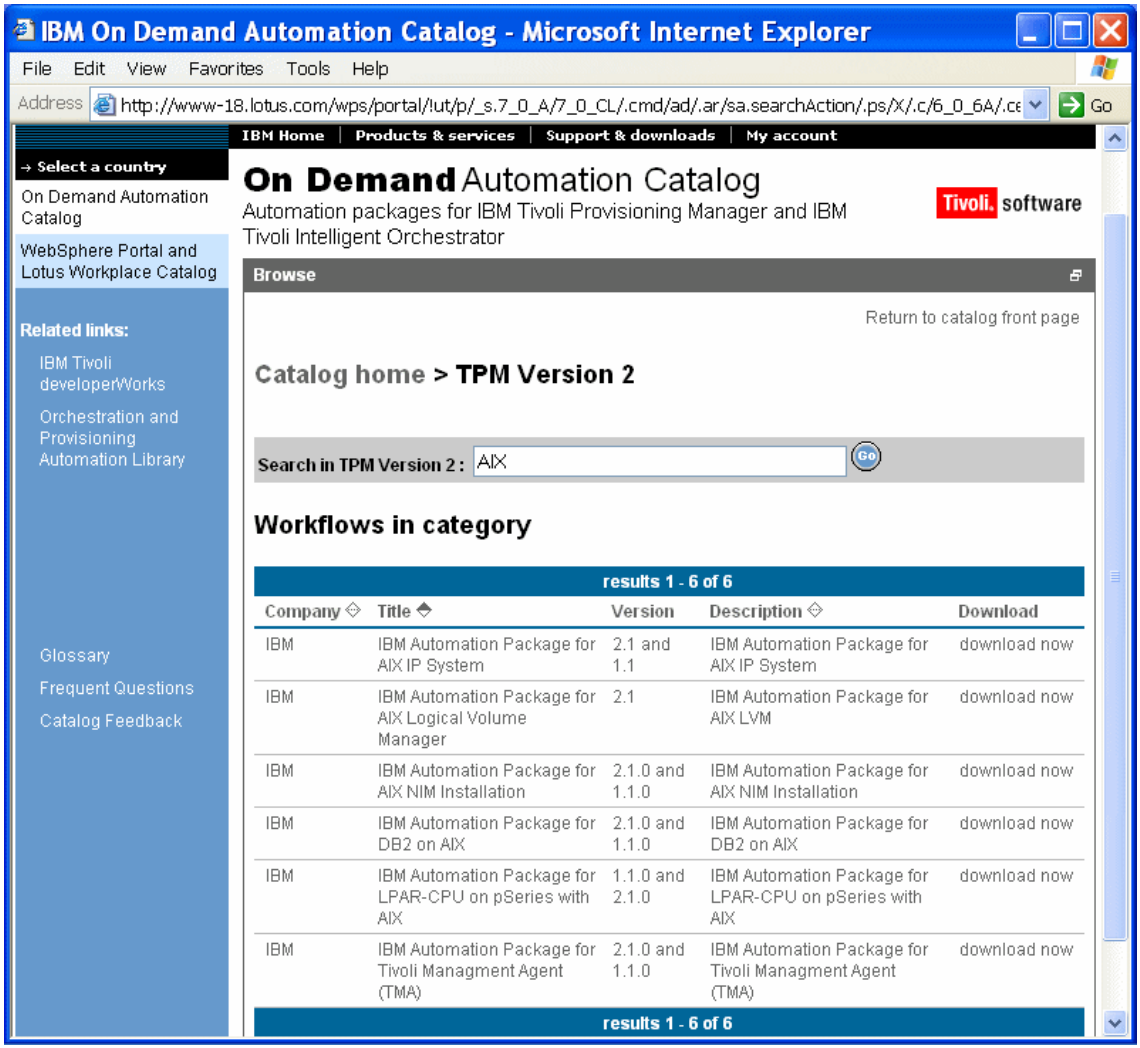


Figure 3-3 Search result for AIX related workflows

Each workflow package has an extensive documentation about the functionalities provided, and about the installation and operation.

In the following sections, we describe some of the workflows related to pSeries provisioning:

IBM Automation Package for AIX NIM installation

This package provides IBM AIX NIM installation support for pSeries servers.

Tivoli Provisioning Manager can utilize NIM to manage the installation of the AIX operating system on IBM pSeries hardware. It is possible to install the entire operating system or an individual application from the Tivoli Provisioning Manager through AIX NIM.

The NIM environment must be configured and the NIM master must be accessible by the Tivoli Provisioning Manager server. The NIM master must be installed with AIX version 5.1 or greater.

The following are necessary DCM object definitions to be made in ITPM before using the workflow:

- Bootserver: represents the NIM master, with the password for the root user and the IP address of the NIM master.
- Service access point (SAP): provides the connection to an HMC by defining the user and the IP address which can be used to connect to it.
- Server: DCM objects with different variable set for standalone, HMC managed LPAR, HMC managed POWER4.
- Software stack: DCM object definition for mksysb installs, containing mksysb, SPOT, bosinst_data NIM resources.
- Software product: DCM object for application install, naming the necessary lpp_source and fileset resources for NIM install.

This workflow implements the following operations:

- mksysb install of a pSeries machine or LPAR
- Application install or uninstall
- NIM client definition
- mksysb NIM resource creation on a pSeries machine or LPAR

Above all, the package provides an inventory export tool which can be used to export information from an HMC database for population of the DCM.

Possible extension of this package could be to handle NIM script resources which would allow further customization of an installed server or LPAR.

AIX Operating System Orchestration

This packages provides basic IP configuration and reboot commands operations. Secure remote command execution and file transfer capability has to be configured to the target server.

This package can be used in conjunction with the NIM package to configure a newly installed machine or LPAR to the network.

Necessary DCM object definitions to be made in ITPM before using the workflow:

- Server: Object for the pSeries machine or LPAR to manage
- Subnet: Object definition for a subnet
- Cluster or resource pool: Contains server objects

This workflow implements the following operations:

- Asynchronous or synchronous reboot of an operating system
- Apply a routing table which were attached to a DCM cluster or resource pool
- Add an IP address to an interface: this will allocate an alias address if an IP address is already configured for the interface
- Remove IP address from an interface: removal of IP address or alias
- Implement a routing table for a server

This package can be used as a template for other administration activities.

pSeries server orchestration

This package is a management interface to pSeries LPARs. It is possible to create, delete, activate and deactivate partitions in POWER4 and POWER5 systems. Dynamic LPAR operations are also provided by this package.

This workflow is utilizing the defined HMC server to the LPAR and dynamic LPAR operations. As such it needs a secure remote command execution and file transfer capability configured between the ITPM server and the HMC.

Necessary DCM object definitions to be made in ITPM before using the workflow:

- Service access point (SAP): provides the connection to an HMC by defining the user and the IP address which can be used to connect to it
- Server: DCM objects with different variable set for standalone, HMC managed LPAR, HMC managed POWER4 or POWER5
- Virtual server template: contains predefined packages of possible resources for an LPAR, like CPU, memory and network interface and other I/O slots (called generic)

This workflow implements the following operations:

- Activate an LPAR, by name or by DCM device ID
- Deactivate an LPAR, by name or DCM device ID
- Create an LPAR using a virtual server template
- Remove an LPAR, and instruct the HMC to set the resources allocated to the LPAR as free

- Change the amount of resources allocated to an LPAR: this interface utilize dynamic LPAR operations on the specified dynamic LPAR capable LPAR
- Power on or off LPARs or POWER4/POWER5 machines

Important: At the time of writing, this workflow cannot handle virtual I/O server provided resources.

It is possible to create a new profile which integrates the inventory discovery from the NIM workflow with the LPAR creation and then NIM install of the machine. Some manual configuration is needed as the sufficient resources have to be chosen to create a virtual server template as an input for the LPAR creation operation. The physical and logical network configuration is a prerequisite for all of these operations. As more and more workflows are developed not only for pSeries hardware and software management, we can already find and integrate some of them. This can raise a higher automation level in our provisioning environment.

3.3 Prerequisites for pSeries provisioning

From the description and the workflows we can see that the low level tools which enable pSeries provisioning like networks, NIM, management server (MS), HMC, and LPAR definitions have to be build and configured before the workflows could utilize them. Advanced virtualization features of POWER5 based pSeries servers will be enabled in future versions. All the information about these should be defined in the data center.

Security

As most of the present workflows are using the existing capabilities of CSM, NIM, AIX and HMC via remote command interfaces a secure connection is a must between the ITPM server and the provisioning building blocks provided by pSeries servers. The preferred way is the usage of ssh an open standard secure remote execution and file transfer protocol.

Workflow specific

Each workflow can define special requirement which is documented in the package itself.

Tivoli Provisioning Manager installation involves the installation of several other components. For a detailed description and prerequisites, refer to *Tivoli Provisioning Manager Install Guide,* GC32-1615-00.

3.4 Product packaging

The IBM Tivoli Provisioning Manager package contains the following software products:

- IBM Tivoli Provisioning Manager
- Tivoli GUID
- NetView® Server
- IBM DB2® Universal Database™
- Tivoli Directory Server
- WebSphere® Application Server
- Fixpacks for WebSphere, DB2 and Directory Server
- Application client code for DB2 and WebSphere

Secure communication path must be configured between the elements of this complex containing the base pSeries provisioning tools and the parts of the Tivoli Provisioning Manager package.

See the official install guide for further prerequisites and installation steps: *Tivoli Provisioning Manager Install Guide,* GC32-1615-00.

4

pSeries provisioning tools overview

In this chapter, we introduce the provisioning tools provided by pSeries. All of these tools are covered in greater depth in other publications. The aim here is to give a general overview in 4.1, “Hardware provisioning tools” on page 30 and 4.2, “Software provisioning tools” on page 35, and to give pointers to more detailed documentation.

In 4.3, “Comparison of the tools available” on page 44, we also compare and contrast these tools by looking at their advantages and disadvantages in particular scenarios.

4.1 Hardware provisioning tools

The following sections briefly describe the pSeries hardware provisioning tools. For more detailed documentation, refer to Table 4-1.

The hardware provisioning tools are described in these topics:

- 4.1.1, “The Hardware Management Console (HMC)” on page 31.
- 4.1.2, “Dynamic Logical Partitions” on page 32.
- 4.1.3, “Micro-partitioning” on page 32.
- 4.1.4, “Virtual I/O (VIO)” on page 33.
- 4.1.5, “Capacity on Demand (CoD)” on page 34.
- 4.1.6, “IBM TotalStorage Productivity Center with Advanced Provisioning” on page 35.

Note: In Table 4-1, Redbooks are available from:

www.redbooks.ibm.com

Other documentation is available from the AIX and pSeries information site unless otherwise noted:

http://publib16.boulder.ibm.com/pseries/en_US/infocenter/base

Table 4-1 Hardware provisioning reference documentation

Tool	References
Hardware Management Console (HMC)	► *IBM @server Hardware Management Console for pSeries Installation and Operations Guide,* SA38-0590 ► *IBM @server Hardware Management Console for pSeries Maintenance Guide,* SA38-0603 ► *IBM @server pSeries 670 and pSeries 690 System Handbook,* SG24-7040-02 ► *The Complete Partitioning Guide for IBM @server pSeries Servers,* SG24-7039-01 ► *What is a Hardware Management Console (HMC)?*, Technote at: http://www.redbooks.ibm.com/abstracts/tips0280.html?Open http://techsupport.services.ibm.com/server/hmc

Tool	References
Dynamic Logical Partitioning	► *The Complete Partitioning Guide for IBM @server pSeries Servers,* SG24-7039-01 ► *IBM @server pSeries 670 and pSeries 690 System Handbook,* SG24-7040-02 ► *AIX 5L Differences Guide, Version 5.2 Edition,* SG24-5765-02 ► *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00 for information about the **cpupstat** command. ► *pSeries - dynamic LPAR Scripts*. Technote at: http://www.redbooks.ibm.com/abstracts/tips0121.html?Open ► Other samples can be found in /usr/samples/dr/scripts on AIX 5.2 and 5.3 machines.
Micro-partitioning	► *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00 ► *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, SG24-7940-00 ► *Advanced POWER Virtualization on IBM @server p5 Servers Architecture and Performance Considerations*, SG24-5768-00
Virtual I/O	► *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00 ► *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, SG24-7940-00 ► *Advanced POWER Virtualization on IBM @server p5 Servers Architecture and Performance Considerations*, SG24-5768-00
Capacity on Demand	► *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, SG24-7940-00 ► *Working with Capacity on Demand*, pdf from http://www-1.ibm.com/servers/eserver/pseries/ondemand/cod
SAN Volume Controller and other storage products	► *Exploring Storage Management Efficiencies and Provisioning,* SG24-6373-00

4.1.1 The Hardware Management Console (HMC)

The HMC can provision CPUs, memory, and I/O slots from managed servers. Version 4 can also control iSeries™ machines.

The HMC is a Linux-based “black box” control point that allows the user to define, start, and add (or remove) resources to, logical partitions. It also provides a number of other system management options, including console sessions on managed LPARs. Although the primary interface is graphical, it is possible to script add and remove (dynamic LPAR) operations.

The HMC is a separately orderable feature. It is required if you want to use LPARs, CoD, or clustering.

4.1.2 Dynamic Logical Partitions

Dynamic LPAR is the ability to add resources to, or remove resources from, a partition while that partition is running. It requires AIX 5.2 or higher; October 2002 system firmware or later; and HMC release 3 version 1 or higher. Dynamic LPAR can be controlled from the HMC GUI or through scripts. The scripts are administered with the **drmgr** command and can be written in any programming language. Alphaworks provide a dynamic LPAR Tool Set for pSeries, (see http://www.alphaworks.ibm.com/tech/dlpar and 6.3.2, "Automated dynamic LPAR" on page 120) or you can write your own.

Note: Dynamic LPAR operations can be performed on CPUs, memory and I/O slots.

The granularity of dynamic LPAR is:

- 1 CPU for a dedicated partition, or 0.01 CPUs for a shared processor partition (other features of shared processor partitions not mentioned here can also be changed dynamically).
- 16 Mb of memory (AIX 5.3) or 256 Mb (AIX 5.2).
- One PCI slot with a PCI adapter (including all the ports on that adapter).

Dynamic LPAR is a standard pSeries offering.

4.1.3 Micro-partitioning

Micro-partitioning is the ability to allocate fractions of a CPU to an LPAR. Each LPAR must have a minimum of 0.1 CPUs, while CPU resource can be allocated in intervals of 0.01 CPUs. Partitions using Micro-partitioning technology are referred to as shared processor partitions (SPLPARs). An LPAR must be created as either a dedicated or shared processor partition, and you cannot mix shared and dedicated processors in one partition. However, you can create a shared processor partition with, for example, an entitlement of 2.25 CPUs. All micro-partitions use CPUs from a single, shared pool.

Note:

- Micro-partitioning only applies to CPUs.
- Each partition must still have dedicated memory, but can use virtual I/O.
- To use micro-partitioning, you must purchase the advanced POWER™ virtualization feature.

4.1.4 Virtual I/O (VIO)

VIO provides virtual ethernet and SCSI adapters to client LPARs. These connections can then be routed through a physical adapter by a VIO server. VIO connections are implemented in firmware.

For client LPARs, virtual I/O adapters are part of the LPAR definition. Configuration information for the virtual adapters is then presented to the operating system by the firmware when the LPAR is booted. Clients can run AIX (5.3) or Linux (SUSE Enterprise Server 9 for POWER or Red Hat Enterprise Linux AS for POWER Version 3).

The VIO server is a single-function LPAR. It provides the virtual SCSI server and shared ethernet adapters and is not designed to run applications or other workloads. It is controlled through a restricted, scriptable command line user interface.

With the exception of virtual ethernet communication between partitions within the same machine, you must purchase the advanced POWER virtualization feature to use VIO. VIO is only available on POWER5.

Virtual ethernet

Client LPARs use a virtual ethernet adapter provided by the firmware to communicate with other LPARs in the same machine. These adapters can be part of the initial definition, or added dynamically through the HMC. Virtual ethernet adapters can be used as boot devices (see 7.3.2, “Step 2. Create the virtual Ethernet device” on page 147).

If a VIO server is present, then the virtual ethernet client adapter can be allocated a share in a (physical) shared ethernet adapter on the VIO server, which acts as a layer 2 switch between internal and external networks. Alternatively, an LPAR with a physical adapter can act as a router, with virtual ethernet connections used only between LPARs on the same machine.

To improve performance, the VIO server can have multiple physical adapters. These can be configured either as separate interfaces, or as a single interface

through link aggregation (EtherChannel or IEEE802.3ad). Both aggregation protocols provide the ability to configure a backup adapter connected to a different switch.

Virtual ethernet is available without the advanced POWER virtualization feature, as long as the machine is attached to an HMC. The standard firmware implements a virtual ethernet switch. However, this only enables communication within the machine, as the advanced feature is required in order to create a shared ethernet adapter.

Virtual SCSI

To a client operating system, a virtual SCSI adapter looks no different to a real adapter. Virtual disks can be used as boot devices.

Physical disks owned by the VIO server can be assigned to client LPARs as a whole, or divided in to multiple logical disks first. To make such a disk available to a client partition it is first assigned to a virtual SCSI server adapter in the VIO server. Clients then use a virtual SCSI client adapter to access the hdisks presented to them. This provides a control point, while data is sent directly from a buffer on the client to the physical adapter owned by the VIO server, through secure direct memory access.

Virtual SCSI devices can be assigned and removed dynamically. It is possible to clone a running system using **`alt_disk_install -C -O`**, then move the virtual disk to another partition and use it as a boot device.

Virtual SCSI requires the presence of a VIO server, and can therefore only be used with the advanced POWER virtualization feature.

4.1.5 Capacity on Demand (CoD)

CoD provisions CPUs and memory. It enables you to activate (and, depending on the CoD option purchased, deactivate) unused CPUs and memory in a machine. These can then be allocated as required, and are charged for on a usage basis. The activation process can be either manual or automatic (known as reserve CoD), and permanent or temporary. Where CoD is used to meet short-term peaks in processing demand, the minimum charge is one “processor day,” which provides one CPU for 24 hours. Multiple CPUs can be provisioned in this way.

Unallocated CPUs and memory, whether active or inactive, are also used to replace failing parts. This process is transparent to the operating systems running on the failing hardware.

Attention: Enabling CoD is a chargeable feature.

4.1.6 IBM TotalStorage Productivity Center with Advanced Provisioning

IBM TotalStorage Productivity Center with Advanced Provisioning is an integrated storage capacity provisioning solution from the IBM TotalStorage Open Software family. At the time of writing, this is the first member of this product group. It can help to reduce the cost and effort of provisioning storage capacity in the enterprise environment while improving availability. It is designed to enable an agile storage infrastructure to respond to changing business needs.

The product package uses open standards such as Web services and CIM object manager for communication and management purposes between its elements.

Note: The TotalStorage Productivity Center is a separate licensed product.

4.2 Software provisioning tools

The following sections briefly describe the pSeries software provisioning tools. For more detailed documentation, refer to Table 4-2 on page 36. These documents are available from the same Web sites documented in 4.1, “Hardware provisioning tools” on page 30.

The software provisioning tools are discussed in these topics:

- 4.2.1, “Network Installation Manager (NIM)” on page 37.
- 4.2.2, “Cluster Systems Management (CSM)” on page 39.
- 4.2.3, “WorkLoad Manager (WLM)” on page 40.
- 4.2.4, “Partition Load Manager (PLM)” on page 41.
- 4.2.5, “The Virtualization Engine” on page 41.
- 4.2.6, “High Availability Cluster Multi-Processing (HACMP)” on page 44.
- 4.2.7, “Service Update Management Assistant (SUMA)” on page 44.

There are a number of products which blur the boundaries of what is and what is not provisioning. These are applications which rely on a fixed pool of resources, but ensure that enough of these resources are available to particular processes to maintain service or service levels. Examples of these products are workload managers and HACMP.

Provisioning and workload management deal with varying load in different ways. A simplified version of these approaches is:

- Workload management works within a fixed resource pool. It delays lower priority work to keep higher priority work running at a satisfactory performance level.
- Provisioning varies the resource pool to keep all work running at a satisfactory performance level.

These approaches can, of course, be combined. Lower priority processes may also have service level agreements to meet, and thus require additional resources to be provisioned.

High availability requires that sufficient resources be available to meet minimum service levels even if part of the cluster fails. It therefore moves the applications around within the resources as required.

Table 4-2 Software provisioning reference documentation

Tool	References
Network Installation Manager (NIM)	► *AIX Installation in a Partitioned Environment,* SC23-4382-04 ► *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00 ► *The Complete Partitioning Guide for IBM @server pSeries Servers,* SG24-7039-01
Cluster Systems Management (including Reliable Scalable Cluster Technology)	► *IBM Cluster Systems Management for AIX 5L, Planning and Installation Guide, Version 1.4,* SA22-7919-07 ► *IBM Cluster Systems Management for AIX 5L, Administration Guide, Version 1.4,* SA22-7918-07 ► *IBM Cluster Systems Management for AIX 5L, Command and Technical Reference, Version 1.4,* SA22-7934-04 ► *Using a Logical Partition as a CSM Management Server,* Technote at: http://www.redbooks.ibm.com/abstracts/tips0278.html?Open ► *Cluster Systems Management (CSM) for Linux on pSeries Clusters.* Technote at: http://www.redbooks.ibm.com/abstracts/tips0295.html?Open ► *IBM Reliable Scalable Cluster Technology Administration Guide,* SA22-7889-05 ► *IBM Reliable Scalable Cluster Technology Technical Reference,* SA22-7890-06 ► *IBM Reliable Scalable Cluster Technology Messages* ► *IBM Reliable Scalable Cluster Technology Managing Shared Disks,* SA22-7937-01 ► *A Practical Guide for Resource Monitoring and Control,* SG24-6615
Workload Manager (WLM)	► *AIX 5L Workload Manager (WLM),* SG24-5977-01 ► *AIX 5L Differences Guide, Version 5.2 Edition,* SG24-5765-02

Tool	References
Partition Load Manager (PLM)	► *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, SG24-7940-00 ► *AIX 5L Version 5.3 Partition Load Manager Guide and Reference*.
Enterprise Workload Manager (eWLM)	► *Enterprise Workload Manager,* SG24-6350-00 ► *IBM Virtualization Engine: IBM Enterprise Workload Manager Version 1 Release 1* ► *Virtualization and the On Demand Business,* REDP-9115-00
Virtualization Engine (VE)	► *IBM Virtualization Engine Version 1 Release 1* ► *Virtualization and the On Demand Business,* REDP-9115-00
Virtualization Engine Console (VEC)	► *Virtualization Engine console Version 1 Release 1*
High Availability Cluster Multi-Processing (HACMP)	► *HACMP Planning and Installation Guide, Version 5.2,* SC23-4861-03 ► *HACMP Administration and Troubleshooting Guide, Version 5.2,* SC23-4862-03 ► *HACMP Concepts and Facilities Guide, Version 5.2,* SC23-4864-03 ► *HACMP and New Technologies for Availability,* whitepaper, June 2004, PDF available from: http://www-1.ibm.com/servers/eserver/pseries/software/whitepapers/hacmp_newtech.html ► *Installation and Support Recommendations for Successful High Availability Environments using IBM HACMP V5.1 for AIX 5L*, whitepaper, July 2004, PDF available from: http://www-1.ibm.com/servers/eserver/pseries/software/whitepapers/hacmp_installsupport.html
Service Update Management Assistant (SUMA)	► *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00

4.2.1 Network Installation Manager (NIM)

NIM provides remote install of the operating system, manages software updates, and can be configured to install and update third party applications.

Although both the NIM server and client filesets are part of the operating system, a separate NIM server has to be configured which will keep the configuration data and the installable product filesets. This server can provision several clients.

Tip: The NIM server must run with an operating system patch or maintenance level that is at least as high as the clients it serves.

The NIM installation of a client machine and the tracking of the installed software version can be automated. This is made possible by defining NIM script resources which will be started after the initiated NIM operation. The NIM resource types are shown in Figure 4-1.

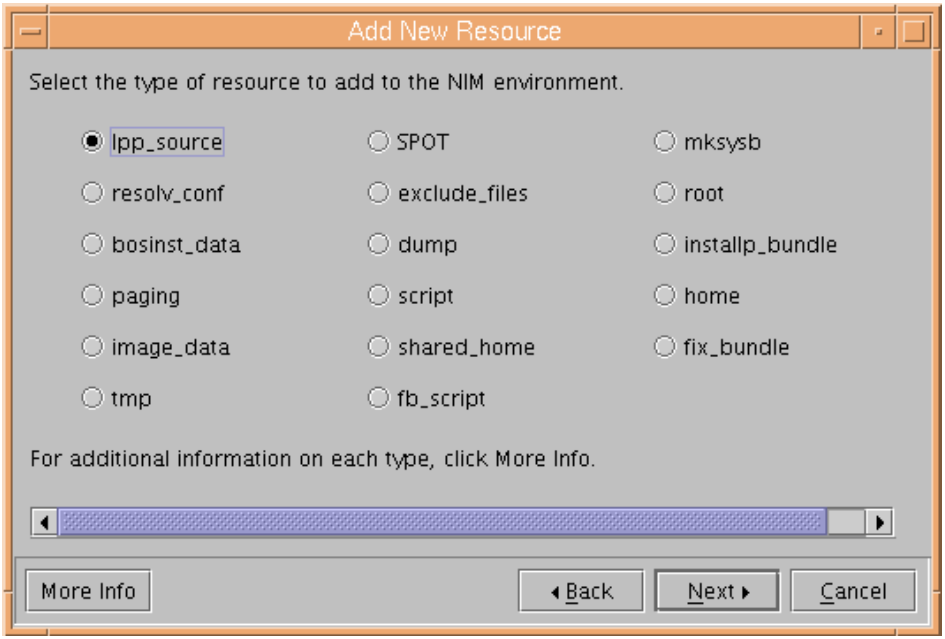


Figure 4-1 NIM resource types

An fb_script will run after the first reboot of the client machine. Scripts which are started after customization and installation operations can help to automate the provisioning process. Scripting third party application installs is also possible if the application program code and parameter files are made available in a repository.

An image_data resource contains information about the physical disks and file systems configured for rootvg at install time. Customizing this resource allows you to predefine settings for provisioned systems. This can be important in the case of cloned installation and storage virtualization.

Note: NIM is provided as part of AIX.

4.2.2 Cluster Systems Management (CSM)

CSM provides a distributed systems management solution for maintaining clusters of AIX and Linux nodes. CSM has a client server architecture. It utilizes the Resource Management and Control (RMC) part of RSCT to manage pSeries servers and LPARs.

The functionality of CSM includes:

- Installing and updating software on nodes
- Distributed command execution
- Hardware control
- File synchronization across managed nodes
- Monitoring resources in the cluster

CSM uses NIM for software installation and update. It provides commands to set up the NIM environment and create machine and other resources. However, it does not prevent independent use of NIM. After CSM setup and cluster configuration we can still manage the software installation and maintenance of machines that are not part of the CSM cluster.

High Availability Management Server (HA MS)

A new feature provided by CSM is the High Availability Management Server (HA MS). HA MS prevents the management server from being a single point of failure in a CSM cluster. HA MS uses an embedded version of Tivoli System Automation Server (TSA) to control the failover of the MS functionality. At failover, all of the MS functionality will be taken over by the backup management server, including NIM, hardware control (except for legacy SP nodes) and Configuration File Manager (CFM).

HA MS keeps the CSM cluster under control if the primary management server goes down. It can also reduce downtime as the unused server can undergo maintenance without disruption to the service.

CSM HA MS requires two identical management servers and a shared external disk subsystem where all the management domain related data will be stored. Figure 4-2 on page 40 shows the configuration elements of an HA MS setup.

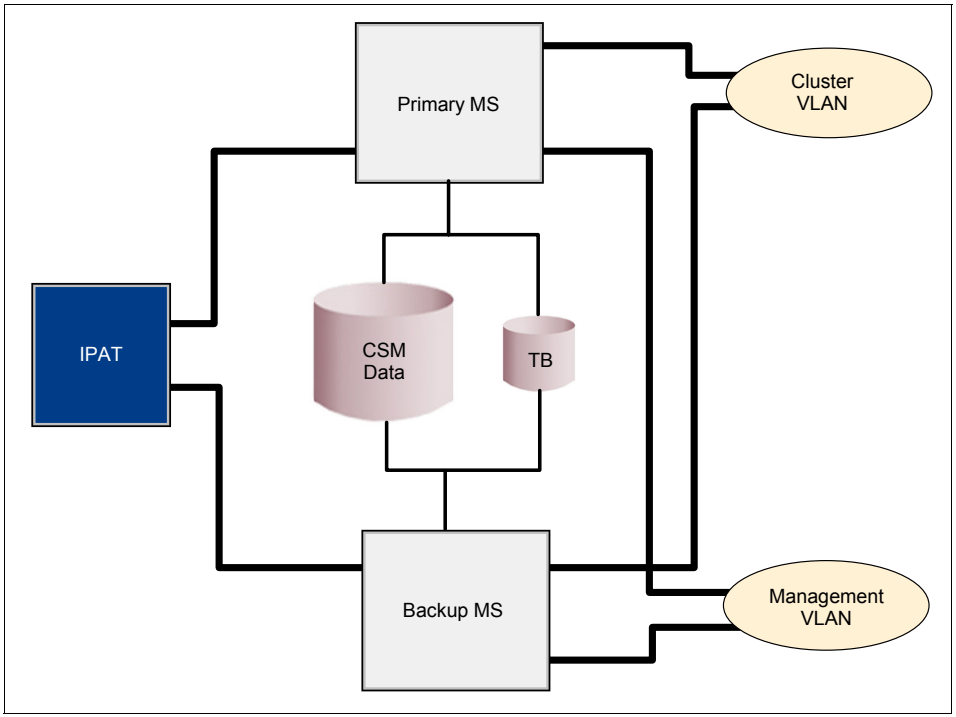


Figure 4-2 HA MS configuration

Although RSCT is part of AIX, the CSM server code is a licensed product, and the CSM HA MS is an optional priced feature.

4.2.3 WorkLoad Manager (WLM)

WLM monitors the use of CPUs, memory and disks within a standalone system or LPAR, providing real-time load statistics. It can also be configured to limit the use of these resources by particular processes or groups of processes. WLM uses a rules file to put each process in a class. This class then has an entitlement to a certain amount of resource, which can be a fixed percentage figure or a share that adjusts according to which classes are active at any given time.

Note:

- WLM can manage CPUs, memory and I/O.
- WLM was introduced in AIX 4.3.3, but features have been added over time. It can be managed from the command line, SMIT or IBM Web-based System Manager.
- WLM is provided as part of AIX.

4.2.4 Partition Load Manager (PLM)

PLM monitors the use of CPUs and memory by partitions, providing real-time load statistics. If desired, it can also automatically move resources between partitions to maximize performance, based on pre-configured rules. It is managed with a GUI (WSM).

PLM can manage partitions running AIX 5.2 maintenance level 4 or higher but not Linux or i5/OS™. It can manage both dedicated and shared processor partitions, with dedicated partitions having a CPU granularity of 1. For shared processor partitions, both the CPU entitlement and the number of virtual CPUs (maximum number of concurrent running processes) can be managed.

Note:

- PLM can manage CPUs and memory.
- To use PLM, you must purchase the advanced POWER virtualization feature.

4.2.5 The Virtualization Engine

The IBM Virtualization Engine is a set of technologies and systems services that enable system administrators to access and manage resources across a heterogeneous environment.

Technologies built into the IBM @server and TotalStorage brands provide the capabilities for server and storage virtualization. The next generation of system products will incorporate virtualization intelligence in the processors, memory and network.

IBM Virtualization Engine systems services, built on the capabilities of IBM @server and TotalStorage products, reduce the complexity of the IT infrastructure by virtualizing server, network, application and data resources.

The virtualization engine provides a consistent interface to these multiple resources, seeing them as a single pool. Having such a single, virtual environment allows you to:

- Manage multiple systems as one logical server
- Provision capacity rapidly
- Reduce complexity of monitoring and management
- Maximize resource utilization

The Virtualization Engine systems services include:

- Tivoli Provisioning Manager
- Enterprise Workload Manager
- IBM Grid Toolbox V3 for Multiplatforms
- Virtualization Engine Console
- IBM Director Multiplatform
- SAN Virtualization Services

The Grid Toolbox is covered in 8.1.3, “GRID computing” on page 165.

Note: The Virtualization Engine is a licensed product. It is not the same as the advanced POWER virtualization feature for AIX.

The Virtualization Engine Console (VEC)

As its name implies, the VEC is the front end for the virtualization engine, providing a graphical interface to the resource pool. Built on the IBM Integrated Solutions Console and WebSphere Portal technology, it is a Web-based platform that can host all the administrative console functions of IBM server, software and storage products.

The VEC provides two applications:

- A health center, which can monitor and manage AIX, Linux, i5/OS and Windows platforms from a single interface
- A console launchpad, which provides a single point from which you can launch local and Web-based administrative consoles for the various platforms (including the HMC).

The VEC communicates with CSM, IBM Director and IBM Director Multiplatform management sources via Web services (see 8.1.2, “Web services” on page 165 for further details on Web services). It uses Extensible Markup Language (XML) for information exchange, LDAP for authentication services and Lightweight

Third-Party Authentication (LTPA) for single sign on, credential and token management. The Web services have been implemented as Enterprise Java Beans (EJB).

Enterprise Workload Manager (eWLM)

Enterprise Workload Manager monitors the performance of systems and applications in a “management domain”. It can work purely as a monitoring application or, additionally, provide recommendations to a load balancer about which server work should be allocated to. Currently the only supported load balancer is the CISCO Content Switch Module for the CISCO Catalyst family of switches.

eWLM with a load balancer is one of the products that is on the boundaries of provisioning. Over-simplifying, provisioning moves the resources to the work, while load balancing moves the work to the resources.

eWLM has a client-server architecture. Managed servers (clients) can run IBM AIX, IBM OS/400, Microsoft® Windows, or Sun™ Solaris. These provide performance data to a domain manager, which can run on AIX, Linux, OS/400 or Windows. The domain manager is accessed through the eWLM control center, which is Web-based. From here, you can monitor the domain, including “goal versus actual” performance statistics, and trigger activities that alter operational states.

The performance data is provided by applications instrumented according to the Open Group’s Application Response Measurement (ARM) 4.0 standard. ARM 4.0 provides interfaces for applications to log certain actions, such that the time to complete these actions can be monitored. In addition to operating system interfaces provided for all the managed servers, IBM currently provides ARM-instrumentation support for DB2 Universal Database, version 8.2, and WebSphere Application Server, version 5.1.1.

What is perceived by the business as a single process or application may be instantiated in several “real” processes on different machines. These can be associated using a “correlator” - a byte array that enables eWLM to see low level processes as part of a single flow of work. If such a flow has a bottleneck “downstream”, eWLM will try to send less work to that flow, even if there are no issues with the initial server or process.

WLM operates on classes within a single operating system. PLM operates on AIX partitions within a single physical pSeries machine. eWLM operates on operating systems and applications in a heterogeneous environment.

4.2.6 High Availability Cluster Multi-Processing (HACMP)

HACMP is a layer above AIX that provides failover between resources, including entire systems, thus avoiding single points of failure. Once configured, it will rapidly reprovision applications with CPU, memory and disk resources in the event of a failure within the initially configured operating environment.

Failover can be local (HACMP) or remote (HACMP/XD, previously HAGEO) and is often combined with a similarly duplicated storage setup. Up to 32 nodes (servers) can provide mutual failover with applications running on each node, minimizing the additional resources required.

Resources that can be made highly available in this way include volume groups, filesystems, IP interfaces and networks and, of course, applications.

Note: HACMP is a separate licensed product.

4.2.7 Service Update Management Assistant (SUMA)

SUMA can automate the download and installation of AIX system fixes.

SUMA can be configured to check for and download particular fixes or types of fixes, including maintenance levels. Pre-defined tasks are scheduled and notifications sent to a specified mail address. Other configurable features include the number of concurrent downloads.

Note: SUMA is provided as part of AIX.

4.3 Comparison of the tools available

This section compares the products available according to the following criteria:

- Ease of installation and configuration of servers and clients
- Requirements (hardware and software)
- Features provided
- The tool it forms a part of
- Performance
- Security
- Interfaces
- Education available: a partial list of the courses available is given as an indicator of product complexity

Comparisons are only given where relevant and differ between the first five and subsequent sub-sections.

The next five sections look at products where one is a subset of the other. For example, CSM requires NIM but provides numerous additional features. These sections are:

- 4.3.1, “HMC and VEC” on page 45.
- 4.3.2, “POWER Hypervisor™ and Partition Load Manager” on page 47.
- 4.3.3, “Virtual I/O and physical networks” on page 49.
- 4.3.4, “NIM and CSM” on page 51.
- 4.3.5, “HACMP” on page 52 (advantages compared to a “standalone” system).

The last two sections look at products which provide similar functionality in different ways, namely:

- 4.3.6, “Dedicated and shared processor partitions” on page 54.
- 4.3.7, “Workload management and partitioning” on page 56.

4.3.1 HMC and VEC

At present, the VEC simply launches a remote HMC session, although further integration is planned. A local HMC is still required. The advantage of the VEC is that it can launch multiple applications from the same screen, and will provide a consistent view of your whole environment. See Table 4-3 for more details.

Table 4-3 Comparison of the HMC and the VE Console

Feature / application	**HMC**	**VEC**
Server installation and configuration	From CD (approx. 2 hours).	From CD (3-4 hours).
Client installation and configuration	The HMC exists to configure pSeries machines, so these are treated as the client here. No specific client software needs to be installed.	No additional client software must be installed on the HMC for VEC to connect. However, connections to Tivoli Provisioning Manager, CSM, IBM Director Multiplatform and other applications require the installation of a WebSphere Application Server “bridge”, usually on the client machine.

Feature / application	HMC	VEC
Requirements	► The rack-mounted version has 1x Intel® processor; 1Gb memory; 40Gb disk. ► Serial and ethernet connections to managed servers.	► Minimum eServer™ configuration of 1x1.4Ghz processor; 1Gb memory; 350 Mb disk space with latest OS. ► WebSphere Application Server V5.1. ► LDAP Directory Server V5.1 ► Web browser. ► Network connection to the HMC and other clients.
Additional features	N/A	► Health center, giving an overview of multiple managed servers on various e-server platforms. ► Launchpad for: – Tivoli Provisioning Manager – IBM Director Multiplatform – IBM Director Console – Web-based System Manager and other applications.
Tool	The HMC is a separately priced product.	The VEC is a separately priced product.
Performance	The HMC is java-based. It takes a few seconds to respond when accessed locally, but can be much slower over the network. Later versions (with Java Web Start) are faster.	The team did not do a performance test with VEC.
Security	The HMC communicates with the LPARs over serial link and by unencrypted ethernet. It is recommended that a separate management LAN is used. For remote connection to the HMC, ssh can be used.	The VEC uses an LDAP server for user authentication but VEC does not support SSL for this connection. Browser access to the VEC requires a username/password to login. The browser connection to the portal server can be SSL enabled. The user must configure the portal server for SSL. Client-server communication is in XML for the major applications, and uses an encrypted LTPA token for authentication.

Feature / application	HMC	VEC
Interfaces	Graphical interface only. Icons plus selection from a menu bar.	Graphical interface only. A “dashboard” gives easy access to frequently used applications, while the healthcenter offers a simple menu system.
Education	HMC p6xx hardware training (2.5 days).	The VEC is introduced as part of the Virtualization Engine Suite in the Virtualization Engine Hands-on Workshop (3 days, currently invitation only). The products it uses are covered in the following courses: ► Administration of WebSphere Application Server V5 (4 days) ► IBM Tivoli Directory Server (LDAP) System Administration (3 days) ► IBM Tivoli Provisioning Manager and ThinkDynamic Intelligent Orchestrator Admin (4 days) ► Systems Management with IBM Director (4 days) ► Advanced Systems Management with IBM Director (2 days)

4.3.2 POWER Hypervisor™ and Partition Load Manager

Both POWER Hypervisor (PHYP) and Partition Load Manager (PLM) will allocate free CPU time to uncapped partitions. Table 4-4 on page 48 looks at the advantages provided by PLM.

PHYP and PLM control the distribution of CPU cycles in different ways. For a detailed discussion of this and an explanation of the terms used below (“entitlement” and “shares”, for example) see Appendix A, “CPU resource distribution by Hypervisor and PLM” on page 173.

Table 4-4 Comparison of POWER Hypervisor and Partition Load Manager

Feature / application	POWER Hypervisor	Partition Load Manager
Server installation and configuration	PHYP is installed by default on all pSeries machines. An HMC must be installed (see Table 4-3 on page 45).	The managed server must have an HMC. LPARs must be defined through the GUI, be installed and have RMC connections to the PLM server, which must be separately installed. POWER Hypervisor (PHYP) entitlements must be manually configured.
Client installation and configuration	If we consider the LPARs as clients, each must be configured with maximum, minimum and desired entitlement settings. They must also be capped or uncapped. These settings must be carefully planned according to known or expected workloads.	If we consider the LPARs as clients, each must be configured with maximum and minimum entitlements, guaranteed entitlements and delta values (the amount to change its entitlement by if required). These settings must be carefully planned according to known or expected workloads.
Requirements	An HMC must be present in order to define LPARs.	Micro-partitioning requires the advanced POWER virtualization feature to be installed. Dynamic LPAR operations require network connections from the HMC to the LPARs and matching name resolution. PLM requires RMC connections to both the LPARs and the HMC.
Additional features	(PHYP does not dynamically distribute memory)	► Entitlement to CPU cycles is changed according to the shares allocated to the partition. ► The number of virtual CPUs will be changed to match the number of cycles allocated. ► Because the number of virtual CPUs can be changed by PLM, it is not necessary to configure a large number on boot, which incurs a performance overhead. ► Entitlement to memory is changed according to the shares allocated to the partition.
Tool	PHYP is provided with pSeries machines.	Requires the purchase of the advanced POWER virtualization feature for AIX.

Feature / application	POWER Hypervisor	Partition Load Manager
Performance	PHYP schedules LPARs on physical CPUs in the same way that AIX schedules processes.	LPARs are configured with trigger points. When a trigger point is reached a certain number of times, an RMC request for more resources is generated. By default, additional resources will be requested after a minute of high load. A Dynamic LPAR operation lasting a few minutes is then initiated. Dynamic LPAR remove operations can take longer.
Security	On POWER5, the HMC should be connected to the pSeries by a private ethernet network. Remote HMC sessions can use ssh.	RMC uses Access Control Lists for authorization. It can use public key exchange or third party authentication.
Interfaces	► The HMC GUI (local or remote)	► The PLM GUI (local or remote) ► PLM configuration files

4.3.3 Virtual I/O and physical networks

As with many virtualization features, the additional flexibility of virtual I/O must be balanced against other effects, as discussed in Table 4-5.

Table 4-5 Comparison of physical and virtual I/O

Feature / application	Physical networks	Virtual I/O
Server installation and configuration	No additional server software is required.	The VIO server must be installed and the disks (physical or logical) exported. Configuration time will depend largely on the number of disks.
Client installation and configuration	Physical adapters will be automatically configured by AIX. Interfaces must be manually configured.	Virtual adapters must be configured on client LPARs before interfaces can be added.
Requirements	Adapters must be physically attached and part of the LPAR definition. The device drivers must be installed.	Adapters must be physically attached and part of the VIO server definition.

Feature / application	Physical networks	Virtual I/O
Additional features	N/A	► Enables physical disks to be shared amongst multiple partitions. ► Enables inter-partition communication without requiring a physical network connection. ► Enables network adapters to be shared amongst multiple partitions. ► IP networks can be divided up in to VLANs (if the network environment supports them).
Tool	Device drivers are provided as part of AIX.	Requires the purchase of the advanced POWER virtualization feature for AIX.
Performance	pSeries provides fast and reliable networking.	► For dedicated partitions, virtual ethernet throughput *should be* comparable to a 1 Gb ethernet for small packets, and better for large packets. ► Virtual SCSI will suffer from performance degradation relative to directly attached disks. ► There is an increase in CPU load for both virtual SCSI and virtual ethernet - nearly double for virtual SCSI but less for virtual ethernet. ► Both types of virtual network can suffer from resource contention.
Security	From AIX 5.2, systems can be installed with the Controlled Access Protection Profile and Evaluation Assurance Level 4+. AIX provides ssh connections; file, LDAP and other authentication methods; and a public key infrastructure certificate authentication service.	The VIO server implements a layer 2 switch for virtual ethernet. The TCP/IP security remains unchanged. Virtual SCSI uses secure direct memory access conforming with the SCSI RDMA Protocol.
Interfaces	► IBM Web-based System Manager ► AIX System Management Interface Tool (SMIT) ► command line	► IBM Web-based System Manager ► Limited command line, either by logging in or on the HMC

Feature / application	Physical networks	Virtual I/O
Education	► AIX Network Management I: configuration and implementation (5 days) ► AIX Network Management II: advanced TCP/IP (3 days)	Not yet available

4.3.4 NIM and CSM

CSM uses NIM for its software management capability, but provides a number of additional features (Table 4-6).

Table 4-6 Comparison of NIM and CSM

Feature / application	NIM	CSM
Server installation and configuration	The NIM master filesets must be installed. There is a nim_master_setup script that walks through a basic configuration. This takes approximately 2 hours, including the creation of the basic resources for one operating system level.	The CSM master filesets must be installed. There are a number of manual steps that must be followed in the configuration of CSM, including the setup of the HMC connection. However, CSM does provide a script to set up NIM in a way suited to use by CSM. This takes approximately 3 hours, including the NIM setup but not that of the network hardware.
Client installation and configuration	Client systems must have fileset bos.sysmgt.nim.client installed and allow remote command access from the server. These can be configured by the NIM installation. If the client is not running, it must be set to boot from the network adapter by other means (manually or from the HMC).	The CSM client fileset is installed as part of AIX. CSM will perform configuration of clients over and above that done by NIM. Remote command access must be granted to the server. If monitoring is desired, it must also be configured.
Requirements	Storage for installable filesets.	► Working NIM environment ► Connection to HMC if hardware control is needed ► License for CSM management server software code ► Optional installation network for higher security

Feature / application	NIM	CSM
Additional features	N/A	► Hardware control including remote power ► Distributed command execution ► File synchronization across managed nodes ► Cluster resource monitoring ► HA MS for high availability management server as a separately priced option (although an HA NIM is available)
Tool	Provided with AIX.	CSM is a separately priced product.
Performance	As it uses NFS between NIM server and client systems, performance may vary for different network setup.	The same as NIM for install functions. Hardware control and monitoring depend on the connection to HMC.
Security	► NIM service handler for client communication, basic nimsh ► NIM cryptographic authentication can further increase the security using OpenSSL	► As NIM for installation ► OpenSSH based remote command execution ► Kerberized OpenSSH or **rsh** (only on AIX V5.2 or higher) for remote command execution ► For RMC operations, RSCT security is used. RMC uses Access Control Lists for authorization. It can use public key exchange or third party authentication.
Interfaces	► IBM Web-based System Manager ► AIX System Management Interface Tool (SMIT) ► command line	► IBM Web-based System Manager ► AIX System Management Interface Tool (SMIT) ► command line
Education	AIX5L: Network Installation Management (2 days)	► @server AIX CSM Administration (5 days) ► Linux Clustering with CSM and GPFS (2 days)

4.3.5 HACMP

This section looks at the benefits HACMP can provide over a “standalone” AIX install. See Table 4-7 on page 53.

Table 4-7 HACMP features

Feature / application	AIX	HACMP
Server installation and configuration	From CD or over the network (approx 3 hours for basic configuration).	The software can be installed from CD or over the network in less than an hour. HACMP V5.2 provides a Two-Node Cluster Configuration Assistant and cluster test tool to help you set up a basic cluster in a matter of hours. However, setting up the hardware and configuring more complex clusters can require many days of planning, configuration, scripting and testing.
Requirements	p or i-series e-server. For AIX 5.3, only CHRP machines are supported, with a minimum of 128 MB of memory and a 2.2 Gb hard drive.	A minimum of 2 pSeries machines running AIX with shared disks and 2 subnets in a no single point of failure configuration.
Additional features	N/A.	Failover is provided for: ► Sites ► Applications ► Volume groups ► Filesystems ► IP interfaces and networks Additional actions can be scripted to run before or after failover events. Cluster information is available from any host.
Tool	AIX is a separately priced product.	HACMP is a separately priced product.
Performance	AIX 5.3 on POWER5 has set performance records in diverse applications, such as online transaction processing, enterprise resource planning (ERP), file sharing, and high performance computing applications such as fluid dynamics. For further details, see: http://www.ibm.com/servers/eserver/pseries/news/pressreleases/2004/jul/power.html	Application performance will be determined by the pSeries resources available. The cluster must therefore contain enough resources to run the applications, even after failover. Small clusters failover in approximately 30 seconds. Clusters with a large number of disks will take longer, but this is improved by fast disk takeover in HA 5.1 and up.

Feature / application	AIX	HACMP
Security	From AIX 5.2, systems can be installed with the Controlled Access Protection Profile and Evaluation Assurance Level 4+. AIX provides ssh connections; file, LDAP and other authentication methods; and a public key infrastructure certificate authentication service.	HACMP 5.2 offers method authentication and encryption. It allows only trusted commands to be run as root, and no longer relies on /.rhosts. Kerberos can be used on PSSP nodes, or nodes can be configured to use Virtual Private Networks for all node-to-node communication.
Interfaces	► IBM Web-based System Manager ► AIX System Management Interface Tool (SMIT) ► command line	► IBM Web-based System Manager ► Cluster Single Point of Control (C-SPOC) ► AIX System Management Interface Tool (SMIT) ► Command line
Education	► AIX 5L System Administration I: Implementation (5 days) ► AIX 5L System Administration II: Problem Determination (5 days) ► AIX 5L System Administration III: Performance Management (5 days) ► AIX 5L System Administration IV: Storage Management (4 days) ► AIX 5L System Administration V: Workload Manager (3 days) ► AIX 5L TCP/IP Administration (3 days)	► HACMP Systems Administration I: Planning and Implementation (5 days) ► AIX HACMP System Administration II: Maintenance and Migration (5 days) ► HACMP System Administration III: Problem Determination and Recovery (5 days) ► AIX HAGEO Implementation (5 days)

4.3.6 Dedicated and shared processor partitions

Although Micro-partitioning offers greater flexibility, there are still advantages to using dedicated partitions in some situations. This is discussed in Table 4-8.

Table 4-8 Comparison of dedicated and shared processor partitioning

Feature / application	Dedicated partitions	Shared processor partitions
Server installation and configuration	The server must have an HMC. LPARs must be defined through the GUI.	The server must have an HMC. LPARs must be defined through the GUI.

Feature / application	Dedicated partitions	Shared processor partitions
Requirements	No additional software is required. Dynamic LPAR operations require network connections from the HMC to the LPARs and matching name resolution.	Micro-partitioning requires the advanced POWER virtualization feature to be installed. Dynamic LPAR operations require network connections from the HMC to the LPARs and matching name resolution.
Features comparison	1. Only whole processors can be assigned to partitions. 2. Each CPU can only be used by one partition. 3. Dedicated memory required 4. Unused CPUs can be kept out of the shared pool, guaranteeing their availability. The default is to return them to the shared pool, where they can only be “locked” by starting additional partitions that increase the total CPU entitlement. 5. The number of processors (and therefore concurrent running processes) is determined by the hardware. 6. AIX processor affinity management will work. 7. Virtual processors have no dispatch latency (delay between one dispatch and the next) in a dedicated partition environment.	1. A minimum of 1/10th of a CPU can be assigned to a partition. 2. Each CPU can be used by multiple partitions. 3. Dedicated memory required 4. Unused CPUs will be returned to the shared pool. They can only be “locked” by starting additional partitions that increase the total CPU entitlement, or adding entitlement to existing LPARs. 5. The number of virtual processors (concurrent processes) can be manually configured. 6. AIX processor affinity management has no meaning in a shared processor partition because virtual processors may be dispatched on different physical processors. Memory is allocated to partitions in a round robin fashion. Cache-sensitive applications should be deployed in a dedicated partition. 7. Virtual processors can have a dispatch latency (a delay between one dispatch and the next) in a micro-partitioned environment, which can particularly affect interrupts. Applications that cannot tolerate this variability should be deployed in a dedicated partition (with simultaneous multi-threading turned off).
Tool	Provided with a pSeries + HMC.	Requires the purchase of the advanced POWER virtualization feature for AIX.

4.3.7 Workload management and partitioning

As discussed in the Technote *Server Consolidation on IBM pSeries Systems*, TIPS0306 there are two approaches to server consolidation: vertical consolidation, where many servers are consolidated on to fewer, more powerful servers; and horizontal consolidation, where multiple servers share a common workload, providing enhanced scalability and reliability.

Within vertical consolidation, pSeries has developed two distinct options: Workload Manager (WLM) and Logical Partitioning (LPAR). Both provide a degree of separation between applications, and provide features that prevent important applications being starved of resources.

With the introduction of shared processor partitions (SPLPARs, a feature of Micro-partitioning) and Partition Load Manager (PLM), partitions are approaching the flexibility and granularity of WLM classes in their responses to changing load, while providing the additional security of separate operating systems. Table 4-9 on page 57 compares WLM classes and Micro-Partitioning™ technology in terms of their abilities to dynamically provision resources (CPU, memory and I/O) to applications, and the features they provide. Micro-Partitioning technology is not necessarily smaller than 1 CPU, but they can be given CPU entitlement in fractions of 0.01 CPUs (1.75 CPUs, for example). We assume that WLM is configured on dedicated processors.

In general, WLM still provides a greater degree of control and granularity, and classes are still more dynamic in their response to changes in load than Micro-Partitioning technology. However, by running separate operating systems, Micro-Partitioning technology provides an additional degree of separation with clear advantages for availability. PLM can also run with dedicated partitions, avoiding the performance overhead of Micro-Partitioning technology, but this reduces the granularity of control still further.

The following documentation is available about server consolidation:

- Redpaper “LPAR Heterogeneous Workloads on the IBM pSeries 690 System” at http://www.redbooks.ibm.com/abstracts/redp0425.html?Open
- Redbook *Server Consolidation on IBM @server pSeries Systems*, SG24-6966
- Redbook *Server Consolidation on RS/6000*, SG24-5507

Table 4-9 Comparison of WLM, Micro-Partitioning technology + PLM

Feature / application	Workload Manager	Micro-Partitioning technology + PLM
Server installation and configuration	WLM is installed by default. Classes, rules, tiers, limits and shares must be manually configured.	The managed server must have an HMC. LPARs must be defined through the GUI, be installed and have RMC connections to the PLM server, which must be separately installed. POWER Hypervisor (PHYP) entitlements and capping, and PLM entitlements and shares, must be manually configured.
Requirements	No additional hardware or software is required.	Micro-partitioning requires the advanced POWER virtualization feature to be installed. Dynamic LPAR operations require network connections from the HMC to the LPARs and matching name resolution. PLM requires RMC connections to both the LPARs and the HMC.

Feature / application	Workload Manager	Micro-Partitioning technology + PLM
Features comparison: resource entitlement	1. Classes can have maximum, minimum and target resource entitlements. A class may be given less than its target if it cannot use the resources. It will only be given less than its minimum if it cannot use the resources, or if a higher tier class (see “prioritization”) takes all the resources. 2. Target entitlements are known as shares. The resources given to a class are determined by its share divided by the total number of shares for active classes. These shares are proportions of the total resources of the machine. An active class is one with running processes. 3. The sum of the defined minimum resource entitlements cannot exceed the total capacity of the system, even if some classes are not active. 4. A class with a maximum entitlement of 100% can use any free resources on the system. 5. I/O throughput can be controlled. I/O resources can be shared between classes.	1. Partitions can have maximum, minimum and guaranteed resource entitlements in the PHYP. A partition will only be given less than its guaranteed amount if it cannot use the resources assigned to it. It will never be given less than its minimum entitlement. 2. Partitions are assigned a share in PLM. The resources given to an LPAR are determined by its share divided by the total number of shares for active LPARs. PLM will override the hypervisor’s normal reallocation of these additional resources. 3. The sum of the defined minimum capacity entitlements can exceed the total capacity of the system as long as not all the partitions are started. 4. An uncapped partition can use any free resources on the system, as PLM will increase a partition’s virtual CPUs in order to exploit additional CPUs. 5. I/O throughput is not controlled. I/O resources can only be shared through a VIO server. PLM cannot move I/O resources between partitions.

Feature / application	Workload Manager	Micro-Partitioning technology + PLM
Features comparison: allocation and separation	1. All processes within an OS are assigned to a class. 2. All classes run within the same OS. An OS crash will stop all the classes. 3. A process in one class can start a process in another class (if inheritance is turned off). 4. Resource sets can be used to restrict a class to particular CPUs.	1. All processes run within a partition. 2. Partitions run separate operating systems. An OS crash in one partition will have no effect on the others. 3. A process in one partition can only start a process in another partition over the network. 4. The administrator has no control over which CPUs in the shared pool are used by a particular partition. However, LPARs can be grouped so they only compete against others in the group.
Features comparison: prioritization	1. Classes can be put in to tiers. Processes in a lower tier class will only run if no higher tier processes are runnable. Higher tier classes therefore cannot be limited by lower tier classes, but lower tier classes can be starved. 2. Processes can be started, and classes activated, even if they cannot achieve their minimum entitlement.	1. PLM has no concept of the importance of a workload beyond the share setting (see “resource entitlement”). Running a lower priority Micro-Partitioning technology will limit the resources available to a higher priority Micro-Partitioning technology because the lower priority Micro-Partitioning technology will still use its guaranteed entitlement. However, lower priority partitions cannot be starved. 2. New partitions will not start if their minimum PHYP requirements are not met.
Features comparison: performance overhead	1. WLM is built in to the definition of a process. Once running, the overhead is minimal. 2. WLM can significantly increase the boot time of an OS if there are a large number of disks attached. 3. It is assumed that WLM will run on dedicated processors. See 4.3.6, “Dedicated and shared processor partitions” on page 54 for a comparison. 4. Only one OS is required.	1. Resource Management and Control (RMC) services gather and export the system status. The RMC daemon also processes reconfiguration requests from the HMC. 2. The RMC services are always started on boot. 3. See 4.3.6, “Dedicated and shared processor partitions” on page 54 for a comparison. 4. Each partition must have its own OS.

Feature / application	Workload Manager	Micro-Partitioning technology + PLM
Features comparison: speed of response to changing load	1. There is no latency associated with a class using additional CPU. 2. Monitoring is constant. Access to a class's resource entitlement is provided on a per-minute basis (as long as the class can use its full entitlement).	1. There is a latency associated with dynamically adding virtual CPUs. Furthermore, if a high number of virtual CPUs are made permanently available instead, a performance overhead is incurred during times of low load. Additional entitlement (up to 100% of a partition's virtual CPUs) can be added without delay. 2. Monitoring is based on 10 second intervals. By default, a threshold must be reached 6 times in order to trigger a dynamic LPAR event. Entitlement changes are made only when an event is triggered, but excess capacity is distributed constantly (based on partition weight).
Tool	Provided with AIX.	Requires the purchase of the advanced POWER virtualization feature for AIX.

5

General scenario description

In this chapter, we provide general information relevant to the sample provisioning scenarios in Chapter 6, “POWER4 provisioning scenario” on page 95 and Chapter 7, “POWER5 provisioning scenario” on page 135. In those two chapters, we show the steps required to configure the two environments.

Here, we discuss the general considerations that need to be taken into account in pSeries provisioning:

- 5.1, “Summary of procedures used for scenarios” on page 62.
- 5.2, “Aim of the scenarios” on page 63.
- 5.3, “General considerations” on page 63.

5.1 Summary of procedures used for scenarios

This section summarizes the procedures we used to implement the two pSeries provisioning scenarios described in this publication. The scenarios implemented were:

- A pSeries POWER4 provisioning scenario
- A pSeries POWER5 provisioning scenario

5.1.1 pSeries POWER4 provisioning scenario

In this scenario, we used several tools such as NIM, CSM, WLM, and dynamic LPAR. These tools are very useful for automation in pSeries systems.

In our scenario, we configured the CSM/NIM management server with the latest software (AIX Version 5.3, CSM Version 1.4.0.1), created two LPARs in the p690 system, and automatically installed the operating system using NIM.

To enhance the ability to meet unexpected or temporary business demands, we utilized Dynamic Logical Partitioning to reallocate the needed resources without system downtime. Moreover, the dynamic LPAR ToolSet is used for automated operation.

Reliable Scalable Cluster Technology (RSCT) provides the Resource Monitoring and Control (RMC) function. The RMC subsystem can be used for system monitoring. In our scenario, we monitor one sample file system to adjust its size automatically.

Chapter 6, “POWER4 provisioning scenario” on page 95 shows the process of implementing this scenario.

5.1.2 pSeries POWER5 provisioning scenario

POWER5 introduces an advanced POWER virtualization feature which includes:

- Firmware enablement for micro-partitioning
- An installation image for the Virtual I/O server software which supports:
  - Ethernet adapter sharing
  - Virtual SCSI server
- Partition Load Manager (PLM)

We decided to implement this new functionality of POWER5 in our environment.

In the POWER5 provisioning scenario, we used the CSM/NIM management server configured during the implementation of the POWER4 scenario.

We already had a POWER5 system environment in our laboratory with two LPARs and one virtual I/O server. We then implemented our provisioning scenario in this environment.

Chapter 7, “POWER5 provisioning scenario” on page 135 shows the process of implementing this scenario.

5.2 Aim of the scenarios

The scenarios show how pSeries systems are integrated, automated, and centralized. pSeries provisioning needs several tools and technologies to accomplish full automation. Therefore, the provisioning process has many steps. However, once these steps are completed in your pSeries environment, you can provision and manage your environment more effectively than before.

5.3 General considerations

In this section, we discuss the general factors our team needed to consider before implementing our scenarios. You should consider the following guidelines provided in these topics before configuring your provisioning environment:

- “Hardware Management Console (HMC)” on page 64.
- “Advanced System Management Interface (ASMI)” on page 65.
- “Operating system” on page 67.
- “NIM” on page 67.
- “Alternate disk installation” on page 68.
- “CSM” on page 70.
- “WLM” on page 72.
- “Dynamic Logical Partitioning” on page 72.
- “Virtualization system technology” on page 75.
- “Capacity on Demand” on page 84.
- “Simultaneous multi-threading” on page 88.
- “Partition Load Manager (PLM)” on page 88.
- “HACMP” on page 92.

We did not implement some available pSeries technologies in our scenario because of limited hardware resources, and time (for example, ASMI, HACMP, CoD, etc.).

5.3.1 Hardware Management Console (HMC)

You must set up a Hardware Management Console (HMC) to manage the POWER systems (LPAR/dynamic LPAR operations, and hardware control).

For information about managing and maintaining the HMC connected to the POWER4 or POWER5 systems, refer to the following Web site:

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

For POWER5 servers, all releases of the HMC Version 4.x machine code will manage an iSeries server, although V4 R1.3 or higher is recommended. A pSeries server requires HMC Version 4.2 (or later). The HMC models 7315-C03 and 7310-C03 must be upgraded to the latest BIOS Version **"24KT38RUS"** if you are upgrading to HMC V3 R3.0 or greater on your HMC, or to HMC V4 R1.0 or greater.

To communicate with a managed system, you can connect the HMC to a private or open network. A private network provides greater security and is easier to set up. It also allows the HMC to automatically detect the managed system. Therefore, it is recommended that you connect the HMC to a private network.

Note: Once you have completed the HMC setup, do not power down or disconnect the HMC from the managed system. If the HMC is powered down or disconnected from a non-partitioned managed system for a period of 14 days, the managed system will no longer recognize the HMC. If this situation occurs, you should set up the HMC again. If your system is partitioned, the 14 day time limit does not apply.

Depending on the level of customization you intend to apply to your HMC configuration, you have several options for setting up your HMC to suit your needs. The guided setup wizard is a tool on the HMC designed to make the setup of the HMC quick and easy. You can choose a fast path through the wizard to quickly create the recommended HMC environment, or you can choose to fully explore the available settings that the wizard guides you through. You can also perform the configuration steps without the aid of the wizard. Refer to the following Web site for details on configuring the HMC using the guided setup wizard:

http://publib.boulder.ibm.com/infocenter/iseries/v1r2s/en_US/index.htm?info/ipha1/hardwaremanagementconsolehmc.htm

5.3.2 Advanced System Management Interface (ASMI)

The Advanced System Management Interface (ASMI) is the interface to the service processor that allows you to perform general and administrator-level service tasks, such as reading service processor error logs, reading vital product data, setting up the service processor, and controlling the system power. The ASMI may also be referred to as the service processor menus. You can access the ASMI through a Web browser, an ASCII console, or the Hardware Management Console (HMC). The service processor and the ASMI are standard on all IBM @server i5 and p5 servers.

The ASMI allows you to perform a variety of tasks associated with managing your server. The following requirements are needed to successfully access and use the ASMI:

- The ASMI requires password authentication.
- The ASMI provides a Web connection to the service processor via the ethernet using Secure Sockets Layer (SSL). To establish an SSL connection, open your browser using https://.
- Supported browsers are Netscape (version 7.1), Internet Explorer (version 6.0), and Opera (version 7.23). Later versions of these browsers are not supported. JavaScript™ and cookies must be enabled.
- Clicking Back in the browser may display outdated data. The recommended way to display the most up-to-date data is to select the desired item from the navigation pane.
- The browser-based ASMI is available during all phases of the system operation, including Initial Program Load (IPL) and run time. Some menu options are blocked during system IPL or run time to prevent conflicts of ownership.
- Terminal access to the ASMI is only available if the system is powered off.

There are several authority levels for accessing the service processor menus using the ASMI. The following levels of access are supported:

- **General user**: The menu options presented to the general user are a subset of the options available to the administrator and authorized service provider. Users with general authority can view settings in the ASMI menus. The login ID is general and the default password is general.
- **Administrator**: The menu options presented to the administrator are a subset of the options available to the authorized service provider. Users with administrator authority can write to persistent storage, and view and change settings that affect the server's behavior. The first time a user logs into the

ASMI after the server is installed, a new password must be selected. The login ID is admin and the default password is admin.

- **Authorized service provider**: This login gives access to all the functions that can be used to gather additional debug information from a failing system, such as viewing persistent storage, clearing all deconfiguration errors, and using extended services. The login ID is celogin. The password is dynamically generated and must be obtained by calling IBM technical support.

The ASMI restricts logins such that only three users can access it at the same time. If three people are logged in to the ASMI and a person with a higher authority level than one of the current logged in users attempts to log in, one of the lowest privileged users is logged out. In addition, if you are logged in and not active for 15 minutes, your session expires. You receive no immediate notification when your session expires. However, when you select anything on the current page, you are returned to the ASMI Welcome pane. To see who else is logged in to the ASMI, view *Current users* on the ASMI Welcome pane after you log in.

If you make five invalid login attempts, your user account is locked out for five minutes and none of the other accounts are affected. For example, if the administrator account is locked, the general user can still log in using the correct password. This login restriction applies to the general user, administrator, and authorized service provider IDs. This login restriction also applies to the managed system HMC access ID, which is set using the HMC.

The following list contains tasks that can be performed using the ASMI:

- **Setting up login profile**: Change passwords, view login audits, and change the default language.
- **Viewing system information**: View system power control network (SPCN) trace data, progress indicator history, and vital product data (VPD).
- **Controlling the system power**: Manually and automatically control the system power.
- **Changing system configuration**: View and perform custom system configurations, such as enabling Peripheral Component Interconnect (PCI) error injection policies, viewing system identification information, and changing memory configuration.
- **Configuring network services**: Configure network interfaces, configure network access, and debug the virtual TTY.
- **Using on-demand utilities**: Activate inactive processors or inactive system memory without restarting your server or interrupting your business.
- **Using concurrent maintenance utilities**: Replace devices in your server without having to power off your server.

- **Troubleshooting the server using service aids**: View and customize troubleshooting information with various service aids (such as viewing error logs and initiating service processor dumps).

For detailed information about the Advanced System Management Interface, refer to:

http://publib.boulder.ibm.com/infocenter/eserver/v1r2s/en_US/info/iphby/iphby.pdf

http://publib.boulder.ibm.com/infocenter/eserver/v1r2s/en_US/info/iphbp/iphbp.pdf

5.3.3 Operating system

There are various methods available to install the operating system. In our scenarios, we installed the operating system automatically via the NIM master server. We set up the NIM master server with AIX 5L Version 5.3 and configured NIM. Remember that your network environment must be defined, and working correctly before installing AIX 5L automatically.

Although we used NIM in our scenario, you can use general installation methods. Refer to the following documents for detailed information about AIX installation methods:

- *AIX Installation in a Partitioned Environment,* SC23-4382-04
- *AIX 5L Version 5.2 Installation Guide and Reference,* SC23-4389-04
- *AIX 5L Version 5.3 Installation Guide and Reference,* SC23-4887-00

5.3.4 NIM

Before installing AIX using Network Installation Manager (NIM) in a partitioned environment, you have to set up a NIM master either on another pSeries server or in one of the partitions. Refer to the following Redbooks for detailed information about installing AIX using NIM:

- *The Complete Partitioning Guide for IBM @server pSeries Servers,* SG24-7039-01
- *AIX 5L Version 5.2 Installation Guide and Reference,* SC23-4389-04
- *AIX 5L Version 5.3 Installation Guide and Reference,* SC23-4887-00

The following guidelines should be considered when implementing NIM:

- The NIM master must always be at the highest level of the AIX release and maintenance level that you are going to install. Therefore, if you are going to install AIX 5L Version 5.3, the NIM master also must be AIX 5L Version 5.3.

- A minimum of 8 MB free space in the /tmp directory is needed.
- You need a network connection between the NIM master and clients. This network can be either ethernet or token ring.
- To speed up NIM installations or other NIM operations, it may be helpful to separate the NIM network from the public network. To improve network throughput, consider that the NIM master network can be configured as Etherchannel (trunking).
- We recommend that you create a separate volume group to be used for NIM operations on the NIM master. The initial free space needed for this setup is about 1.2 GB (usually one release of AIX requires approximately 4.5 GB).
- It is recommended that NIM resources are mirrored against an unexpected storage failure.
  Also, the most significant single point of failure in a NIM environment is the NIM master. AIX 5L Version 5.3 introduces a way to define a backup NIM master that can take over and then fallback to the primary master (called *HANIM*). This helps create a more reliable NIM environment.
- The NIM master and NIM client host names must be consistently resolvable.
- In previous versions of AIX, NIM used **rsh** and remote commands to perform remote execution of commands on clients. These r-commands were a potential security exposure. AIX 5L Version 5.3 is enhanced by the nimsh environment that is part of the *bos.sysmgt.nim.client* fileset. Use this **nimsh** instead of **rsh** or rcmd for additional security.

Note: The following prerequisites must be satisfied while configuring NIMSH:

- The NIM client must already be configured.
- The client must have AIX 5.3 or later installed.
- The client's NIM master must have AIX 5.3 or later.

In other words, you cannot use **nimsh** between a NIM master running AIX 5.3, and a client running AIX 5.2 or AIX 5.1.

Refer to 6.1.4, "NIM and the CSM management server" on page 100, and 7.2.2, "NIM setup for the new environment" on page 138 for NIM configuration procedures in our scenarios.

5.3.5 Alternate disk installation

We can also install AIX using the alternate disk installation method. The AIX 5L Version 5.3 introduces enhanced functionalities to make the *alt_disk_install*

operations easier to use, document, and maintain. For more information about alternate disk installation methods, refer to the following documentation:

- *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00
- *The Complete Partitioning Guide for IBM @server pSeries Servers,* SG24-7039-01
- *AIX 5L Version 5.3 Installation Guide and Reference,* SC23-4887-00

An alternate disk installation uses the following filesets:

- **bos.alt_disk_install.boot_images**: Must be installed for alternate disk **mksysb** installations.
- **bos.alt_disk_install.rte**: Must be installed for **rootvg cloning** and alternate disk **mksysb** installations.

To reduce the business impact of an operating system migration, you can use alternate disk migration installation method to simultaneously migrate it through NIM to a new release level. This is a new feature introduced in AIX 5.3.

Alternate disk migration installation through NIM has the following requirements and limitations:

- An operating system version of the NIM master must be same or higher than the NIM client system.
- *bos.alt_disk_install.rte* must be installed on the NIM master.
- The client system to be migrated must be at AIX 4.3.3 or later.
- The client must be registered to the master.
- The NIM master must be able to execute remote commands on the client using the **rshd** protocol.
- The client must have a minimum of 128 MB of memory.
- A reliable network, which can facilitate large amounts of NFS traffic, must exist between the NIM master and the client.
- If the client's rootvg has the Trusted Computing Base (TCB) option enabled, either disable it permanently or perform a conventional migration. TCB must access file metadata that is not visible over NFS.
- All NIM resources used must be local to the NIM master.
- During the migration, the client's active rootvg may experience a small performance decrease due to increased disk I/O, nfsd activity, and some CPU usage associated with alt_disk_install cloning. Therefore, NFS tuning may be required to optimize performance.

Attention: If you install AIX using alt_disk_install, the target system will be given the same node_id as the source system. To prevent this problem, issue **/usr/sbin/rsct/install/bin/recfgct** command to reconfigure RMC configuration on target system.

Refer to 6.5, “OS migration using NIM alt_disk_install feature” on page 128 for alt_disk_install method in our scenario.

5.3.6 CSM

You must run either AIX 5.2 or AIX 5.3 for the management server. For the managed nodes, you can run AIX 5.1.

Table 5-1 Coexistence for AIX and CSM levels in a cluster

Management server AIX	Management server CSM level	Managed node distribution	Managed node CSM level
AIX 5.2	CSM 1.4	AIX 5.1 AIX 5.2	For AIX 5.1, CSM 1.1.x, where x is 0 or higher For AIX 5.2 or AIX 5.3, CSM 1.3 with a Version, Release, Maintenance, FIX (VRMF) of 1.3.1.0 or higher, or CSM 1.4 with a VRMF of 1.4.0.1 or higher.
AIX 5.3	CSM 1.4	AIX 5.1 AIX 5.2 AIX 5.3	For AIX 5.1, CSM 1.1.x, where x is 0 or higher For AIX 5.2 or AIX 5.3, CSM 1.3 with a VRMF of 1.3.1.0 or higher, or CSM 1.4 with a VRMF of 1.4.0.1 or higher.

We suggest creating one VLAN for the CSM management server, managed devices, and hardware control points, and a separate VLAN for the CSM management server and cluster nodes. The recommended configuration for system management in CSM is as follows:

- **Management VLAN**: Hardware control commands such as **rpower** and **rconsole** are run on the management server and communicate to nodes through the management VLAN. The management VLAN connects the management server to the cluster hardware through an ethernet connection. For optimal security, the management VLAN must be restricted to hardware control points, remote console servers, the management server, and root

users. Routing between the management VLAN and cluster or public VLANs could compromise security on the management VLAN.

> **Note:** The management VLAN is subject to the RSA restriction of 10/100 Mb/s.

- **Cluster VLAN**: The cluster VLAN connects nodes to each other and to the management server through an ethernet connection. Installation and CSM administration tasks such as running **dsh** are done on the cluster VLAN. Host names and attribute values for nodes on the cluster VLAN are stored in the CSM database.
- **Public VLAN**: The public VLAN connects the cluster nodes and management server to the site network. Applications are accessed and run on cluster nodes over the public VLAN. The public VLAN can be connected to nodes through a second ethernet adapter in each node, or by routing to each node through the ethernet switch.

A physically separate CSM management server is safer than an LPAR management server that is part of a Central Electronics Complex (CEC) that can go down during a hardware or power failure. The following are general limitations and considerations for using an LPAR management server:

- The CSM management server can be brought down inadvertently by someone on the Hardware Management Console (HMC) deactivating that LPAR. Even someone without access to the CSM management server could power off the management server if they have access to the HMC. A system down can also be caused by someone moving resources such as CPU or I/O from that LPAR.
- If the firmware needs to be upgraded, the LPAR management server may go down along with the rest of the CEC for the firmware upgrade. However, upon bringing the CEC back up, the system will return to normal
- There is no direct manual hardware control of the CSM management server. The administrator must go through the HMC for power control of the management server.
- An LPAR management server may not have an attached display. This may affect the performance of your CSM GUIs.
- An LPAR management server may not contain media devices such as CD, tape, or diskette drives. This can affect your back-up strategy. In machines such as the p690, you may be able to assign a CD-ROM drive to the management server LPAR.
- An LPAR management server should not also be defined as a managed node.

AIX nodes must have AIX 5.2 or higher installed to enable Kerberos to use **rsh**. CSM does not support Kerberos Version 5 with Distributed Computing Environment (DCE).

Note: CSM 1.4.0 does not support Virtual I/O devices.

For detailed information about Cluster Systems Management (CSM), refer to the following documents:

- *IBM Cluster Systems Management for AIX 5L, Planning and Installation Guide, Version 1.4,* SA22-7919-07
- *IBM Cluster Systems Management for AIX 5L, Administration Guide, Version 1.4,* SA22-7918-07
- *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00

Refer to 6.1.4, “NIM and the CSM management server” on page 100, and 7.2.2, “NIM setup for the new environment” on page 138 for CSM configuration procedures in our scenarios.

5.3.7 WLM

An efficient use of WLM requires extensive knowledge of existing system processes and performance. Repeated testing and tuning will probably be needed before you can develop a configuration that works well for your workload. If you configure WLM with extreme or inaccurate values, you can significantly degrade system performance.

The process of configuring WLM is simpler when you already know one or more of the classification attributes of a process (for example, user, group, or application name). If these are unfamiliar, use a tool such as *topas* to identify the processes that are the top resource users and use the resulting information as the starting point for defining classes and rules

For detailed information about Workload Manager (WLM), refer to the following document:

- *AIX 5L Workload Manager (WLM),* SG24-5977-01

5.3.8 Dynamic Logical Partitioning

In the pSeries implementation of LPARs, you can dynamically add and remove resources (CPUs, memory, and I/O slots) to or from a partition while the operating system is running. For dynamic partition operations, the partition must be running AIX 5.2 or later, the managed system firmware must be a version

dated October 2002 or later, and HMC software must be Release 3 Version 1 or later.

You can also use the Dynamic Logical Partitioning Tool Set. This is a set of tools that enhance the usability of dynamic LPAR in AIX 5.2 and later running on pSeries servers.

When an LPAR is defined on the HMC with a partition profile, the “minimum”, ”desired” and “maximum” numbers of CPUs (and LMBs of memory) have to be specified. When the LPAR is activated and booted, the operating system instance will come up with the “desired” number of CPUs and LMBs if such amounts are available; else it will boot up with less. After the LPAR boots up, the number of CPUs and LMBs in that LPAR can be increased or decreased using dynamic LPAR operations. A dynamic LPAR operation will not be allowed if that operation will reduce a dynamic LPAR resource below the "minimum" value, or increase a dynamic LPAR resource above the "maximum" value.

An administrator can initiate a dynamic LPAR operation using the HMC GUI or with an HMC command. The HMC communicates with the affected AIX instances to release/add the resource, and the AIX instances communicate with the firmware to unassign/assign the resource. Optionally, unattended secure scripting via **`ssh`** can be set up for a userid on an AIX instance as an **`ssh`** client, so that this userid can **`ssh`** to an administrator userid on the HMC to issue HMC commands, e.g. **`chhwres`** and **`lshwres`**. With the dynamic LPAR Tool Set, we assume **`ssh`** is configured.

Dynamic LPAR Tool Set scripts enable the customer to easily use dynamic LPAR in the following scenarios:

- **Time-based usage scenario**: Consider a (simplified) customer environment with a batch partition and an interactive partition. During the day, the customer wants the interactive partition to have the bulk of the processors; and at night, the customer wants the processor resources to be moved to the batch partition. Once unattended ssh scripting between the AIX management instance and the HMC is set up, a cron job on the AIX management instance can be used to initiate the resource movements between the LPARs.
- **Load-based usage scenario**: Partitions in a user defined <hostList> file co-operate to share processor (and/or memory) resources based on load. If the load of one LPAR rises above a threshold, a Resource Manager script running in the AIX management instance will attempt to acquire a processor from the free pool. If no processor is available from the free pool, this script will choose the best donor LPAR from the <hostList>, probably the one with the least load. This Resource Manager will not initiate a dynamic LPAR add operation if doing so will put this LPAR above its "maximum allowed resource" as defined in this LPAR's "activated" profile. Similarly, it will not

initiate a dynamic LPAR remove operation if doing so will put this LPAR below its "minimum allowed resource".

The latest version of this toolset is v2.0.1.1. The dynamic LPAR Tool Set has several assumptions and prerequisites as follows:

- The toolset is meant to be run from a userid (usually that of a system administrator) on a AIX manager instance, which can be in one of the managed AIX LPARs, or in a separate machine.
- This toolset has been tested to work for ONE instance of p690, managed by ONE HMC, for ONE set of LPARs defined by the customer in a <hostList> file. This toolset should work for ONE instance of p670 or p630 also.
- The moveSlot.pl script works only with HMC software Release R3V2.4 and later.
- All the other main scripts work with old and new HMC software releases.
- **ssh** is set up from a userid (usually that of a system administrator) in the AIX management instance to an administrator userid on the HMC as described in the **ssh** section.
- The loginid in the AIX management instance has **rsh** privileges to all the hosts listed in the <hostList> file.
- The loginid in the AIX management instance knows the root password, and the root user on this AIX management instance has rsh privileges to all the hosts listed in the <hostList>. This is required by the script moveSlot.pl.
- The lparLsLoads.pl script can be used by any regular userid which has **rsh** privileges to all the hosts listed in the <hostList> file.
- Perl5 is required. It comes with AIX 5.2 and is available as /usr/bin/perl after installation.
- The /tmp filesystem on all the hosts mentioned in the <hostList> file must have some free space. This is needed by some AIX commands (e.g. /usr/bin/oslevel) used by the scripts.
- The partition names managed by the HMC must not contain a space or tab character.

Note: If you want to use dynamic LPAR Tool Set in AIX 5.3 or POWER5 environments, you must install version 2.0.2.0.

Refer to 6.3, “Dynamic LPAR operations” on page 118 for dynamic LPAR operations in our scenario.

5.3.9 Virtualization system technology

The following sections provide information about pSeries virtualization technology.

Micro-Partitioning

The workload running on other partitions in the shared processor pool can affect the amount of “wall clock” time required to process a given amount of work in another partition. In other words, the elapsed execution time is much less deterministic in a shared processor partition compared to a dedicated processor partition. The variation in execution time may be unacceptable for some applications. Additionally, partitions that use shared processor resources are not well suited to workloads that use a large amount of memory and depend on a high cache hit ratio for maximum performance.

The following limitations must be considered when implementing shared processor partitions:

- The limitation for a shared processor partition is 0.1 processing units of a physical processor. Therefore the number of shared processor partitions you can create for a system depends mostly on the number of processors of a system.
- The maximum number of partitions planned is 254.
- In a partition there is a maximum number of 64 virtual processors.
- A mix of dedicated and shared processors within the same partition is not supported.
- Only one shared processor pool is supported on a single system.
- Tuning parameters that should be considered for shared processor partitions include:
  - Number of processors assigned to the shared processor pool.
  - Amount of memory for each partition.
  - Number of virtual processors assigned to each partition (integer).
  - Entitled capacity for the virtual processors (fractional).
  - Capped or uncapped entitled capacity.
  - Shared weight (partition priority to use extra resources)
- If you dynamically remove a virtual processor you cannot specify a particular virtual CPU to be removed. The operating system will choose the virtual CPU to be removed.
- Shared processors may render AIX affinity management useless. Virtual processors may be dispatched on different physical processors during the time a partition is running. Therefore, there is no way to implement affinity domains. Memory is allocated to partitions in a round-robin fashion, and this

tends to reduce processor time consumption variability due to variation in memory allocation.

Virtual I/O Server

The Virtual I/O server partition is a special purpose partition ("hosting partition") that exists solely to provide virtual I/O resource sharing (Shared Ethernet Adapter and Virtual SCSI Server) to partitions within the POWER5 systems. The Virtual I/O Server partition owns the real resources (storage and LAN) that are shared between the partitions. Multiple Virtual I/O server partitions can run on a single server to provide redundancy and to spread the Virtual I/O server workload across multiple partitions.

The following is a list of minimum hardware requirements that must be available to create the Virtual I/O server:

- **POWER5 server**: The Virtual I/O capable machine
- **Hardware Management Console (HMC)**: HMC to create the partition and assign resources
- **Storage adapter**: The server partition needs at least one storage adapter
- **Physical disk**: If you want to share your disk to client partitions you need a disk large enough to make sufficient-sized logical volumes on it
- **Ethernet adapter**: If you want to securely route network traffic from a virtual Ethernet to a real network adapter
- **Memory**: At least 128 MB of memory

If a Virtual I/O server has to host a lot of resources to other partitions, you must ensure that enough processor power is available. In case of high load, or high traffic across virtual Ethernet adapters and virtual disks, partitions can observe delays in accessing resources.

The following limitations should be considered when implementing Virtual I/O server:

- The Virtual I/O Server supports the following operating system as a Virtual I/O client:
  - AIX 5L Version 5.3.
  - SUSE LINUX Enterprise Server 9 for POWER.
  - Red Hat Enterprise Linux AS for POWER Version 3
- The Virtual I/O server partition is intended only to be used to serve I/O to the client partitions. Therefore you cannot run any other application in the Virtual I/O server.

- You cannot install AIX or Linux in Virtual I/O server. The Virtual I/O server software is the only software supported in the partition.
- There is no hard limit on the number of Virtual I/O server partitions that can be used, but the amount of system resources available and the system workload put a practical limit on the number of Virtual I/O server partitions that can be used on a system.
- Since the Virtual I/O server is an AIX based appliance, redundancy for physical devices attached to the Virtual I/O server can be provided by using capabilities like LVM mirroring, Multipath I/O, and Etherchannel.

Virtual Ethernet

Virtual Ethernet allows the administrator to define in-memory connections between partitions. These connections appear to the operating system to be high-bandwidth Ethernet connections

The following limitations must be considered when implementing Virtual Ethernet:

- A maximum of up to 256 Virtual Ethernet adapters are permitted per partition.
- Virtual Ethernet can be used in both shared and dedicated processor partitions provided the partition is running IBM AIX 5L Version 5.3.
- A mixture of Virtual Ethernet connections, real network adapters or both are permitted within a partition.
- Virtual Ethernet can only connect partitions within a single system.
- Virtual Ethernet requires a POWER5 system and an HMC to define the Virtual Ethernet adapters.
- Virtual Ethernet uses the system processors for all communication functions instead of off loading that load to processors on network adapter cards. As a result, there is an increase in system processor load using Virtual Ethernet.

Shared Ethernet Adapter

Shared Ethernet Adapter is a new service that acts as a layer 2 network switch to securely route network traffic from a Virtual Ethernet Adapter to a real network adapter. The Shared Ethernet Adapter service must run in a Virtual I/O server partition. It cannot run in a general purpose AIX 5L Version 5.3 partition.

The following limitations must be considered when implementing Shared Ethernet Adapters in the Virtual I/O server:

- In order to bridge network traffic between the Virtual Ethernet and external networks, the Virtual I/O server partition has to be configured with at least one physical Ethernet adapter. One Shared Ethernet Adapter can be shared by

multiple VLANs and multiple subnets can connect using a single adapter on the Virtual I/O server.

- Because Shared Ethernet Adapter depends on Virtual Ethernet which uses the system processors for all communications functions, a significant amount of system processor load can be generated by the use of Virtual Ethernet and Shared Ethernet Adapter.
- One of the virtual adapters in the Shared Ethernet Adapter on the Virtual I/O server must be defined as the default adapter with a default Physical Volume Identifier (PVID). This virtual adapter is designated as the PVID adapter. Ethernet frames without any VLAN ID tags are assigned the default PVID and directed to this adapter.
- Do not use the Shared Ethernet Adapter functionality of the Virtual I/O server if you expect heavy network traffic between Virtual LANs and local networks. Use a dedicated network adapter instead.
- Up to 16 Virtual Ethernet adapters with 18 VLANs on each can be shared on a single physical network adapter. There is no limit on the number of partitions that can attach to a VLAN so the theoretical limit is very high. In practice, the amount of network traffic will limit the number of clients that can be served through a single adapter.
- For redundancy, configure two Virtual Ethernet paths to two separate Virtual I/O servers and use the AIX 5L multipath routing and dead gateway detection functionality to automatically direct network traffic to an available Virtual I/O Server partition. Figure 5-1 on page 79 shows an example using multipath routing and dead gateway detection. In the client partition, two default routes with dead gateway detection are defined. One default route goes to gateway 9.3.5.10 via Virtual Ethernet adapter with address 9.3.5.12. The second route goes to gateway 9.3.5.20 using the Virtual Ethernet adapter with address 9.3.5.22. In the case of a failure of the primary route, access to the external network will be provided through the second route. AIX will detect route failure and adjust the cost of the route accordingly.

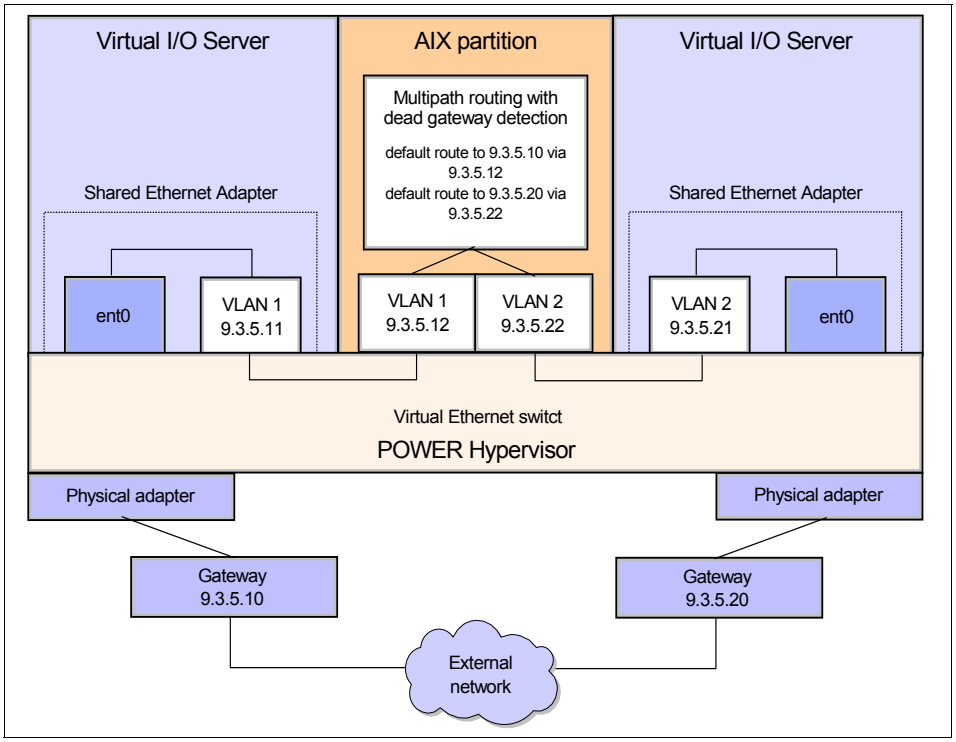


Figure 5-1 Configuration with multipath routing and dead gateway detection

Restriction: It is important to note that multipath routing and dead gateway detection do not make an IP address highly available. In case of the failure of one path, dead gateway detection will route traffic through an alternate path. The network adapters and their IP addresses remain unchanged. Therefore, when using multipath routing and dead gateway detection, only your access to the network will become redundant, but not the IP addresses.

You can also configure Etherchannel to provide highly available access to Virtual Ethernets. Figure 5-2 on page 80 shows an example using Etherchannel backup adapter. If the primary adapter fails the Etherchannel will automatically switch to the backup adapter. The IP address of the client server partition which is configured on the Etherchannel network interface will remain available.

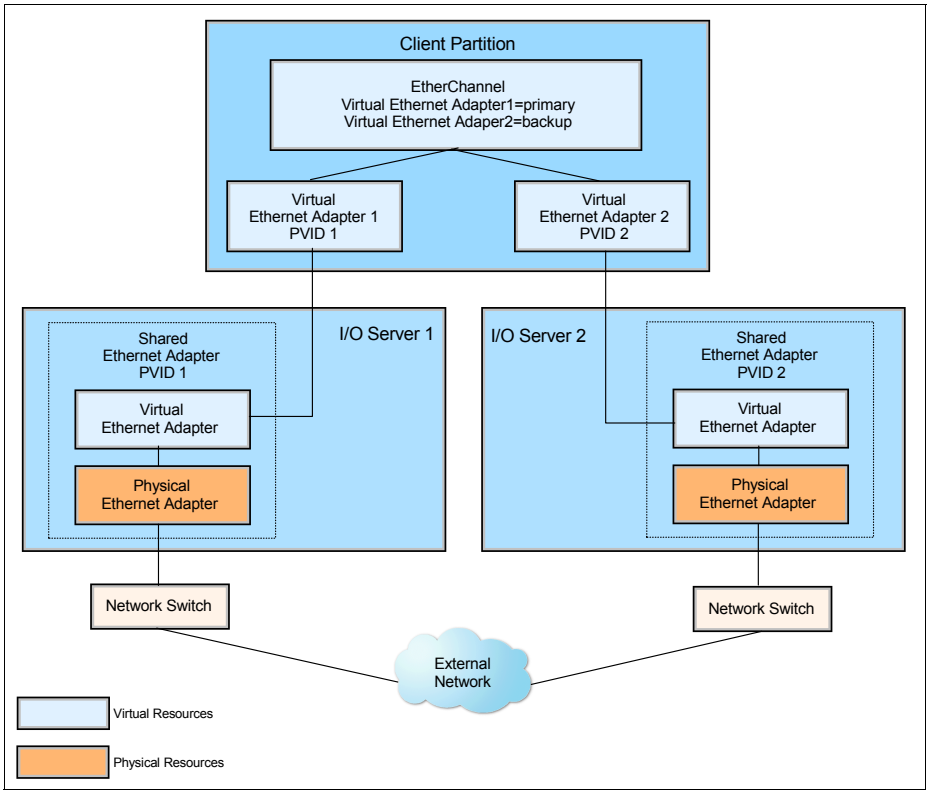


Figure 5-2 Virtual I/O Server configuration with Etherchannel backup adapter

Note: When using the Etherchannel with two adapters and configuring one adapter as backup, no aggregation resulting in higher bandwidth will be provided. No network traffic will go through the backup adapter unless there is a failure of the primary adapter.

Virtual SCSI

Virtual SCSI is a new service to meet the storage needs of client partitions. Real storage devices are attached to a “Virtual I/O server partition”. The Virtual I/O server partition defines logical volumes that are exported to client systems. On the client systems, these logical volumes appear to be SCSI disks.

The following should be considered when implementing Virtual SCSI:

- In general most types of storage can be used to host the Virtual SCSI disks including all AIX 5L supported SAN, SCSI and RAID devices. There are two exceptions to this rule: iSCSI and SSA disks are not supported.

- Virtual SCSI itself does not have any limitations in terms of number of supported devices or adapters. However the Virtual I/O Server supports a maximum of 65535 virtual I/O slots. A maximum of 256 virtual I/O slots can be assigned to a single partition.
- Every I/O slot needs some resources to be instantiated. Therefore the size of the Virtual I/O Server puts a limit to the number of virtual adapters that can be configured.
- The SCSI protocol defines mandatory and optional commands. While Virtual SCSI supports all the mandatory commands not all optional commands are supported.
- There are performance implications when using virtual SCSI devices. It is important to understand that due to the overhead associated with POWER Hypervisor calls virtual SCSI will use additional CPU cycles when processing I/O requests. When putting heavy I/O load on virtual SCSI devices this means you will use considerably more CPU cycles. Provided that there is sufficient CPU processing capacity available the performance of virtual SCSI should be comparable to dedicated I/O devices.
- Since VSCSI is a server/client model, the CPU utilization will always be higher than doing local I/O. A reasonable expectation is a total of twice as many cycles to do VSCSI as a locally attached disk I/O. This is more or less evenly distributed between the client and server.
- If multiple partitions are competing for resources from a VSCSI server, care must be taken to ensure enough server resources (CPU, memory, and disk) are allocated to do the job.
- There is no data caching in memory on the server partition. Therefore, all I/Os that it services are essentially synchronous disk I/Os. Because there is no caching in memory on the server partition, its memory requirement should be modest.
- Partitions with high performance and disk I/O requirements are not recommended for implementing VSCSI. Partitions with very low performance and disk I/O requirements can be configured at minimum expense to use only a logical volume. Using a logical volume for virtual storage means that the number of partitions is no longer limited by hardware, but the trade-off is that some of partitions will have less than optimal storage performance. The suitable applications for VSCSI might be the boot disks for the operating system or Web servers that will typically cache a lot of data.
- For redundancy, use AIX 5L MPIO to configure two paths through two Virtual I/O Servers to the same physical disks. and use AIX 5L LVM to mirror a logical volume to two separate Virtual I/O servers as shown in Figure 5-3 on page 82 (This data resides on two different physical disks).

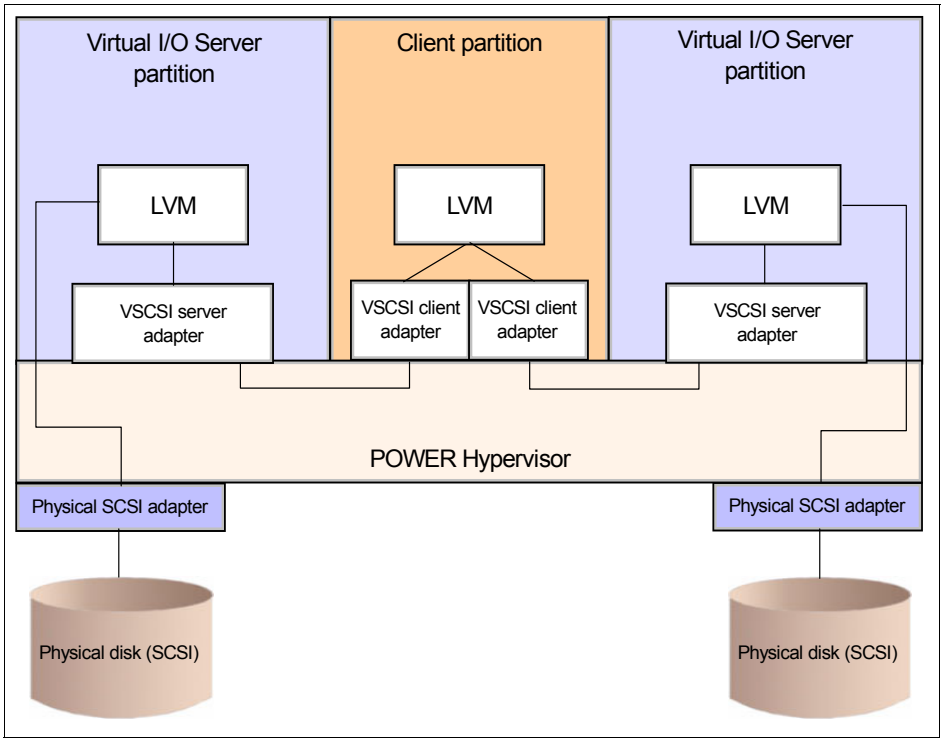


Figure 5-3 Virtual I/O Server configuration with LVM mirroring

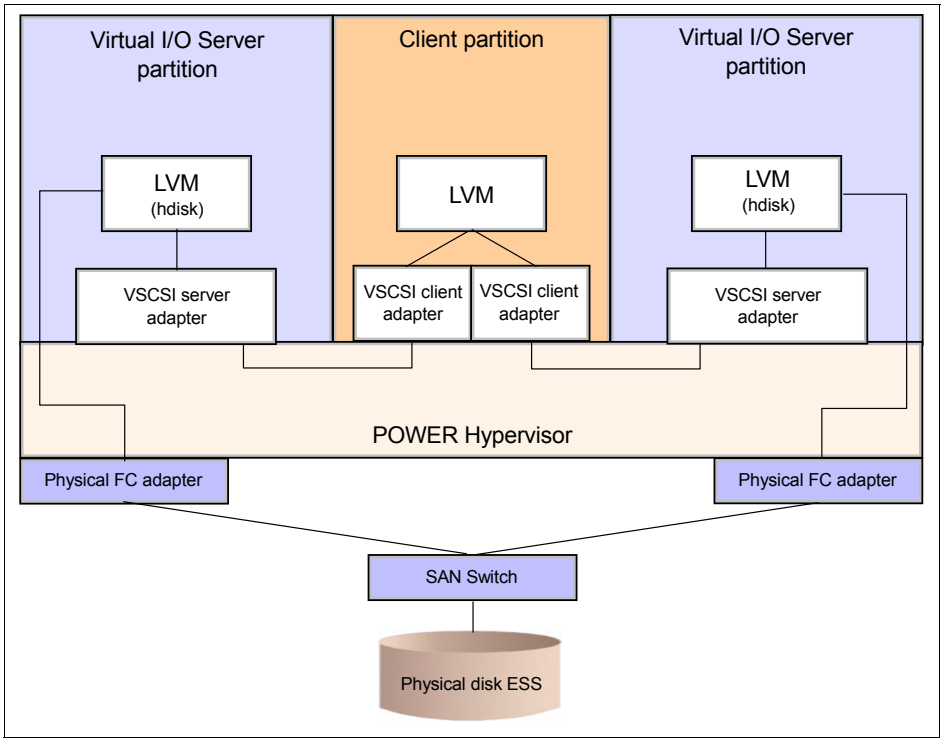


Figure 5-4 Virtual I/O Server configuration with Multi-path I/O

Note: For optimum stability, the Virtual I/O server product requires the system microcode to be at level SF 220_049 or greater, which can be obtained from the following Web site:

`http://techsupport.services.ibm.com/server/mdownload/download.html`

The Virtual SCSI function is fully supported for production use with APAR IY58231. Also, APAR IY58143 delivers fixes for the Virtual I/O Server Version 1.1 that are required for optimal system stability and performance. These APARs can be obtained from the following Web site:

`http://techsupport.services.ibm.com/server/virtualization/vios/download`

For detailed information about virtualization system technology, refer to the following documents:

- *Introduction to Advanced POWER Virtualization on IBM p5 Servers,* SG24-7940-00

- *Advanced POWER Virtualization on IBM @server p5 Servers Architecture and Performance Considerations*, SG24-5768-00

Refer to 7.3, “Virtual I/O devices” on page 146 for VIO configuration procedures in our scenario.

5.3.10 Capacity on Demand

Capacity on Demand (CoD) for @server POWER5 (550 and 570) and pSeries (p650, p670 and p690) servers with dynamic logical partitioning offers clients the ability to non-disruptively activate (no reboot required) processors and memory. You can activate inactive processors or memory units that are already installed on your server on a temporary or permanent basis.

Capacity on Demand also gives @server POWER5 and pSeries owners the option to temporarily activate processors to meet intermittent performance needs, to activate additional capacity on a trial basis and to access capacity to support operations in times of crisis.

The types of Capacity on Demand for @server POWER5 and pSeries servers are:

- **Capacity Upgrade on Demand** (Permanent capacity for nondisruptive growth)

  Capacity Upgrade on Demand for @server POWER5 and pSeries servers allows companies to install inactive CUoD processors and memory at an extremely attractive price and then bring new capacity online quickly and easily. With AIX 5L Versions 5.3 and 5.2, processors and memory can be activated dynamically without interrupting system or partition operations.

  Capacity Upgrade on Demand processor options for p670 and p690 servers are available in units of four active and four inactive processors so up to 50% of the system processors and memory can be inactive. With Capacity Upgrade on Demand for the p650 server, processors are available in pairs with a maximum of six inactive processors. As your workload demands require more processing power, you can activate inactive processors in pairs simply by placing an order for an activation feature code, sending current system configuration data to IBM. An electronically encrypted activation key that unlocks the desired number of processors is then sent over the internet. There is no hardware to ship and install, and no additional contract is required.

  Memory activation works the same way. Capacity Upgrade on Demand memory is available in various sizes for the p650, p670 and p690 systems.

You can activate memory in 4GB increments by ordering an activation feature code for the desired amount of memory.

The p5-550 and p5-570 servers offer increased activation granularity for both processors and memory (p5-570). Processors can be activated in increments of 1 processor while memory can be activated in increments of 1GB.

- **On/Off Capacity on Demand** (Temporary capacity for fluctuating workloads)

  For temporary workloads, pSeries 650, 670, and 690 servers offer innovative and flexible processor activation. When you order an On/Off Capacity on Demand feature, you receive an activation key that enables you to activate two processors for 30 days of use. You can turn processor pairs on and off whenever you need to. Deductions are made against the Processor Day allocation only when processors are activated. Increments of usage are measured in Processor Days, and the minimum usage is one day per activated processor.

  On/Off CoD for p5-550 and p5-570 servers allows self-managed temporary activation of CUoD processor and memory resources. The user can turn on and then turn off resources as needed. The system monitors the amount and duration of the activations and generates a usage report. Billing for the activations is then based on the usage report.

  To temporarily activate some or all of your inactive processors or memory units using On/Off Capacity on Demand, you need to order an On/Off Capacity on Demand enablement feature. An On/Off CoD enablement code permits the temporary use of a limited number of processor or memory days. You can make requests for temporary capacity over the life of the machine as long as your total requests do not exceed this limit. When the limit is reached, a new On/Off CoD enablement feature must be ordered and a new enablement code entered on your server. Each time a new enablement code is entered, it will reset the limit of processor or memory days that can be requested as temporary capacity.

  Table 5-2, and Table 5-3 on page 86 list the server models, processor feature codes, memory feature codes, and billing feature codes that can be temporarily activated using On/Off Capacity on Demand.

Table 5-2 On/Off CoD processor feature codes and billing feature codes

M/T and Model	Orderable F/C	Billing F/C
9113-550	5237	7931
9117-570	7830	7952
9117-570	7832	7953
9117-570	7833	7955

M/T and Model	Orderable F/C	Billing F/C
9406-550	8958	7931
9406-570	8961	7952
9406-570	8962	7953
9406-570	8971	7952

Table 5-3 On/Off CoD memory feature codes and billing feature codes

M/T and Model	Orderable F/C	Billing F/C
9117-570	7890	7957
9406-570	7890	7957

- **Reserve Capacity on Demand** (Temporary capacity for fluctuating workloads)

  Reserve CoD allows p5-550 and p5-570 servers to have optimized, automatically managed temporary activation of CUoD processors. The user purchases a block of 30 Processor Days of usage time and then assigns inactive processors to the shared processor pool. The server then automatically manages the workload and only charges against the Processor Day account when the workload requires over 100% of the base (permanently activated) processing power.

  To purchase a quantity of processor days in advance for Reserve Capacity on Demand, you need to order one or more reserve capacity prepaid features. Each reserve capacity prepaid feature represents a number of processor days.

  Based on the number of reserve capacity prepaid features ordered, a reserve capacity prepaid code is generated.

  A reserve capacity prepaid code enables the server to use a limited number of processor days as reserve capacity. You can make requests for temporary capacity using Reserve CoD over the life of the server as long as reserve processor days remain on your server. When all the reserve processor days have been used, the server sends a message and automatically removes the reserve processors from the shared processor pool.

  Table 5-4 lists the server models, reserve CoD processor feature codes and prepaid processor activation feature codes for Reserve Capacity on Demand.

Table 5-4 Reserve CoD and prepaid processor activation feature codes

M/T and Model	Processor F/C	Prepaid processor activation F/C
9113-550	5237	7934
9113-550	5238	7934
9117-570	7830	7956
9117-570	7832	7959
9117-570	7833	7959

- **Trial Capacity on Demand** (Temporary capacity for workload peaks and testing)

  Owners of @server POWER5 and pSeries systems with Capacity Upgrade on Demand capabilities can activate these resources once, for up to 30 contiguous days, at no charge. This Trial Capacity on Demand feature does not require any special activation keys. It can be used on pSeries p650, p670, and p690 servers to meet an immediate need for additional resources or to give inactive processor and memory resources a test run. @server POWER5 550 and 570 servers can enable the trial by registering at the pSeries CoD Web site and electronically receiving an activation key.

- **Capacity BackUp for Disaster Recovery** (Interim capacity for continued operation)

  Capacity BackUp for pSeries servers (p670 and p690 only) can provide emergency processing capacity for up to 30 days in the event of lost capacity in part of your operation, helping you recover by adding reserved capacity on a designated pSeries system. Capacity BackUp is intended for companies requiring an off-site disaster recovery machine at an extremely affordable price. Using On/Off Capacity on Demand capabilities, Capacity BackUp offerings have a minimum set of inactive processors that can be used for any workload, as well as a large number of inactive processors that can be used at no charge in the event of a disaster.

  Inactive processors cannot be permanently activated. In addition to no-charge usage in case of a disaster, inactive processors can be used on a for-charge basis for production role swapping during an unscheduled outage, tape backup, failover testing and role swapping during upgrades or PTF installations.

  Capacity BackUp offerings are not intended or priced to be used as full-time backup servers for 24x7 high-availability solutions. High usage of inactive processors for purposes other than a disaster may add significant cost to an IT operation.

You can order the CoD activation code from the following IBM Web site:

http://www-1.ibm.com/servers/eserver/pseries/ondemand/cod/activation.html

5.3.11 Simultaneous multi-threading

Simultaneous multi-threading is a POWER5 microprocessor capability that allows two threads to be executed at the same time on a single CPU (processor). Simultaneous multi-threading can significantly improve performance, particularly for commercial applications.

Simultaneous multi-threading is not well suited to all workloads. Typically, applications that use nearly all of a processor's physical compute or bandwidth capacity will see little benefit from Simultaneous multi-threading. Simultaneous multi-threading can also increase the variability in execution time (elapsed wall clock time) from run to run.

Workloads that do not use the full compute (processing) capability or the full bandwidth of the POWER5 chip should benefit from Simultaneous multi-threading. Commercial workloads such as databases or Web servers typically meet this profile and should see benefit from Simultaneous multi-threading. Many High Performance Computing (HPC) applications can also benefit from Simultaneous multi-threading. but applications that are compute and/or memory intensive typically do not benefit from Simultaneous multi-threading.

All processors in a partition must have Simultaneous multi-threading either enabled or disabled. Even if Simultaneous multi-threading is enabled, AIX 5L V5.3 will only use Simultaneous multi-threading if there are more active threads than there are physical processors.

For more information about Simultaneous multi-threading, refer to the following documents:

- *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00
- *Introduction to Advanced POWER Virtualization on IBM p5 Servers,* SG24-7940-00
- *Advanced POWER Virtualization on IBM @server p5 Servers Architecture and Performance Considerations*, SG24-5768-00

5.3.12 Partition Load Manager (PLM)

The Partition Load Manager software is part of the Advanced POWER Virtualization feature and provides automated CPU and memory resource management across dynamic LPAR capable logical partitioning running AIX 5L. Partition Load Manager allocates resources to partitions on-demand, within the constraints of a user-defined policy. Partitions with a high demand for resources

are given resources from partitions with a lower demand, improving the overall resource utilization of the system. Resources that would otherwise be unused, if left allocated to a partition that was not utilizing them, can now be used to meet resource demands of other partitions in the same system.

Partition Load Manager works much like any other system management software in that it allows you to view the resources across your partitions, group those resources into manageable chunks, allocate and reallocate those resources within or across the groups, and locally log activity on the partitions.

PLM can be used to adjust memory resources for shared processor partitions or to change the entitled processor capacity, number of virtual processors and/or the processor share (prior for uncapped partitions) for a shared processor partition.

To configure resource management for AIX partitions with Partition Load Manager, follow these steps:

1. Install Partition Load Manager
2. Install OpenSSH Software Tools
3. Configure RMC
4. Configure the policy file
5. Verify the installation of OpenSSH and RMC

For a more detailed procedure for configuring PLM, see document:

http://publib.boulder.ibm.com/infocenter/eserver/v1r2s/en_US/info/iphbk/iphbk.pdf

The following should be considered when managing your system with the Partition Load Manager:

- PLM can be used in partitions running AIX 5L V5.2 ML04 or AIX5L V5.3. Linux and i5OS are not supported.
- At least one PLM group must be created. All partitions in a group must have the same processor type.
- A single instance of Partition Load Manager can only manage a single server. However, multiple instances of Partition Load Manager can be run on a single system, each managing a different server.
- One instance of Partition Load Manager server can manage multiple groups of partitions. All of the partitions in a group must be of the same type, either all dedicated processor partitions or shared processor partitions. One Partition Load Manager server can manage multiple groups of dedicated processor partitions and multiple groups of shared processor partitions. Partition Load Manager does not move resources between groups.

- Partition Load Manager cannot move I/O resources between partitions. Only processor and memory resources can be managed by Partition Load Manager.
- Partition Load Manager requires HMC Release 3 Version 2.6 or higher on the HMC.

Note: Partition Load Manager handles memory and both dedicated and shared processor partitions. These two types of partitions have to be defined in different PLM groups.

- Any CPU or memory values specified in the policy must be compatible with the partition's HMC partition definition. Partition Load Manager is not able to decrease a partition's minimum below the HMC's minimum, nor increase a partition's maximum over the HMC's maximum. System administrators are responsible for ensuring Partition Load Manager policies and HMC partition definitions are compatible.

Partition Load Manager requires an **ssh** connection to the HMC and network connections to the managed partitions to communicate over RMC. If you meet a problem related to SSH and RMC connection, refer to the following steps for verifying the installation of OpenSSH and RMC:

Verifying the OpenSSH installation

To verify OpenSSH installation and setup, use the HMC user name which you used in the setup of OpenSSH in place of HMC_USER, and the HMC host name in place of HMC_HOST in the following command:

```
ssh HMC_USER@HMC_HOST ls
```

If the setup was done correctly, a listing of the HMC_USER's home directory is displayed. If this home directory listing is not displayed, do the following:

1. Check for public keys. In the Partition Load Manager user's home directory, there is a ~/.ssh directory. In that directory, there is an *id_rsa.pub file*. This file contains the user's public key. Perform a cat on the file with the following command:

   ```
   cat id_rsa.pub
   ```

   On the HMC, under the HMC_USER's home directory, there is a ~/.ssh directory. In that ~/.ssh directory, there is an *authorized_key2* file. This file contains the public keys of all hosts that are authorized to **ssh** as this user. Perform a cat on the file with the following command:

   ```
   cat authorized_key2
   ```

There should be an entry that matches the contents of the *id_rsa.pub* file in the Partition Load Manager user's ~/.ssh directory. If not, then append the contents of the *id_rsa.pub* file to the *authorized_key2* file.

2. Check for known hosts. In the Partition Load Manager user's home directory, there is a ~/.ssh directory. In that directory, you should find a *known_hosts* file. This file contains the hosts that Partition Load Manager has performed an OpenSSH setup with. Perform a cat on the file with the following command:

   ```
   cat known_hosts
   ```

   The output should contain an entry for the HMC_HOST. The host name listed for the HMC should be the one used when you are running OpenSSH commands. For example, if the host name listed is hmc_name.ibm.com, then use hmc_name.ibm.com.

3. Check file permissions. On both the Partition Load Manager server and the HMC_HOST machine, make sure that the file permissions are set as follows:
   - Directories and key files: write bits turned off for group and world.
   - Private keys: set permissions to 600.

Verify the RMC setup

To verify the RMC setup, run the following as the Partition Load Manager user for each of the partitions that were used with the plmsetup script. Replace PART_HOST with the name of the partitions in the following:

```
CT_CONTACT=PART_HOST lsrsrc IBM.LPAR
```

If setup was done correctly, the persistent attributes of the resource class will be displayed. If not, then try the following steps:

1. Perform host-based authentication. On both the Partition Load Manager server and the partition, run the following command:

   ```
   /usr/sbin/rsct/bin/ctsvhbal
   ```

   A list of identities are displayed. These are identities as which the known partition host can be identified. On each machine, run the following command:

   ```
   /usr/sbin/rsct/bin/ctsthl -l
   ```

   On the Partition Load Manager server, there is an entry for the partition. On the partition, there is an entry for the Partition Load Manager server. The HOST_IDENTITY value should match one of the identities listed in the respective ctsvhbal command output. If an entry does not match, remove it by running the following command:

   ```
   /usr/sbin/rsct/bin/ctsthl -d -n HOST_IDENTITY
   ```

   After the entry is removed, add the new entry using the identity listed in the **ctsvhbal** command output by running the following command:

```
/usr/sbin/rsct/bin/ctsthl -a -n IDENTITY -m METHOD -p ID_VALUE
```

The value for the METHOD parameter can be obtained from the **ctsthl** command. Look for an entry for the machine itself. In that entry, there is an Identifier Generation Method field. This is the value you must use. One example is rsa512. For the ID_VALUE parameter value, use the Identifier Value field in the same entry.

2. On partitions, check the ACL file. In the /var/ct/cfg/ctrmc.acls file, there is a stanza for IBM.LPAR towards the end of the file that looks similar to the following:

```
IBM.DLPAR plmuser@server.ibm.com * rw
```

The user should match the name of the user to run Partition Load Manager. The host name should match what was returned by the **ctsvhbal** command which was run on the Partition Load Manager server machine. If it does not, run the **plmsetup** script again, this time using the IDENTITY provided by the **ctsvhbal** command.

For detailed information about the Partition Load Manager, refer to the following documents:

- *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, SG24-7940-00
- *Advanced POWER Virtualization on IBM @server p5 Servers Architecture and Performance Considerations*, SG24-5768-00
- *AIX 5L Version 5.3 Partition Load Manager Guide and Reference*

Refer to 7.5.1, “PLM installation and configuration” on page 151 for PLM configuration procedures in our scenario.

5.3.13 HACMP

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

HACMP 5.2.0 requires AIX 5L on pSeries, Cluster 1600, or RS/6000 servers with at least four slots. HACMP V5 supports new p5 models, as well as all previously announced pSeries servers. The specific requirements for AIX 5L are:

Table 5-5 HACMP versus AIX software matrix

HACMP with hwd	HACMP V 5.1 with AIX V5.2	HACMP V5.1 with AIX V5.3	HACMP V5.2 with AIX V5.2	HACMP V5.2 with AIX V5.3
@server p5 520, 9111-520	HACMP APAR IY60534 HACMP APAR IY59022 HACMP APAR IY53044 AIX V5.2 5200-04 RMP AIX APAR IY56554 AIX APAR IY61014	HACMP APAR IY60534 HACMP APAR IY56436 HACMP APAR IY59022 HACMP APAR IY53044 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191	HACMP APAR IY60340 HACMP APAR IY58496 AIX V5.2 5200-04 RMP AIX APAR IY56554 AIX APAR IY61014	HACMP APAR IY60340 HACMP APAR IY58496 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191
@server p5 550, 9113-550	HACMP APAR IY60534 HACMP APAR IY59022 HACMP APAR IY53044 AIX V5.2 5200-04 RMP AIX APAR IY56554 AIX APAR IY61014	HACMP APAR IY60534 HACMP APAR IY56436 HACMP APAR IY59022 HACMP APAR IY53044 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191	HACMP APAR IY60340 HACMP APAR IY58496 AIX V5.2 5200-04 RMP AIX APAR IY56554 AIX APAR IY61014	HACMP APAR IY60340 HACMP APAR IY58496 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191
@server p5 570, 9117-570	HACMP APAR IY60535 HACMP APAR IY59022 HACMP APAR IY53044 AIX V5.2 5200-04 RMP AIX APAR IY56554 AIX APAR IY61014	HACMP APAR IY60535 HACMP APAR IY56436 HACMP APAR IY59022 HACMP APAR IY53044 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191	HACMP APAR IY60341 HACMP APAR IY58496 AIX V5.2 5200-04 RMP AIX APAR IY56554 AIX APAR IY61014	HACMP APAR IY60341 HACMP APAR IY58496 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191

HACMP with hwd	HACMP V 5.1 with AIX V5.2	HACMP V5.1 with AIX V5.3	HACMP V5.2 with AIX V5.2	HACMP V5.2 with AIX V5.3
HACMP V5 on AIX 5.3 (ALL H/W)		HACMP APAR IY56436 HACMP APAR IY59022 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191		HACMP APAR IY58496 RSCT APAR IY61770 AIX APAR IY61034 AIX APAR IY60930 AIX APAR IY62191

Restriction: HACMP does not support micro-partitioning, Virtual SCSI (VSCSI) or Virtual LAN (VLAN) on p5 models at this time. The p5 520, 550 and p5 570 integrated serial ports are not enabled when the HMC ports are connected to a Hardware Management Console. Either the HMC ports or the integrated-serial ports can be used, but not both. Moreover, the integrated serial ports are supported only for modem and asynchronous terminal connections. Any other applications using serial ports, including HACMP for heartbeat, requires a separate serial port adapter to be installed in a PCI slot.

For more details on HACMP 5.2, refer to the following documents:

- *HACMP Concepts and Facilities Guide, Version 5.2,* SC23-4864-03
- *HACMP Planning and Installation Guide, Version 5.2,* SC23-4861-03
- *HACMP Administration and Troubleshooting Guide, Version 5.2,* SC23-4862-03

6

POWER4 provisioning scenario

In this chapter, we describe the scenario for provisioning tools on POWER4 systems. We use different AIX based tools and technology that can be used to automate provisioning in the customer environment. In this scenario we focus on tools supplied with AIX 5.2 and POWER4 systems. The benefits of AIX 5.3 are covered in Chapter 7, “POWER5 provisioning scenario” on page 135.

In this scenario, we utilize these technologies:

- Network Installation Manager (NIM)
- Cluster Systems Management (CSM)
- Dynamic LPAR ToolSet
- Resource Monitoring and Control (RMC)

We do not intend to describe all features and capabilities of the tools. They are covered in other Redbooks and Whitepapers. However, we set them up together to make the environment ready for provisioning.

6.1 Preparation of the environment

Provisioning is a challenging process. To create a fully automated computing environment, you have to do a lot of work to prepare the hardware and software. But once everything is in place, the system management and orchestration will be much easier and faster than the initial work. There is no tool on the market for the pSeries systems to prepare entire environment for provisioning. Therefore we use generally available tools shipped with the AIX operating environment. Some of them, CSM and the dynamic LPAR ToolSet, require special licensing fees, but these tools are widely used with pSeries systems.

In this scenario, we use the existing systems in our laboratory:

- An IBM p690 system with two LPARs running AIX 5.2.
- An IBM p630 management server running AIX 5.3.
- A Hardware Management Console (HMC)

For a diagram of our environment, refer to Figure 6-1.

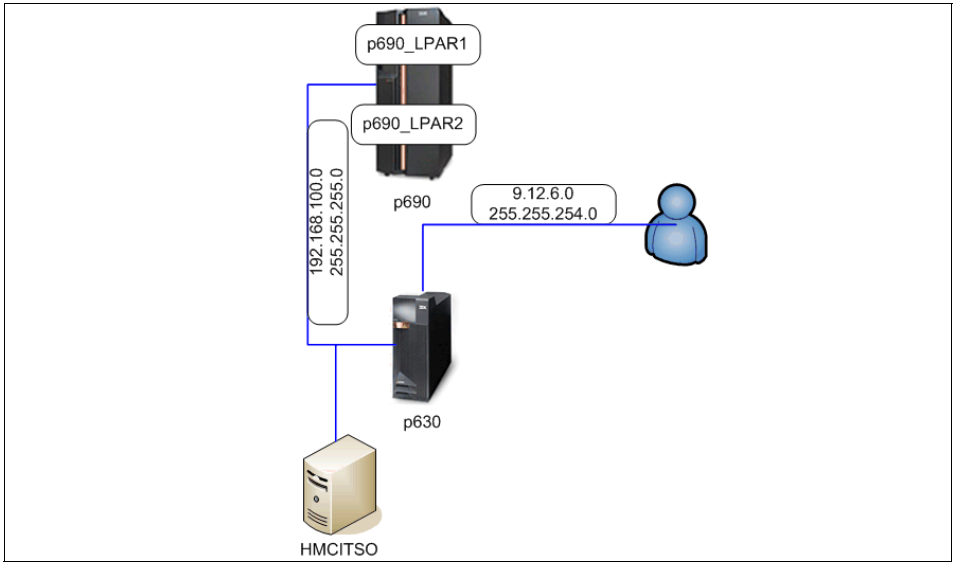


Figure 6-1 POWER4 scenario diagram

Although we install only two LPARs in our scenario, we designed the workflow to be use with as many systems ready for provisioning in a changing customer environment.

6.1.1 Hardware preparation

There is limited possibility to automate the hardware installation. Although IBM's achievement in hardware virtualization and system granularity allows us to create a very flexible solution. Dynamic LPARs provide a very convenient way to create the system ready to be reconfigured for our application need. The POWER4 system allows us to create a single or multiple CPU partitions with the required IO adapters and memory.

The p690 system is already configured in a rack, the power must be connected. We set up the Storage Area Network (SAN) and Local Area Network (LAN). Because our scenario is not very complex and for lacked of resources, we decided not to create separate VLANs for the management network and client stations, although it is a good idea to do so. For the recommended VLAN configuration for the proper system management in CSM configuration refer to 5.3.6, "CSM" on page 70.

For the network setup in our environment see Figure 6-2.

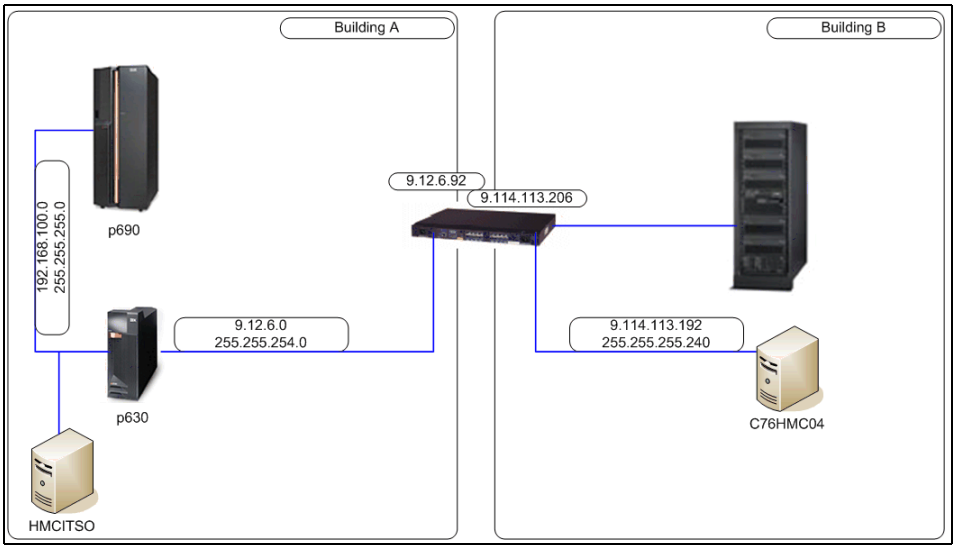


Figure 6-2 LAB network diagram

6.1.2 Hardware Management Console (HMC) setup

The Hardware Management Console (HMC) is installed and connected to the network. We recommend you to update the HMC software to the latest level. You can do this via **`Software Maintenance -> HMC -> Install Corrective Services`** menu in HMC graphical console as shown in Figure 6-3 on page 98:

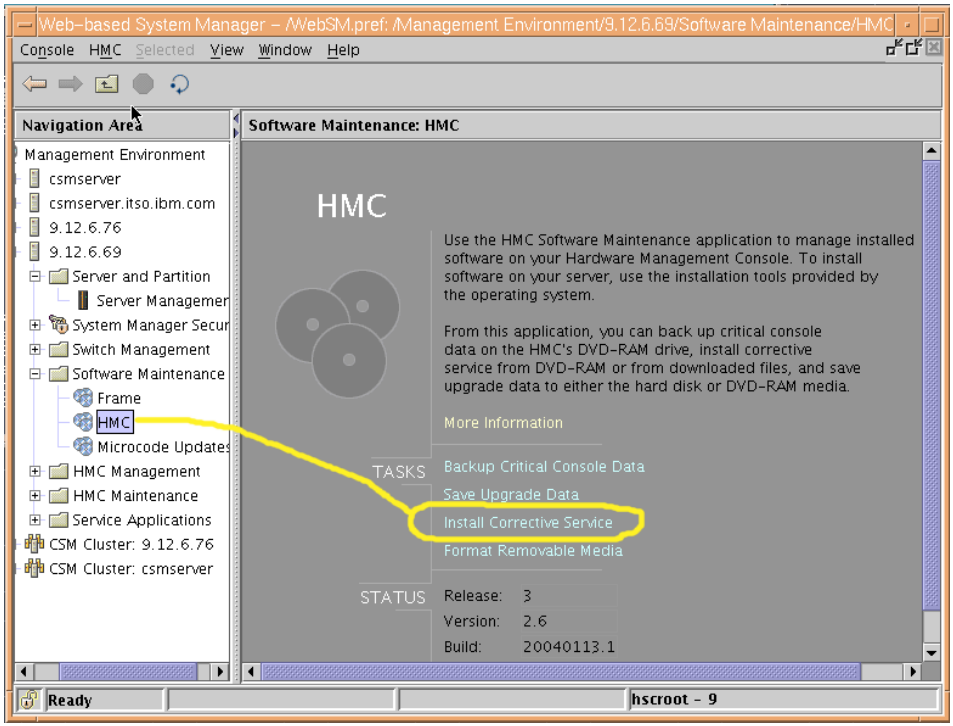


Figure 6-3 Install Corrective Service menu

In this scenario, we share the p690 system with several p650 nodes, so we used IBM 8-port asynchronous adapter for RS232 connectivity.

6.1.3 Creation of partitions

For the scenario, we created two partitions:

- A p690_LPAR1 with:
  - 4 CPUs
  - 4 GB RAM

For a detailed resource list, see Figure 6-4 on page 99.

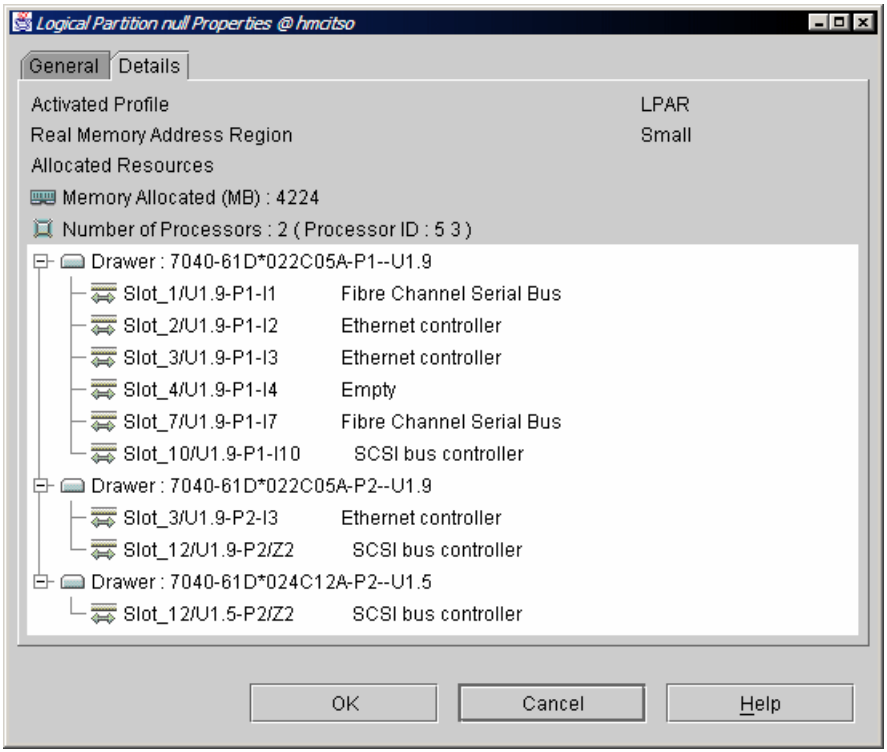


Figure 6-4 LPAR1 resources

- A p690_LPAR2 with:
  - 4 CPUs
  - 4 GB RAM

For a detailed resource list, see Figure 6-5 on page 100.

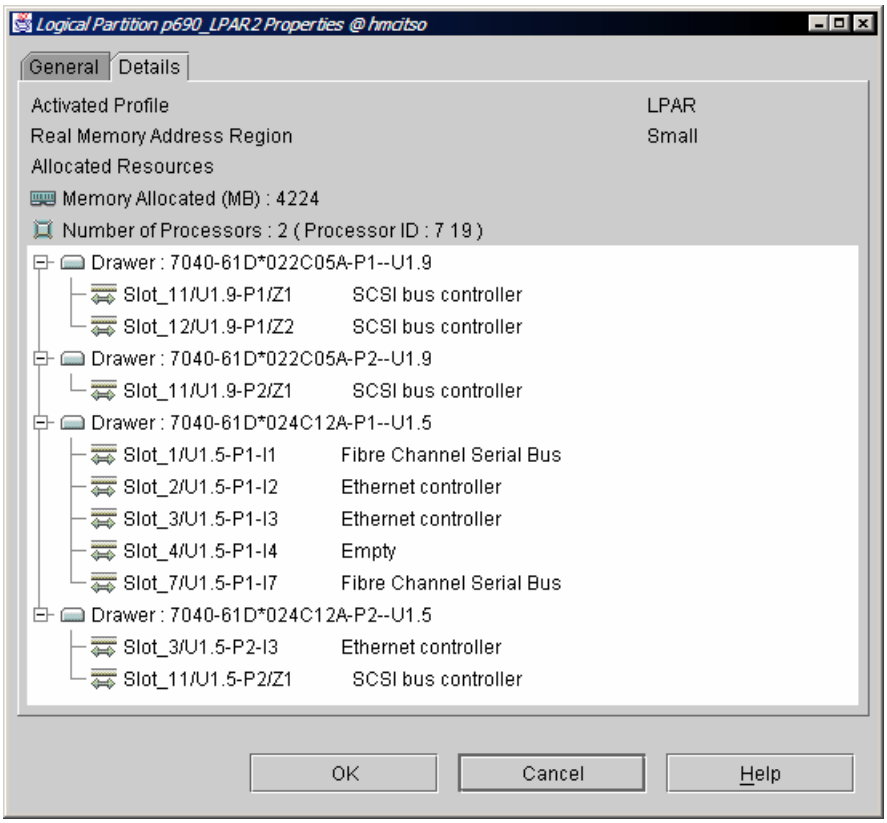


Figure 6-5 Resources of LPAR2

6.1.4 NIM and the CSM management server

We use CSM to automate the node installation and management. This tool allows us to control the hardware and software on the client nodes, and also provides connection to the nodes. In this scenario, we use basic rsh-based host communication.

CSM utilizes NIM for node installation and automation, although NIM is part of AIX operating system. CSM provides us an interface to convert node definitions to the NIM resources. Remaining NIM tasks, like *lpp_source* and *SPOT* creation must be performed manually.

Preparing the environment

In order to install the CSM software, we perform some administration tasks:

- Update the /etc/hosts file with the names of the nodes and the HMC. The file might contain the following lines, for example:

```
192.168.100.72  p690_LPAR2.itso.ibm.com node2 p690_LPAR2
192.168.100.71  p690_LPAR1.itso.ibm.com node1 p690_LPAR1
192.168.100.69  hmcitso.itso.ibm.com p690_HMC.itso.ibm.com hmcitso
192.168.100.75  csmserver2.itso.ibm.com csmserver2
9.12.6.76       csmserver.itso.ibm.com csmserver
```

Tip: We decided to use long host names. We recommend not to mix long and short host naming convention because this may confuse the RSCT configuration and lead to problems.

- Update the $PATH and $MANPATH in the ~/.profile file:

We add the following entries to the ~/.profile file:

```
export PATH=$PATH:/opt/csm/bin
export MANPATH=$MANPATH:/opt/csm/man
```

- Prepare the /csminstall repository

  The installation manual recommend creation of the separate filesystem for the /csminstall repository, this is usually required when configuring the HACSM solution, and it must be created on a separate shared storage. We decided to implement a basic CSM configuration, and a separate /csminstall is not needed.

  We create the /csminstall directory using the **mkdir** command.

CSM cluster software installation

You must install the CSM software on the management server. In this scenario, we use CSM 1.4.0.1 release. We install the following software:

- csm.client
- csm.core
- csm.diagnostics
- csm.dsh
- csm.gui.dcem
- csm.gui.websm
- csm.msg.EN_US.core
- csm.msg.en_US.core
- csm.server

CSM requires various RSCT packages to be installed. In our environment, we use the following filesets:

- rsct.basic.rte
- rsct.compat.basic.rte

- rsct.core.auditrm
- rsct.core.errm
- rsct.core.fsrm
- rsct.core.hostrm
- rsct.core.lprm
- rsct.core.rmc
- rsct.core.sec
- rsct.core.sensorrm
- rsct.core.sr
- rsct.core.utils

Also some third-party software must be installed:

- Java14.license
- Java14.sdk
- conserver-7.2.4-1
- expect-5.32-1
- tcl-8.3.3-1
- tk-8.3.3-1
- openCIMOM-0.8-1

After CSM has been installed, we accept the full license by issuing the following command:

```
csmserver:/export/lppsource/CSM14:> csmconfig -L \
/export/lppsource/CSM14/csmlum.full
The license agreement has been accepted.
```

And we check our CSM configuration:

```
csmserver:> csmconfig
    AddUnrecognizedNodes = 0 (no)
    ClusterSNum =
    ClusterTM = 9078-160
    DeviceStatusFrequency = 12
    DeviceStatusSensitivity = 8
    ExpDate =
    HAMode = 0
    HeartbeatFrequency = 12
    HeartbeatSensitivity = 8
    LicenseProductVersion = 1.4
    MaxNumNodesInDomain = -1 (unlimited)
    PowerStatusMode = 0 (Mixed)
    RegSyncDelay = 1
    RemoteCopyCmd = /usr/bin/rcp
    RemoteShell = /usr/bin/rsh
    SetupKRB5 = 0
    SetupRemoteShell = 1 (yes)
```

The next step is to create the /csminstall file structure:

```
csmserver:> csmconfig -c
   About to copy CSM  command binaries.
```

We store the user ID and password for our HMC on the management server using the **systemid** command.

The **systemid** command stores the user ID and password required for internal programs to access remote hardware. You must run the command for each hardware control point in the cluster except for those associated with SP nodes or p660 servers. In other words you do not have to run the **systemid** command for any hardware control points that are to be used with a power method of CSP. For example:

```
csmserver:> systemid hmcitso.itso.ibm.com hscroot
   Password:
   Verifying, please re-enter password:
   systemid: Entry created.
```

Once the ID and password are stored, we verify the CSM configuration by issuing the following command as shown in Example 6-1.

Example 6-1 CSM verification

```
probemgr -p ibm.csm.ms -l 0
   csmserver:/csminstall:> probemgr -p ibm.csm.ms -l 0
   Running probe /opt/diagnostics/probes/network-enabled.
   Probe network-enabled was run successfully.
   Running probe /opt/diagnostics/probes/network-hostname.
   Probe network-hostname returned the following information.
   network-hostname:trace:My hostname seems to be csmserver.
   Probe network-hostname was run successfully.
   Running probe /opt/diagnostics/probes/network-ifaces.
   Probe network-ifaces was run successfully.
   ...
   ...
   ibm.csm.ms:trace:Checking that directory /opt/csm exists.
   ibm.csm.ms:trace:Checking that directory /opt/csm/bin exists.
   ibm.csm.ms:trace:Checking that directory /opt/csm/csmbin exists.
   ibm.csm.ms:trace:Checking that directory /usr/sbin/rsct/bin exists.
   ibm.csm.ms:trace:Checking that directory /var/log/csm exists.
   ibm.csm.ms:trace:Checking for packages : csm.client*, csm.core*,
   csm.diagnostics*, csm.dsh*, csm.gui.dcem*, csm.gui.websm*, csm.msg.*_*,
   csm.server*.
   ibm.csm.ms:trace:Checking for packages : conserver-7.2.4-1, expect*,
   openCIMOM-0.8-1, tcl*, tk*.
   ibm.csm.ms:trace:Check if the CFM cronjob is enabled.
Probe ibm.csm.ms was run successfully.
```

Defining nodes to the cluster

We use a node stanza file to feed the nodes to the **definenode** command. First we create the /tmp/mymapfile. For example:

```
csmserver:> lshwinfo -s -v -o - -c 192.168.100.69 -p \
hmc  | grep LPAR > /tmp/mymapfile
csmserver:> cat /tmp/mymapfile
   p690_LPAR1::hmc::192.168.100.69::p690_LPAR1::002::7040::681::022BE2A
   p690_LPAR2::hmc::192.168.100.69::p690_LPAR2::001::7040::681::022BE2A
```

In most cases, you must edit the file and add the host name of your nodes manually. Next we provide to the **definenode** command the previously created /tmp/mymapfile file as follows:

```
definenode -M /tmp/mymapfile ManagementServer=csmserver InstallOSName=AIX
   Defining CSM Nodes:
   Defining Node "p690_LPAR1.itso.ibm.com"("192.168.100.71")
   Defining Node "p690_LPAR2.itso.ibm.com"("192.168.100.72")
```

Issuing the **lsnode** command, we learn that our nodes are not managed by our CSM cluster yet:

```
csmserver:/csminstall:> lsnode -l | grep InstallStatus
    InstallStatus = PreManaged
    InstallStatus = PreManaged
```

Next, we create the node group:

```
nodegrp -a p690_LPAR1.itso.ibm.com,p690_LPAR2.itso.ibm.com p690_nodegroup
```

And, check if the communication between the management server and HMC works fine:

```
csmserver:/csminstall:> rpower -a query
   p690_LPAR1.itso.ibm.com off
   p690_LPAR2.itso.ibm.com off
```

Getting the network adapter information

To install a cluster node on a network, you must collect information about the Ethernet adapter that you plan to use to install and manage the node and to store that information in the CSM database. Each node must have at least one network adapter that can reach the management server for this purpose. You store the information for this adapter in the CSM database as node installation adapter attributes.

We use the following command to gather the boot interfaces information from the nodes as shown in example Example 6-2 on page 105.

Example 6-2 Gather the boot interfaces information.

```
csmserver:/:> getadapters -a  -D -t ent -s 100 -d full\
   -z /tmp/p690_stanzafile
   Can not use dsh - No nodes in Managed or MinManaged mode.
   Acquiring adapter information from Open Firmware for node
   p690_LPAR1.itso.ibm.com.
   Acquiring adapter information from Open Firmware for node
   p690_LPAR2.itso.ibm.com.

   # Name::Adapter Type::Adapter Name::MAC Address::Location Code::Adapter
   Speed::Adapter Duplex::Install Server::Adapter Gateway

   p690_LPAR1.itso.ibm.com::ent::::0002556A5352::U1.9-P1-I3/E1::100::full::csm
   server.itso.ibm.com::0.0.0.0::ok
   p690_LPAR2.itso.ibm.com::ent::::0002556A8FE9::U1.5-P1-I3/E1::100::full::csm
   server.itso.ibm.com::0.0.0.0::ok
   #---Stanza Summary----------------------
   #   Date: Thu Sep 16 17:01:09 CDT 2004
   #   Stanzas Added: 2
   #   Stanzas Updated: 0
#---End Of Summary----------------------
```

You can track the command progress in the log files located in /var/log/csm/getadapters directory.

Important: The **getadapters** command will power off the nodes if their mode is other than Managed or MinManaged.

Our network adapter stanza file is shown in Example 6-3.

Example 6-3 /tmp/p690_stanzafile

```
###CSM_ADAPTERS_STANZA_FILE###--do not remove this line
#---Stanza Summary----------------------
#   Date: Thu Sep 16 17:01:09 CDT 2004
#   Stanzas Added: 2
#   Stanzas Updated: 0
#---End Of Summary----------------------
p690_LPAR1.itso.ibm.com:
     MAC_address=0002556A5352
     adapter_duplex=full
     adapter_speed=100
     cable_type=N/A
     install_gateway=0.0.0.0
     install_server=csmserver.itso.ibm.com
     location=U1.9-P1-I3/E1
     machine_type=install
```

```
     netaddr=
     network_type=en
     ping_status=ok
     subnet_mask=

p690_LPAR2.itso.ibm.com:
     MAC_address=0002556A8FE9
     adapter_duplex=full
     adapter_speed=100
     cable_type=N/A
     install_gateway=0.0.0.0
     install_server=csmserver.itso.ibm.com
     location=U1.5-P1-I3/E1
     machine_type=install
     netaddr=
     network_type=en
     ping_status=ok
     subnet_mask=
```

Now, we feed the CSM database with the stanzafile:

```
csmserver:/:> getadapters -w -f /tmp/p690_stanzafile
   # Name::Adapter Type::Adapter Name::MAC Address::Location Code::Adapter
   Speed::Adapter Duplex::Install Server::Adapter Gateway

   p690_LPAR1.itso.ibm.com::en::::0002556A5352::::100::full::csmserver.itso.ib
   m.com::0.0.0.0::
   p690_LPAR2.itso.ibm.com::en::::0002556A8FE9::::100::full::csmserver.itso.ib
   m.com::0.0.0.0::
```

NIM lpps preparation

NIM enables a cluster administrator to centrally manage the installation and configuration of AIX and optional software on machines within a network environment. Setting up NIM involves various tasks including the following:

- Installing NIM filesets
- Configuring basic resources
- Creating machine and network definitions
- Creating resources that are used to install the nodes

The specific tasks that you need to perform depend on which features of NIM you plan to use. For more information about NIM and nodes installation refer to *AIX Installation in a Partitioned Environment,* SC23-4382-04.

NIM master initialization

First, we initialize the NIM master network. This creates the /etc/niminfo file and start NIM daemons.

We use the *nimconfig* SMIT panel to enter the appropriate information as shown in example Example 6-4.

Example 6-4 smitty nimconfig menu

```
Configure Network Installation Management Master Fileset

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

                                                      [Entry Fields]
* Network Name                                       [9_12_net]
* Primary Network Install Interface                  [en1]                    +

  Allow Machines to Register Themselves as Clients?  [no]                    +
  Alternate Port Numbers for Network Communications
       (reserved values will be used if left blank)
    Client Registration                               []                       #
    Client Communications                             []                       #

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

Note: We configured the 9.12.6.76 network because it is required for Chapter 7, “POWER5 provisioning scenario” on page 135.

To simplify the NIM setup process, we use the **nim_master_setup** command. This command automatically performs basic NIM tasks. It installs NIM filesets, configures NIM, creates basic resources and creates a resource group with the resources that are created. Example 6-5 shows the output from the **nim_master_setup** script command.

Example 6-5 output from the nim_master_setup command.

```
csmserver:/:> nim_master_setup -B
############################## NIM master setup ##############################
#                                                                            #
#  During script execution, lpp_source and spot resource creation times      #
#  may vary. To view the install log at any time during nim_master_setup,    #
#  run the command: tail -f /var/adm/ras/nim.setup in a separate screen.     #
#                                                                            #
##############################################################################

Device location is /dev/cd0
Resources will be defined on volume group rootvg.
Resources will exist in filesystem /export/nim.
Checking for backup software...already installed.
```

```
Checking /tmp space requirement...done
Installing NIM master fileset...already installed.
Located volume group rootvg.
Creating resolv_conf resource master_net_conf...done
Creating bosinst_data resource bid_ow...done

Please insert AIX 5.2 product media in device /dev/cd0
If the location for AIX 5.2 product media differs from
device /dev/cd0, supply the absolute path BEFORE pressing the ENTER key.
=> /dev/cd0
Checking /export/nim space requirement...done
Creating lpp_source resource 520lpp_res...done
Checking /export/nim space requirement...done
Checking /tftpboot space requirement...done
Creating spot resource 520spot_res...done
Creating resource group basic_res_grp...done
The following resources now exist:
boot                resources       boot
nim_script          resources       nim_script
master_net_conf     resources       resolv_conf
bid_ow              resources       bosinst_data
520lpp_res          resources       lpp_source
520spot_res         resources       spot
NIM master setup is complete - enjoy!
```

This script created the following NIM resources:

```
csmserver:/:> lsnim
   master              machines        master
   boot                resources       boot
   nim_script          resources       nim_script
   nim_network         networks        ent
   master_net_conf     resources       resolv_conf
   bid_ow              resources       bosinst_data
   520lpp_res          resources       lpp_source
   520spot_res         resources       spot
   basic_res_grp       groups          res_group
```

We update the *520lpp_res* resource with the latest Maintenance Level using the following command:

```
csmserver:/:> nim_update_all -d /export/lppsource/AIX52_august_update/ -l
520lpp_res -s 520spot_res -B

############################## NIM update all ##############################
#                                                                          #
#  During script execution, NIM client and resource updating times         #
#  may vary. To view the install log at any time during nim_update_all,    #
#  run the command: tail -f /var/adm/ras/nim.update in a separate screen.  #
```

```
#                                                                              #
################################################################################

NSORDER=local,bind
Adding updates to 52Olpp_res lpp_source...
done
Updating 52Ospot_res using updated lpp_source 52Olpp_res...done

Generating list of client objects in NIM environment...
Unable to obtain NIM client objects list - Exiting.
```

We create another NIM network for the p690 nodes as shown in Example 6-6.

Example 6-6 Create NIM network for p690 nodes

```
Define a Network

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

                                                      [Entry Fields]
* Network Name                                       [192_168_net]
* Network Type                                        ent
* Ethernet Type                                     Standard
+
* Network IP Address                                 [192.168.100.0]
* Subnetmask                                         [255.255.255.0]
  Default Gateway for this Network                   [192.168.100.126]
  Other Network Type
+
  Comments                                           []

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

We create routes between the two NIM networks defined as shown in Example 6-7.

Example 6-7 Create routes between NIM networks

```
Define a Static Network Install Route

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

                                                      [Entry Fields]
* Originating Network Name                           192_168_net
* Gateway Used by Originating Network                [192.168.100.75]
```

```
* Destination Network Name                          9_12_net
* Gateway Used by Destination Network              [9.12.6.76]
  Force                                            no
+

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

We have two NIM networks and the routes created between them are shown in the following example:

```
csmserver:/tmp:> lsnim -l 9_12_net
   9_12_net:
      class      = networks
      type       = ent
      Nstate     = ready for use
      prev_state = ready for use
      net_addr   = 9.12.6.0
      snm        = 255.255.254.0
      routing1   = default 192.168.100.60
      routing2   = 192_168_net 9.12.6.76
csmserver:/tmp:> lsnim -l 192_168_net
   192_168_net:
      class      = networks
      type       = ent
      Nstate     = ready for use
      prev_state = ready for use
      net_addr   = 192.168.100.0
      snm        = 255.255.255.0
      routing1   = default 192.168.100.126
      routing2   = 9_12_net 192.168.100.75
```

We set the *ipforwarding* and *ip6forwarding* network tunables to allow routing of the *tftp* and *nfs* communication through the management server as shown in the following example:

```
   csmserver:/:> no -po ipforwarding=1
      Setting ipforwarding to 1
      Setting ipforwarding to 1 in nextboot file
   csmserver:/:> no -po ip6forwarding=1
      Setting ip6forwarding to 1
      Setting ip6forwarding to 1 in nextboot file
```

CSM provides us with a tool that converts cluster node definition into NIM resources. We use the *csm_nimnodes* SMIT panel to input the appropriate information as shown in Example 6-8 on page 111. This panel invokes the *csm2nimnodes* script.

Example 6-8 csm_nimnodes SMIT panel

```
Convert CSM to NIM Nodes

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

                                                      [Entry Fields]
  Setup all nodes                                      no                      +
   OR
  List of nodes                                       [p690_LPAR1.itso.ibm.c> +
   OR
  List of groups                                      []                      +

  Update existing NIM group definitions                yes                     +
  NIM machine type                                     standalone              +
  NIM network name                                    [192_168_net]            +
  Ring speed (required if token ring)                 []                       +
  Cable type (required if ethernet)                   [tp]                     +
  Platform                                             chrp                    +
  Netboot kernel                                       mp                      +
  Display Verbose Messages?                            no                      +

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

The *csm2nimnodes* creates the following objects in NIM:

```
   p690_LPAR1:
      class          = machines
      type           = standalone
      connect        = shell
      platform       = chrp
      netboot_kernel = mp
      if1            = 192_168_net p690_LPAR1 0002556A5352 ent
      cable_type1    = tp
      Cstate         = ready for a NIM operation
      prev_state     = ready for a NIM operation
      Mstate         = currently running
   csmserver:/tmp:> lsnim -l p690_LPAR2
   p690_LPAR2:
      class          = machines
      type           = standalone
      connect        = shell
      platform       = chrp
      netboot_kernel = mp
      if1            = 192_168_net p690_LPAR2 0002556A8FE9 ent
      cable_type1    = tp
```

```
Cstate         = ready for a NIM operation
prev_state     = ready for a NIM operation
Mstate         = currently running
```

Define openssh and openssl installp bundles

In order to provide a secure environment, we install the OpenSSH bundle. We use NIM to perform the automatic installation of the openssh during the installation of the operating system. To perform this task we create two bundle files, openssh.bnd and openssl.bnd in the /export/nim directory, as specified in Example 6-9.

Example 6-9 openssh and openssl bundles

```
csmserver:/export/nim:> cat openssh.bnd
I:openssh.base
I:openssh.license
I:openssh.man.en_US
I:openssh.msg.en_US
I:openssh.msg.EN_US
csmserver:/export/nim:> cat openssl.bnd
R:openssl*
```

Then we create the openssh bundle resources in NIM:

```
nim -o define -t installp_bundle -a server=master \
-a location=/export/nim/openssl.bnd openssl
nim -o define -t installp_bundle -a server=master \
-a location=/export/nim/openssh.bnd openssh
```

Because the openssh and openssl install bundles are located on different CDs, we have to transfer them to the NIM lppsources. We use the *smitty nim_res_op -> 520lpp_res -> update -> add* and specify the installation media and the bundle to perform this task. We also have to update the *SPOT* from the changed *lppsource*.

Name service customization

The resolv.conf is a file which is part of a default NIM resource allocated to each new node. The NIM resource name is *master_net_conf*. If for any reason the new node cannot resolve the CSM server using the DNS server specified in the resolv.conf file the CSM customization of the node will fail.

To prevent this we create a new NIM *fb_script* resource called *set_netsvc_script*. The resource contains a script which is started before the first reboot after the AIX has been installed by NIM. For example:

```
csmserver:/:> lsnim -l set_netsvc_script
   set_netsvc_script:
```

```
      class       = resources
      type        = fb_script
      Rstate      = ready for use
      prev_state  = unavailable for use
      location    = /export/nim/set_netsvc
      alloc_count = 0
      server      = master
csmserver:/:> cat /export/nim/set_netsvc
   echo "hosts=local,bind" >> /etc/netsvc.conf
```

With this setting, the installed node uses the local /etc/hosts file as a first source for host name resolution.

Define the CSM customization script into NIM

We create the CSM customization NIM script resource:

```
csmsetupnim -a
```

This allocates the script resource to all NIM clients as shown in Example 6-10.

Example 6-10 NIM resource allocation

```
csmserver:/tmp:> lsnim -l p690_LPAR1
   p690_LPAR1:
      class          = machines
      type           = standalone
      connect        = shell
      platform       = chrp
      netboot_kernel = mp
      if1            = 192_168_net p690_LPAR1 0002556A5352 ent
      cable_type1    = tp
      Cstate         = ready for a NIM operation
      prev_state     = BOS installation has been enabled
      Mstate         = currently running
      script         = csmprereboot_script
      cpuid          = 0022BE2A4C00
      control        = master
      Cstate_result  = reset
```

6.1.5 Automatic node customization and application deployment

In order to automate the client installation, we create a script that is executed once the client has been installed and rebooted. This script can perform various tasks to suit our needs.

We use the *Apache* software bundle as the sample application. Although *Apache* can be automatically installed as NIM software bundle, we want to utilize the

CSM *installpostreboot* script, because it is more flexible and can perform more customization on a the fresh installed node. For example, it can install and configure an Oracle database.

We create a sample *node_customize* script for the client and put it into the /csminstall/csm/scripts/installpostreboot directory. Scripts in this directory will run after the first reboot of the node.

Also we copied the *.profile* and *Apache* filesets to the /bff directory. This will be used for our application installation.

The CSM commands will check each of these directories and run each executable script at the appropriate time. Subdirectories will not be checked. If there are multiple scripts in a directory, they will run in alphabetical order as determined by the **ls** command on the management server. The naming convention for these files is scriptname[._target]. The ._ characters following the script name are required if the script is only used for a specific node or node group. The target value must be a single node name or group name that has been defined in the CSM database. If the target extension is not used then the script will run on all nodes. If there is a script and additional multiple versions for subsets of nodes, such as myscript, myscript._groupA, myscript._groupB, then the script with no target extension will be run only for those nodes that are not included in one of the specific groups; for example, not in groupA or groupB.

Note: When using customization scripts during node installation, you must still follow the node installation procedures described in the IBM CSM for AIX 5L: Planning and Installation Guide, SA22-7919-07. You must run the **csmsetupnim** command before each node installation, but you do not have to run the command each time you add or delete a customization script.

We export the /bff directory with apache install bundles from the management server as follows:

```
csmserver:> mknfsexp -B -d /bff
```

We prepare a script to show the flexibility of this approach. This script mounts via NFS from the previously exported /bff filesystem on the management server and performs the following tasks:

- Create root's .profile on the installed node
- Create the /etc/hosts file
- Installs the *Apache* application
- Adds the automatic startup of the application to the /etc/inittab file
- Performs *rootvg* volume group mirroring

The script is very simple. It does not check return codes from various operations and availability of resources. But its primary idea is to provide to the reader the basic idea of CSM's post install customization capabilities. You can use this script to automate your nodes' installation. Example 6-11 shows our sample customization script.

Example 6-11 node_customize script

```
# Some definitions
export NIMMASTER=192.168.100.75
mount $NIMMASTER:/bff /mnt
# On the host server we install the apache application
/usr/lib/instl/sm_inst installp_cmd -a -l -d '/mnt' -f 'ALL' '-c' '-N' '-g'
'-X' '-Y'

# We copy the .profile...
cp /mnt/.profile /.profile

# copy the /etc/hosts
cp /mnt/hosts /etc/hosts

# add the automatic application startup
echo "httpd:2:once:/usr/sbin/httpd start" >> /etc/inittab

# perform the rootvg mirroring and reboot the node
/usr/sbin/extendvg -f rootvg hdisk1
/usr/sbin/mirrorvg -m rootvg hdisk1
chvg -Qn rootvg
/usr/sbin/bosboot -a -d /dev/ipldevice
/usr/bin/bootlist -m normal hdisk0 hdisk1
shutdown -Fr
```

6.2 The client installation

In this section, we cover the installation of the client nodes.

Note: This is the part that IBM Tivoli Provisioning Manager provides if it is implemented in your environment.

6.2.1 Set the nodes to install

Once we have CSM and NIM configured, we can perform installation of the operating system on the client nodes.

First, we allocate the NIM resources to the nodes:

```
csmserver:> nim -o bos_inst -a source=rte -a boot_client=no \
-a group=basic_res_grp -a fb_script=set_netsvc_script \
-a installp_bundle=openssl -a installp_bundle=openssh \
   p690_LPAR1 p690_LPAR2<
   +-------------------------------------------------------------------------------+
                        Enabling "bos_inst" Operation
   +-------------------------------------------------------------------------------+
    Allocating resources ...

   +-------------------------------------------------------------------------------+
                        "bos_inst" Operation Summary
   +-------------------------------------------------------------------------------+
    Target                  Result
    ------                  ------
    p690_LPAR1              ENABLED
    p690_LPAR2              ENABLED


   Note: ENABLED targets must be rebooted to initiate the operation.
```

6.2.2 Installp bundle prerequisite handling

In the previous section, we used a specific order for the installp bundle allocation. First *openssl* and then *openssh* packages, because *openssl* is a prerequisite for *openssh*.

Note: Before AIX V5.2 maintenance level 4 all the allocated AIX filesets were installed before the allocated RPM packages. For *openssh* installation, this caused a prerequisite failure.

To check the order before the client installation is started use the following command:

```
csmserver:/tmp:> lsnim -c resources p690_LPAR2
   set_netsvc_script     fb_script
   openssl               installp_bundle
   openssh               installp_bundle
   bid_ow                bosinst_data
   520lpp_res            lpp_source
   master_net_conf       resolv_conf
   520spot_res           spot
   nim_script            nim_script
   boot                  boot
```

Note: The **lsnim -l "hostname"** command gives back different order for the allocated NIM resources, than the order they will be installed.

6.2.3 Routing issues

In our scenario, we have a private network for the nodes, which is not routable. The management server is set to use the IP label assigned to the 9.12.6.76 address. Because of this, we have to create two NIM networks and a route between them as shown in Example 6-6 on page 109, and Example 6-7 on page 109. The *ipforwarding* and the *ip6forwarding* for the management server is already set in "NIM master initialization" on page 106.

When the *bos_inst* operation is initiated on the management server an info file is created in the /tftboot directory for each nodes. Additionally the /etc/bootptab file is created and contains the following:

```
p690_LPAR1.itso.ibm.com:bf=/tftpboot/p690_LPAR1.itso.ibm.com:ip=192.168.100
.71:ht=ethernet:ha=0002556A5352:sa=9.12.6.76:gw=192.168.100.126:sm=255.255.
255.0:
p690_LPAR2.itso.ibm.com:bf=/tftpboot/p690_LPAR2.itso.ibm.com:ip=192.168.100
.72:ht=ethernet:ha=0002556A8FE9:sa=9.12.6.76:gw=192.168.100.126:sm=255.255.
255.0:
```

The routing information is set to use the default route. This is not good for our scenario, as the management server (9.12.6.76) is not reachable via the default route (192.168.100.126). We have two possibilities to avoid this problem:

- Change the gateway in the bootptab file to 192.168.100.75. After this the bootpd daemon has to be stopped and restarted to have the new settings.
- Remove the default route information of the client network. A new NIM script could be defined and allocated to the node, which will set the default route on the node after installation.

Note: We configured the 9.12.6.76 network because it is required for Chapter 7, "POWER5 provisioning scenario" on page 135.

6.2.4 Network boot the nodes

And finally, we initiate the network boot and installation of the nodes by issuing the following command:

```
csmserver:> netboot -N p690_nodegroup
```

The installation progress can be monitored by using the **rconsole** command as shown in Example 6-12 on page 118.

Example 6-12 output from the rconsole command.

```
smserver:/:> rconsole -r -t -N p690_nodegroup
[Enter `^Ec?' for help]


          1 = SMS Menu                          5 = Default Boot List
          6 = Stored Boot List                  8 = Open Firmware Prompt


     memory      keyboard     network     scsi     speaker  ok
0 >
```

6.3 Dynamic LPAR operations

Once the client nodes have been installed, we take advantage of the provisioning capabilities of dynamic LPARs provided by POWER4 systems. For more information about dynamic LPAR, refer to 4.1, “Hardware provisioning tools” on page 30.

6.3.1 Dynamic LPAR using the IBM Web-based System Manager GUI

We use the IBM Web-based System Manager GUI to dynamically migrate one CPU from partition p690_LPAR1 to partition p690_LPAR2.

Step 1. Verify the nodes’ configuration

We use the **`lscfg`** command to verify the number of CPUs in the partitions as shown in the following table:

p690_LPAR1	+ proc1 + proc3	U1.18-P1-C1 Processor U1.18-P1-C1 Processor
p690_LPAR2	+ proc7 + proc19	U1.18-P1-C1 Processor U1.18-P1-C4 Processor

Step 2. Perform the CPU migration

Use the *Dynamic Logical Partition -> Processors* IBM Web-based System Manager menu to migrate one CPU between LPARs as shown in Figure 6-6 on page 119.

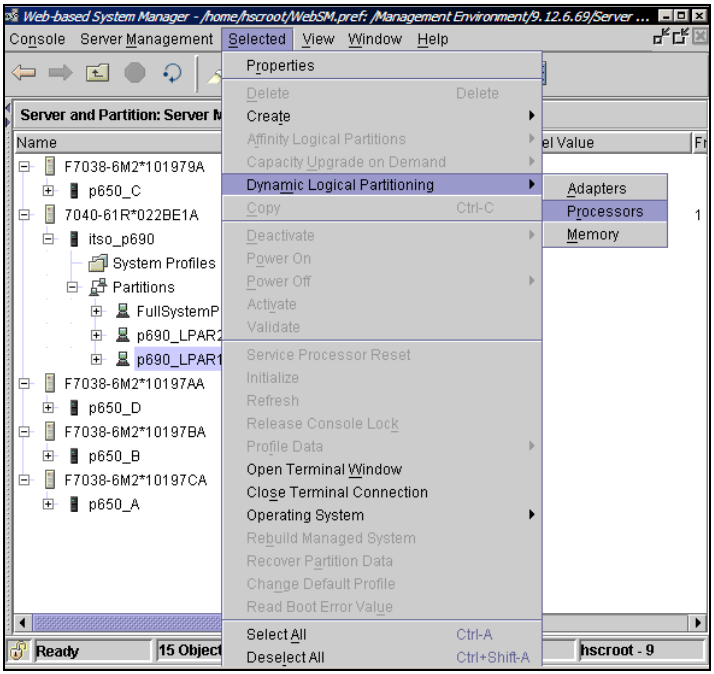


Figure 6-6 GUI for the dynamic LPAR menu

Tip: If your environment meets all dynamic LPAR requirements, your nodes are up and you do not have the dynamic LPAR menu active, you should check if clients hostnames are resolvable by HMC. If this is the issue, add the clients' hostnames to the DNS or HMC's /etc/hosts file and refresh the RSCT on HMC and client nodes:

1. Clean out keys database:

```
lspartition -dlpar
/usr/sbin/rsct/bin/ctsthl -l
/usr/sbin/rsct/bin/ctsthl -d <name/IP>
```

2. Restart RSCT daemons:

```
/usr/sbin/rsct/bin/rmcctrl -z (stop subsystem)
/usr/sbin/rsct/bin/rmcctrl -A (add and start subsystem)
/usr/sbin/rsct/bin/rmcctrl -p (enable remote clients)
```

For more information about fixing the dynamic LPAR problems, refer to: 4.1, "Hardware provisioning tools" on page 30.

Move the CPU from one p690_LPAR1 to p690_LPAR2 as shown in Figure 6-7.

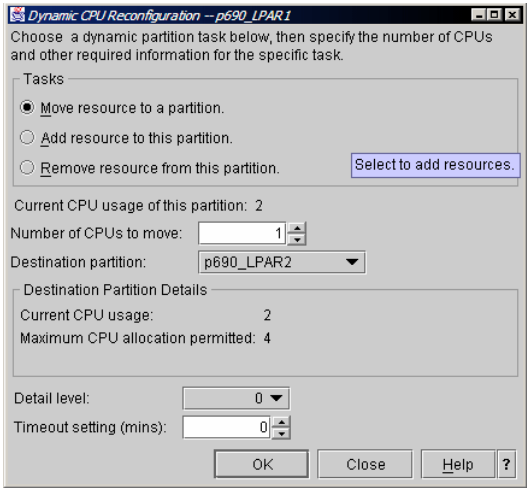


Figure 6-7 Move CPU between LPARs

Step 3. Verify the operation

If the operation succeeded, we verify the nodes' configuration using the **`lscfg`** command:

p690_LPAR1	+ proc3	U1.18-P1-C1 Processor
p690_LPAR2	+ proc1 + proc7 + proc19	U1.18-P1-C1 Processor U1.18-P1-C1 Processor U1.18-P1-C4 Processor

6.3.2 Automated dynamic LPAR

In our system environment, we want our resources to be provisioned automatically. IBM provides a set of tools to create fully automated system, where resources, like CPUs, memory and IO slots migrate between LPARs without downtime and according to the system load. In a POWER4 environment, this goal could be achieved by using the IBM dynamic LPAR Tool Set for pSeries from Alphaworks (http://www.alphaworks.ibm.com/tech/dlpar). In a POWER5 system, this is achieved by using Partition Load Manager. Refer to 7.5, "Partition Load Manager (PLM)" on page 151.

In our scenario, we perform the following steps:

- Install the dynamic LPAR Tool Set on our management server and client nodes

- Configure the SSH connection between the management server and the HMC
- Customize dynamic LPAR Toolset configuration files on the management server
- Start the resource monitor script on the management server
- Run the **cpu-stress** command
- Watch the automatic resource migration in LPARs

The dynamic LPAR Tool Set can perform many different tasks, for example, time - based automatic resource reallocation with performing the resource consistency check, but we do not cover this in the scenario.

Step 1. Installing the dynamic LPAR Tool Set

We untar the dynamic LPAR Tool Set on our management station to the /opt/dlparToolset directory. Our management server will run the resource monitor script. We also install the dynamic LPAR Tool Set on our client nodes, but we use only system stress scripts on them.

Step 2. Configure the SSH connection

We setup the SSH connection to allow the management server to issue operations on HMC without prompting for the password. This is required for the resource monitor scripts to be able to relocate resources between nodes. We add the CSM server root's public ssh key to the hscroot's ~/.ssh/authorized_keys2 file as shown in Example 6-13.

Example 6-13 hscroot's authorized_keys2 file

```
[hscroot@hmcitso .ssh]$ cat authorized_keys2
ssh-rsa
AAAAB3NzaC1yc2EAAAABIwAAAIEA1hnVBhzJdunwxN1TA+2hSYTO7qOToOJvYZ29NrOCSplU7MYiGH2
hahS3QA/ts6oWq8CA23qVh44ZJFYwZ8bOTnJ+n+NjZuvYS18odcMTW8yOzPc9PnaweloF+W3pJ3w+/O
NrlHmAtKGv8ubL3RPv9PM8Uq8Am/OVU1NrAeQXFQE= root@csmserver
```

Step 3. Customize the dynamic LPAR configuration files

We edit the *setEnv* script to enter our site data. Our script is shown in Example 6-14.

Example 6-14 setEnv script

```
export DR_TS_DIR=/opt/dlparToolset/bin
 export DR_HMC_HOST=hmcitso.itso.ibm.com
 export DR_HMC_USER=hscroot
 export DR_MANAGED_SYS=itso_p690
 export DR_HOST_LIST=$DR_TS_DIR/hostList  # Used in directories 1,2,5
```

```
# For lparLsCfgs.pl:
 export DR_OUTFILE_LSCFG=~/lparLsCfgs.outFile
# For lparLsCfgsAll.pl:
 export DR_OUTFILE_LSCFG_ALL=~/lparLsCfgsAll.outFile

# For lparLsCfgs.pl:
 export DR_INTVLSECS_LSLOAD=2 # Seconds.
 export DR_HISTCNT_LSLOAD=5   # Number of historical snapshots.
 export DR_HISTOUT_LSLOAD=~/lparLsLoads.histOut

# For lparLoadRM.pl:
 export DR_HOSTS_WTS=$DR_TS_DIR/hostsWts # Defines an LPAR group.
 export DR_INTVLSECS_LOADRM=4   # Seconds.
 export DR_HISTCNT_LOADRM=5     # Number of historical snapshots.

 export DR_TRGR_CPU_LOAD=98   # Integer Percentage of non-idle CPU time.
 export DR_TRGR_RQ_DEPTH=1    # Integer Number of threads/CPU/sec waiting
                              #         for CPU service.

 export DR_TRGR_MEM_LOAD=85   # Integer Percentage of non-free real memory.
 export DR_TRGR_PGSTEALS=0    # Integer Number of page-steals/LMB/sec.

# For moveSlot.pl:
  export DR_IO_SLOT=U1.9-P1-I10 # An IO slot; typically the IO-media slot.

# Other environment variables that may be rarely changed:
 export DR_RSVD_FREP_CPUS=4
 export DR_RSVD_FREP_LMBS=0
 export DR_DEBUG_LVL=0

# To display the sorted values on the screen:
 env | grep DR_ | sort
```

You can specify the host name of your HMC, location of output files, and threshold values for dynamic resource reallocation.

Important: If you make any changes to this script, remember to reload the new environmental values using the following command:

```
# . /opt/dlparToolSet/bin/setEnv
```

Another important files are the *hostsWts* and the *hostList* as shown in Example 6-15 on page 123, where you specify the hostnames of your client nodes, their weight (priority for accessing the resources), and override the default threshold values set in the *setEnv* script.

Example 6-15 hostsWts and hostList files

```
csmserver:/opt/dlparToolset/bin:> cat hostsWts
#hostName WEIGHT    CPU_LOAD   MEM_LOAD
p690_LPAR1     20        95       85
p690_LPAR2     80        95       85
csmserver:/opt/dlparToolset/bin:> cat hostList
p690_LPAR1
p690_LPAR2
```

Step 4. Start the resource monitor script on MS

We start the resource monitor script (**`lparLoadRM.pl`**) on the management server as shown in Example 6-16.

Example 6-16 Starting the resource monitor script on the management server

```
This script is being run by user "root" on host "csmserver".
Checking ping and ssh to hmcitso.itso.ibm.com.....
HMC Commands are running under "Restricted Bash" shell.....
INPUT:
   <hostsWts> file                 : /opt/dlparToolset/bin/hostsWts
   Refresh interval                : 4 seconds
   Show history data for the past  : 5 refreshes
   HMC Hostname                    : hmcitso.itso.ibm.com
   HMC Username (for ssh)          : hscroot
   Managed System Name             : itso_p690
   Reserved #CPUs in the Free Pool : 4
   Reserved #LMBs in the Free Pool : 0
Checking ping and rsh to p690_LPAR1.....
Checking ping and rsh to p690_LPAR2.....

hostName  WEIGHT  CPU_LOAD  RQ_DEPTH  MEM_LOAD  PGSTEALS
p690_LPAR1      20        95        1        85        0

p690_LPAR2      80        95        1        85        0

Retrieving the current configuration from HMC...
NOTE: It takes roughly 30 to 45 seconds to get the current
      configuration of each LPAR from the HMC.
....
--------------------------------------------------------------------------
CURRENT CONFIGURATIONs:
       Hostname       LPARname CPUmin CPUcur CPUmax LMBmin LMBcur LMBmax
       --------       -------- ------ ------ ------ ------ ------ ------
     p690_LPAR1     p690_LPAR1      1      2      4      4     16     32
     p690_LPAR2     p690_LPAR2      1      2      4      4     16     32
   Available (=unreserved) #CPUs in the Free Pool = 0
   Available (=unreserved) #LMBs in the Free Pool = 29
```

```
DESIRED RESOURCE CHANGES based on (min,max) values and historical Loads
       of CPU & memory (Negative numbers indicate unused resources):
       Hostname   Change in #CPUs   Change in #LMBs    Weight
       --------   ---------------   ---------------    ------
     p690_LPAR1                -1               -12      20
     p690_LPAR2                -1               -12      80

PLANNED RESOURCE CHANGES based on FREE POOL, WEIGHTs and historical
     Loads for CPU and Memory (Negative numbers indicate unused resources):
       Hostname   Change in #CPUs   Change in #LMBs    Weight
       --------   ---------------   ---------------    ------
     p690_LPAR1                 0                 0      20
     p690_LPAR2                 0                 0      80
```

As you can see, there are no CPUs available in the Free Pool (not allocated to any LPAR), although four CPUs are used by the p690_LPAR1 and p690_LPAR2. The remaining four CPUs have been disabled in the *setEnv* script by the following entry:

```
export DR_RSVD_FREP_CPUS=4
```

This is because we want the LPARs to share the CPUs according to their weight settings.

The nodes are idle, we can check this by running the *lparLsLoads.pl* script as follows:

```
CURRENT LOADs (CPU, MEMORY) -- snapshot taken every 2 seconds.

_____________ _____ _____ _____ _____________ _____ _____ _____ _____ _____________
|    Hostname| |#P|#RTh/P| %   CPU Load | |#LMB|   #MB| #FR/s/L| % Memory Load|
|------------| |--|------|---|----------| |----|------|--------|---|----------|
|  p690_LPAR1| | 2| 0.000|  8|#         | |  16|  4096|   0.000|  6|#         |
|  p690_LPAR2| | 2| 0.000| 10|#         | |  16|  4096|   0.000|  6|#         |
|____________| |__|______|___|__________| |____|______|________|___|__________|
HISTORICAL LOADs -- averaged over the last 5 snapshots.

_____________ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____ _____
|    Hostname||avg #RTh/P|% cpuLoad Hist|freP| |freL|Avg#FR/s/L|% memLoad Hist|
|------------||----------|---|----------|----| |----|----------|---|----------|
|  p690_LPAR1||     0.000|  8|#         |   1| |  14|     0.000|  6|#         |
|  p690_LPAR2||     0.100| 12|#         |   1| |  14|     0.000|  6|#         |
|____________||__________|___|__________|____| |____|__________|___|__________|
For definitions of the headings, see the DEFs file in the DOC/ subdirectory.
Press Control-C to exit!
```

Step 5. Run the cpu-stress script on the client node

We check the CPU resources on the nodes:

```
P690_lpar1:
+ proc3            U1.18-P1-C1 Processor
+ proc7            U1.18-P1-C1 Processor
p690_LPAR2:
+ proc5            U1.18-P1-C1 Processor
+ proc19           U1.18-P1-C4 Processor
```

Then we run the cpu-stress script on the p690_LPAR2 node:

```
# ./cpuStress.ksh 240
```

This generates set of cpu-intensive processes to rise the system load. We verify the load by looking at the *lparLsLoads.pl* script's output:

```
CURRENT LOADs (CPU, MEMORY) -- snapshot taken every 2 seconds.

_____________ _____________________ _____________________________________
|    Hostname| |#P|#RTh/P| %   CPU Load | |#LMB|   #MB| #FR/s/L| % Memory Load|
|------------| |--|------|---|----------| |----|------|--------|---|----------|
|  p690_LPAR1| | 2| 0.000|  8|#         | |  16|  4096|   0.000|  6|#         |
|  p690_LPAR2| | 2| 1.500|100|##########| |  16|  4096|   0.000|  6|#         |
|____________| |__|______|___|__________| |____|______|________|___|__________|
HISTORICAL LOADs -- averaged over the last 5 snapshots.

_____________ _____________________ _____ _____________________________
|    Hostname||avg #RTh/P|% cpuLoad Hist|freP| |freL|Avg#FR/s/L|% memLoad Hist|
|------------||----------|---|----------|----| |----|----------|---|----------|
|  p690_LPAR1||     0.000|  8|#         |   1| |  14|     0.000|  6|#         |
|  p690_LPAR2||     0.400| 25|###       |   1| |  14|     0.000|  6|#         |
|____________||__________|___|__________|____| |____|__________|___|__________|
For definitions of the headings, see the DEFs file in the DOC/ subdirectory.
Press Control-C to exit!
```

Also we keep eye on the output from **lparLoadRM.pl**:

```
--------------------------------------------------------------------------
CURRENT CONFIGURATIONs:
       Hostname       LPARname CPUmin CPUcur CPUmax LMBmin LMBcur LMBmax
       --------       -------- ------ ------ ------ ------ ------ ------
     p690_LPAR1     p690_LPAR1      1      2      4      4     16     32
     p690_LPAR2     p690_LPAR2      1      2      4      4     16     32
   Available (=unreserved) #CPUs in the Free Pool = 0
   Available (=unreserved) #LMBs in the Free Pool = 29

DESIRED RESOURCE CHANGES based on (min,max) values and historical Loads
       of CPU & memory (Negative numbers indicate unused resources):
```

```
       Hostname   Change in #CPUs   Change in #LMBs    Weight
       --------   ---------------   ---------------    ------
     p690_LPAR1                -1               -12      20
     p690_LPAR2                +1               -12      80

PLANNED RESOURCE CHANGES based on FREE POOL, WEIGHTs and historical
     Loads for CPU and Memory (Negative numbers indicate unused resources):
       Hostname   Change in #CPUs   Change in #LMBs    Weight
       --------   ---------------   ---------------    ------
     p690_LPAR1                -1                 0      20
     p690_LPAR2                +1                 0      80
/usr/bin/ssh hscroot@hmcitso.itso.ibm.com chhwres -r cpu  -o m -m itso_p690 -p
p690_LPAR1 -t p690_LPAR2 -q 1 -d 0
```

As we can see, the script notices the lack of processing power on p690_LPAR2 and transfers one CPU from p690_LPAR1 to p690_LPAR2. The resource monitor does not deallocate resources when the node becomes idle in order to avoid trashing.

Step 6. Verify the resources on the client nodes

We verify the change in the number of CPUs:

```
P690_LPAR1:
+ proc7            U1.18-P1-C1        Processor
p690_LPAR2:
+ proc3            U1.18-P1-C1    Processor
+ proc5            U1.18-P1-C1    Processor
+ proc19           U1.18-P1-C4    Processor
```

6.4 RSCT event manager

RSCT plays key role in provisioning for pSeries. We can use some of its functions to make the system more stable. For more information about RSCT and it's components refer to *IBM Reliable Scalable Cluster Technology Administration Guide,* SA22-7889-05.

In this sample scenario, we show some of its functions related to the system monitoring based on events generated by the operating system. These event are monitored by RMC, the Resource Monitor & Control RSCT subsystem. We define an RMC monitor to control the size of the /fs filesystem (We consider this filesystem should never be 100% full). When the size of the filesystem reaches 90%, RMC automatically runs the script that extends the filesystem by the size of one Logical Partition. Then we copy some random data, to fill the /fs filesystem and observe the effect. There are GUI panels for RMC management, but in this

scenario we use command line. You can then insert the commands into the first boot script (see Example 6-11 on page 115) and have them executed after the node has been installed.

The basic flow of the monitor creation is as follows:

- Create condition
- Create response
- Link response with condition
- Start the monitor

6.4.1 Prepare the monitor

In the following steps we prepare and start the /fs filesystem monitor.

Step 1. Define the RMC condition

We issue the following command to create the “/fs space used” condition:

```
mkcondition -r "IBM.FileSystem" -e "PercentTotUsed > 90" -d "/fs 90% full"\
   -s "Name == \"/fs\"" -S c "/fs space used"
```

This creates the condition that triggers the response when the size of the /fs filesystem reaches 90%.

Step 2. Define the RMC response

mkresponse command is used to create the response:

```
mkresponse -n "Extend /fs filesystem" -s /bff/rmcchfs "Extend /fs filesystem"
```

This runs the */bff/rmcchfs* script when this response is called. The sample */bff/rmchfs* script is presented in Example 6-17.

Example 6-17 /bff/rmcchfs script

```
#/usr/bin/ksh
# This script extends the /fs filesystem by 1 logical partition
chfs -a size=+1 /fs
if [ $? = 0 ]
then
        wall "extension of /fs success"
else wall "extension of /fs filesystem failed"
fi
```

Step 3. Link the condition with the response; start the monitor

We issue the following command to link the condition and the response:

```
   #mkcondresp "/fs space used" "Extend /fs filesystem"
```

We start monitoring by issuing the next command:

```
#startcondresp "/fs space used" "Extend /fs filesystem"
```

And, verify that the monitor is operational:

```
#lscondresp
   Displaying condition with response information:
   Condition            Response                Node        State
   "Inetd daemon state" "Extend /fs filesystem" "csmserver" "Active"

#lscondition | grep "/fs space used"
   "/fs space used"                    "csmserver" "Monitored and event
   monitored"
```

Step 4. Test the monitor

We test the monitor by filling the /fs filesystem so its utilization reaches 90%:

```
#cat /dev/zero > /fs/foo
#df -k | grep "/fs"
   /dev/fslv         1572864    133760   92%        5     1% /fs
```

After a short while we can see the *wall* from the *rmcchfs* script:

```
Broadcast message from root@csmserver (tty) at 16:17:07 ...

extension of /fs success
```

And, we verify the size of the filesystem:

```
#df -k | grep "/fs"
   /dev/fslv        3670016    524536   86%        5     1% /fs
```

6.5 OS migration using NIM alt_disk_install feature

In this section, we migrate our p690_LPAR1 from AIX 5.2 to AIX 5.3 from the management server using the *alt_disk_install* NIM feature. We prepare the environment, and then migrate the operating system to the inactive hard drive while our application is up and running. After the system has been upgraded, we reboot the node with AIX5.3 installed and perform the verification tests.

6.5.1 System preparation

Our node p690_LPAR1 has only two disks, and both belong to the *rootvg* volume group because of operating system mirroring created by the script in Example 6-11 on page 115. We manually unmirror the *rootvg* volume group and de-configure *hdisk1* from it as follows:

```
p690_LPAR1.itso.ibm.com:/:> unmirrorvg rootvg hdisk1
p690_LPAR2.itso.ibm.com:/:> reducevg rootvg hdisk1
p690_LPAR1.itso.ibm.com:/:> lspv
   hdisk0          0022be2a0edc421d                    rootvg active
   hdisk1          0022be2afeae99f4                    None
```

We also prepare the AIX 5.3 lppsource and SPOT. Refer to “NIM and the CSM management server” on page 139.

6.5.2 Operating system upgrade

We use the *nimadm_migrate* SMIT menu on our management server to perform the operating system migration using the alt_disk_install as shown in Example 6-18.

Example 6-18 SMIT nimadm_migration panel

```
Perform NIM Alternate Disk Migration

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

                                                      [Entry Fields]
* Target NIM Client                                  [p690_LPAR1]             +
* NIM LPP_SOURCE resource                            [530lpp_res]             +
* NIM SPOT resource                                  [530spot_res]            +
* Target Disk(s) to install                         [hdisk1]
  DISK CACHE volume group name                       []                       +

  NIM IMAGE_DATA resource                            []                       +
  NIM BOSINST_DATA resource                          []                       +
  NIM EXCLUDE_FILES resource                         []                       +
  NIM INSTALLP_BUNDLE resource                       []                       +
  NIM PRE-MIGRATION SCRIPT resource                  []                       +
  NIM POST-MIGRATION SCRIPT resource                 []                       +

  Phase to execute                                   [all]                    +
  NFS mounting options                               []
  Set Client bootlist to alternate disk?              yes                     +
  Reboot NIM Client when complete?                    yes                     +
  Verbose output?                                     no                      +
  Debug output?                                       no                      +

  ACCEPT new license agreements?                      yes                     +

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

The management server connects to the node and performs the alt_disk_install task using the AIX 5.3 lppsource and SPOT. The new operating system is created on hdisk1. Example 6-19 shows the output from the alt_disk_install operation.

Example 6-19 alt_disk_install operation on p690_LPAR1

```
MASTER DATE: Wed Oct  6 15:14:01 CDT 2004
CLIENT DATE: Wed Oct  6 15:14:13 CDT 2004
NIMADM PARAMETERS: -H -cp690_LPAR1 -l530lpp_res -s530spot_res -dhdisk1 -Y
Starting Alternate Disk Migration.

+-----------------------------------------------------------------------------+
Executing nimadm phase 1.
+-----------------------------------------------------------------------------+
Cloning altinst_rootvg on client, Phase 1.
Client alt_disk_install command: alt_disk_install -M 5.3 -C -P1 hdisk1
Calling mkszfile to create new /image.data file.
Checking disk sizes.
Creating cloned rootvg volume group and associated logical volumes.
Creating logical volume alt_hd5.
Creating logical volume alt_hd6.
...
Creating logical volume alt_paging00.
Creating /alt_inst/ file system.
...
Creating /alt_inst/var file system.
Generating a list of files
for backup and restore into the alternate file system...
Backing-up the rootvg files and restoring them to the alternate file system...
Phase 1 complete.

+-----------------------------------------------------------------------------+
Executing nimadm phase 2.
+-----------------------------------------------------------------------------+
Exporting alt_inst filesystems from client p690_LPAR1.itso.ibm.com
to NIM master csmserver2.itso.ibm.com:
Exporting /alt_inst from client.
...
Exporting /alt_inst/var from client.

+-----------------------------------------------------------------------------+
Executing nimadm phase 3.
+-----------------------------------------------------------------------------+
NFS mounting client's alt_inst filesystems on the NIM master:
Mounting p690_LPAR1.itso.ibm.com:/alt_inst.
...
Mounting p690_LPAR1.itso.ibm.com:/alt_inst/var.
```

```
+-----------------------------------------------------------------------------+
Executing nimadm phase 4.
+-----------------------------------------------------------------------------+
nimadm: There is no user customization script specified for this phase.

+-----------------------------------------------------------------------------+
Executing nimadm phase 5.
+-----------------------------------------------------------------------------+
Saving system configuration files.
Checking for initial required migration space.
Setting up for base operating system restore.
Restoring base operating system.
Restoring device ODM database.
Merging system configuration files.
Rebuilding inventory database.
Running migration merge method: ODM_merge Config_Rules.
Running migration merge method: ODM_merge SRCextmeth.
...
Running migration merge method: ODM_merge PdAt.
Running migration merge method: merge_smit_db.
Running migration merge method: SysckMerge.

+-----------------------------------------------------------------------------+
Executing nimadm phase 6.
+-----------------------------------------------------------------------------+
Installing and migrating software.
Checking space requirements for installp install.
Expanding /alt_inst/opt client filesystem.
File System size changed to 262144
Expanding /alt_inst/usr client filesystem.
File System size changed to 2097152
Installing software with the installp installer.
+-----------------------------------------------------------------------------+
                    Pre-installation Verification...
+-----------------------------------------------------------------------------+
Verifying selections...done
Verifying requisites...done
Results...
...

  Requisites
  ----------
  (being installed automatically;  required by filesets listed above)
  perl.libext 2.1.0.0                         # Perl Library Extensions

  << End of Success Section >>

FILESET STATISTICS
------------------
```

```
  487  Selected to be installed, of which:
      478  Passed pre-installation verification
        9  Already installed (directly or via superseding filesets)
    1  Additional requisites to be automatically installed
  ----
  479  Total to be installed

+-----------------------------------------------------------------------------+
                        Installing Software...
+-----------------------------------------------------------------------------+
...
install_all_updates: Result = SUCCESS

+-----------------------------------------------------------------------------+
Executing nimadm phase 7.
+-----------------------------------------------------------------------------+
nimadm: There is no user customization script specified for this phase.

+-----------------------------------------------------------------------------+
Executing nimadm phase 8.
+-----------------------------------------------------------------------------+
Creating client boot image.
bosboot: Boot image is 22221 512 byte blocks.
Writing boot image to client's alternate boot disk hdisk1.

+-----------------------------------------------------------------------------+
Executing nimadm phase 9.
+-----------------------------------------------------------------------------+
Unmounting client mounts on the NIM master.
forced unmount of /p690_LPAR1_alt/alt_inst/var
...
forced unmount of /p690_LPAR1_alt/alt_inst

+-----------------------------------------------------------------------------+
Executing nimadm phase 10.
+-----------------------------------------------------------------------------+
Unexporting alt_inst filesystems on client p690_LPAR1.itso.ibm.com:
unexported /alt_inst
...
unexported /alt_inst/var

+-----------------------------------------------------------------------------+
Executing nimadm phase 11.
+-----------------------------------------------------------------------------+
Cloning altinst_rootvg on client, Phase 3.
Client alt_disk_install command: alt_disk_install -M 5.3 -C -P3 hdisk1
## Phase 3 ###################
Verifying altinst_rootvg...
Modifying ODM on cloned disk.
```

```
forced unmount of /alt_inst/var
...
forced unmount of /alt_inst
Changing logical volume names in volume group descriptor area.
Fixing LV control blocks...
Fixing file system superblocks...
Bootlist is set to the boot disk: hdisk1

+-----------------------------------------------------------------------------+
Executing nimadm phase 12.
+-----------------------------------------------------------------------------+
Cleaning up alt_disk_migration on the NIM master.
Cleaning up alt_disk_migration on client p690_LPAR1.
```

6.5.3 Verification of the nodes

After the node has been upgraded, we verify the physical disk configuration, check the bootlist and reboot the node:

```
p690_LPAR1.itso.ibm.com:/:> lspv
   hdisk0          0022be2a0edc421d                    rootvg          active
   hdisk1          0022be2afeae99f4                    altinst_rootvg
p690_LPAR1.itso.ibm.com:/:> bootlist -m normal -o
   hdisk1
p690_LPAR1.itso.ibm.com:/:> shutdown -Fr
```

The *altinst_rootvg* volume group contains the upgraded operating system. The *rootvg* volume group contains the current AIX 5.2. Now we can boot the node with the operating system of our choice.

After the reboot, we verify the installation. First we verify the physical disk configuration:

```
p690_LPAR1.itso.ibm.com:/:> lspv
   hdisk0          0022be2a0edc421d                    old_rootvg
   hdisk1          0022be2afeae99f4                    rootvg          active
```

As you can the *rootvg* volume group is active on hdisk1. Our old AIX 5.2 exists on hdisk0 in *old_rootvg* volume group.

We verify the operating system level by issuing the following commands:

```
p690_LPAR1.itso.ibm.com:/:> oslevel
   5.3.0.0
p690_LPAR1.itso.ibm.com:/:> instfix -i | grep AIX_ML
       All filesets for 5.3.0.0_AIX_ML were found.
```

Next, we verify the lpp consistency:

```
lppchk -v
```

There is no output from this command meaning the filesets' versions are consistent.

Finally, we check if our *Apache* application is up and running:

```
p690_LPAR1.itso.ibm.com:/:> ps -ef | grep http
     nobody 14982 15740   0 16:53:49      -  0:00 /usr/sbin/httpd start
     nobody 15496 15740   0 16:53:49      -  0:00 /usr/sbin/httpd start
       root 15740     1   0 16:53:48      -  0:00 /usr/sbin/httpd start
     nobody 16516 15740   0 16:53:49      -  0:00 /usr/sbin/httpd start
     nobody 16772 15740   0 16:53:49      -  0:00 /usr/sbin/httpd start
     nobody 17032 15740   0 16:53:49      -  0:00 /usr/sbin/httpd start
```

7

POWER5 provisioning scenario

In this chapter, we describe the installation of the POWER5 system in our environment. We describe the tools for provisioning and the benefits that are related to this new technology. We use AIX 5L V5.3, so we can utilize the new features of server virtualization provided with the POWER5 systems.

This scenario takes advantage of the following tools:

- Virtual IO
- Multiple networks in NIM
- Dynamic LPAR with micro-partitions
- Service Update Management Assistant (SUMA)
- Partition Load Manager (PLM)

Note: At the time of writing, there is no official support for POWER5 Virtual I/O devices in CSM. IBM plans to introduce support in 2005.

7.1 Hardware preparation

For the scenario, we use the following hardware:

- p630 - the Management Server running AIX 5.3. This server is used for the nodes installation and system management.
- HMC - workstation running Linux. This server is used for hardware maintenance and hardware control.
- pSeries 520: Two POWER5 CPUs, 2 GB of memory. This server has three partitions: one Virtual IO (VIO) server and two client LPARs running AIX 5.3.

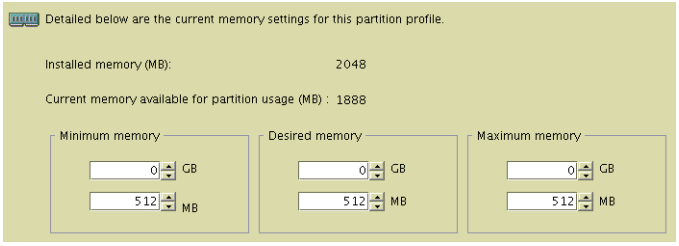
HMC setup and LPAR preparation

On our HMC, we create three partitions, one VIO server and two virtual LPARS. The VIO server has the following resources:

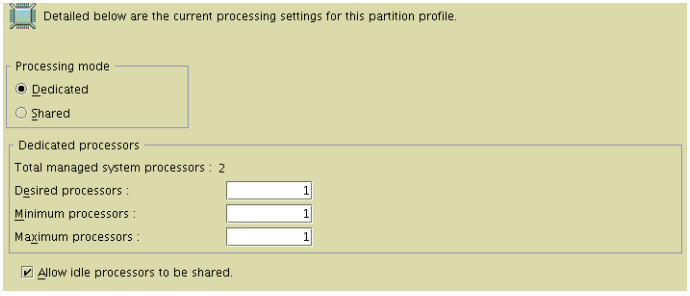
VIO LPAR settings

We setup the VIO partition with the following settings specified:

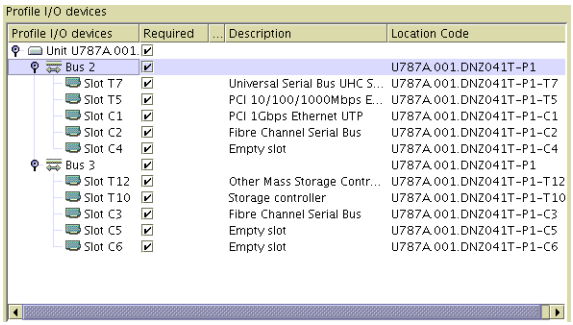
- Memory



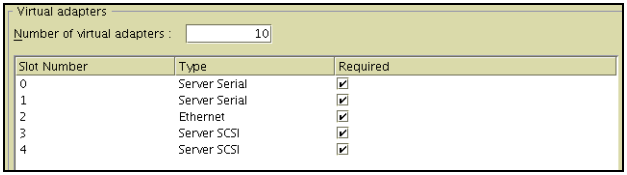
- Dedicated CPU



- Physical adapters



- Virtual adapters



Virtual LPAR settings

We create two LPARS: c97a3l4vp01 and c97a3l4vp01. The LPARs are *uncapped*, use *shared processor mode* processing mode, and have the following characteristics as shown in Table 7-1.

Table 7-1 Scenario virtual LPAR settings

Resource	c97a3l4vp01	c97a3l4vp01
CPU	minimum: 0.1 desired: 0.2 maximum: 1	minimum: 0.1 desired: 0.3 maximum: 1
Memory	minimum: 512MB desired: 512MB maximum: 1GB	minimum: 512MB desired: 512MB maximum: 1GB
Weight	128	178

7.2 Installation of Virtual LPARs

This section describes the steps that we take in order to install the AIX 5.3 operating system on the virtual LPARs, and include them in our cluster. For

information how to setup the virtual devices, refer to 7.3, “Virtual I/O devices” on page 146.

7.2.1 HMC definition to CSM

Use the **systemid** command to create the hardware control point for the new HMC, on the CSM server:

```
csmserver:> systemid c76hmc04.ppd.pok.ibm.com hscroot
   Password:
   Verifying, please re-enter password:
   systemid: Entry created.
```

7.2.2 NIM setup for the new environment

The pSeries 520 is located in a different building. It is connected to a different HMC than our p690 server installed in Chapter 6, “POWER4 provisioning scenario” on page 95. Also the network subnets are different, thus we update the NIM configuration with the new networks.

Figure 7-1 shows our scenario’s network configuration.

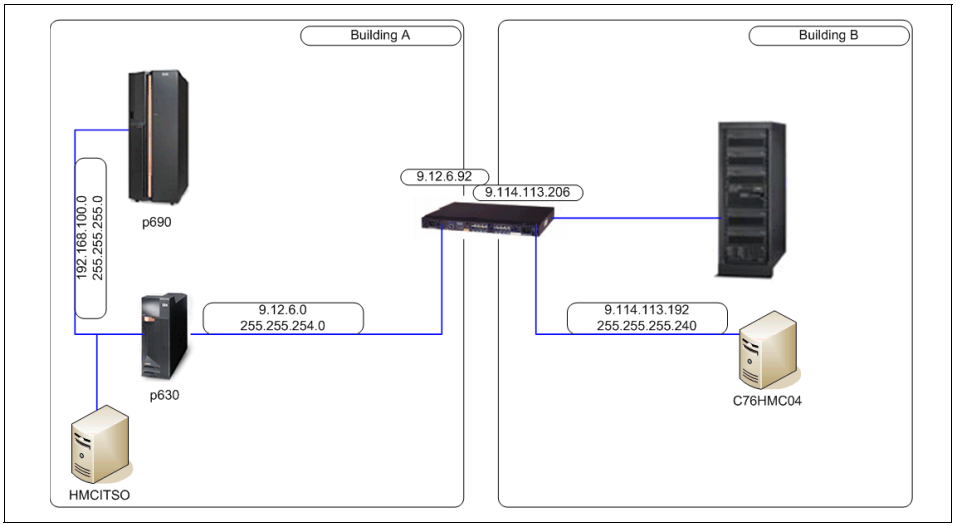


Figure 7-1 Scenario network diagram

As you notice in Figure 7-1, the server p520 is connected to a different subnet than the management server. As the VIO server is already installed, we use NIM to install the two virtual LPARS. In order to accomplish this task, the network configuration in NIM must be updated.

We define a network NIM resource for the remote subnet using the *mk_net* SMIT panel as shown in Example 7-1.

Example 7-1 Defining a NIM network resource

```
Define a Network

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

                                                    [Entry Fields]
* Network Name                                     [9_114_net]
* Network Type                                      ent
* Ethernet Type                                     Standard              +
* Network IP Address                               [9.114.113.192]
* Subnetmask                                       [255.255.255.240]
  Default Gateway for this Network                 [9.114.113.206]
  Other Network Type                                                     +
  Comments                                         []

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

The other networks are already created from our POWER4 scenario. Refer to the Chapter 6, “POWER4 provisioning scenario” on page 95 for the network configuration.

NIM and the CSM management server

The management server is already set up and running on AIX 5L V5.3 as it is described in Chapter 6, “POWER4 provisioning scenario” on page 95. However the NIM server manages only AIX v 5.2 resources. For our POWER5 machines, we install AIX V5.3, and because of this we have to create a new AIX 5L V5.3 lppsource as shown in Example 7-2.

Example 7-2 530lpp_res lppsource creation

```
Define a Resource

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

                                                    [Entry Fields]
* Resource Name                                    [530lpp_res]
* Resource Type                                     lpp_source
* Server of Resource                               [master]              +
```

```
* Location of Resource                                [/export/nim/lpp_source>
/
  Architecture of Resource                            [power]                  +
  Source of Install Images                            [/export/lppsource/AIX> +/
  Names of Option Packages                            []
  Show Progress                                       [yes]                    +
  Comments                                            []

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

We add the *openssh* and *openssl* packages to the new lppsource the same way as we did for the AIX v 5.2 lppsource. We use the *530lpp_res* lppsource to create the SPOT as shown in Example 7-3.

Example 7-3 530spot_res SPOT creation

```
Define a Resource

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

                                                      [Entry Fields]
* Resource Name                                      [530spot_res]
* Resource Type                                       spot
* Server of Resource                                 [master]                 +
* Source of Install Images                           [530lpp_res]             +
* Location of Resource                               [/export/nim/spot]        /
  Expand file systems if space needed?                yes                     +
  Comments                                           []

  installp Flags
  COMMIT software updates?                            no                      +
  SAVE replaced files?                                yes                     +
  AUTOMATICALLY install requisite software?           yes                     +
  OVERWRITE same or newer versions?                   no                      +
  VERIFY install and check file sizes?                no                      +

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

We create a new resource group with the AIX 5L V5.3 lppsource and SPOT with the *bosinst_data* and *master_net_conf* resources as follows:

```
csmserver:/:> lsnim -l basic_p5_res_group
   basic_p5_res_group:
```

```
class   = groups
type    = res_group
member1 = 530spot_res
member2 = 530lpp_res
member3 = bid_ow
member4 = master_net_conf
```

CSM definition

We use a node stanza file to feed the nodes to the **definenode** command. First we create /tmp/mymapfile:

```
csmserver:> lshwinfo -s -v -o - -c 9.114.113.68 -p \
hmc  | grep LPAR > /tmp/mymapfile
csmserver:> cat /tmp/mymapfile
   c97a3l4vp01.ppd.pok.ibm.com::hmc::c76hmc04.ppd.pok.ibm.com::c97a3l4vp01::1:
   :9111::520::106D84D
   c97a3l4vp02.ppd.pok.ibm.com::hmc::c76hmc04.ppd.pok.ibm.com::c97a3l4vp02::2:
   :9111::520::106D84D
```

In most cases, you must edit and add the host name of your nodes manually.

Next, we feed the **definenode** command with previously created /tmp/mymapfile file:

```
definenode -M /tmp/mymapfile ManagementServer=csmserver InstallOSName=AIX
   Defining CSM Nodes:
   Defining Node "c97a3l4vp01.ppd.pok.ibm.com"("9.114.113.202")
   Defining Node "c97a3l4vp02.ppd.pok.ibm.com"("9.114.113.203")
```

Next, we create node group:

```
nodegrp -a c97a3l4vp01.ppd.pok.ibm.com,c97a3l4vp02.ppd.pok.ibm.com
p520_nodegroup
```

And, check if the communication between the management server and HMC works fine:

```
csmserver:/tmp:> rpower -a -l query
   p690_LPAR1.itso.ibm.com on   LCDs are blank
   p690_LPAR2.itso.ibm.com on   LCDs are blank
   c97a3l4vp01.ppd.pok.ibm.com off   LCD1 = 00000000   LCD2 is blank
   c97a3l4vp02.ppd.pok.ibm.com off   LCD1 = 00000000   LCD2 is blank
```

Getting the network adapter MAC address via the **getadapters** command does not work for LPARs using shared Ethernet adapter for the install network.

Note: At the time of writing CSM does not support virtual I/O adapters.

We collect the adapter information manually from the HMC and create a stanza file for the new nodes. We do this by powering on the LPAR to SMS menu, open the terminal and gather the adapter information from the RIPL menu. Refer to Example 7-4.

Example 7-4 p520_stanzafile

```
###CSM_ADAPTERS_STANZA_FILE###--do not remove this line
#---Stanza Summary----------------------
#   Date: Thu Sep 23 12:09:25 CDT 2004
#   Stanzas Added: 0
#   Stanzas Updated: 0
#---End Of Summary----------------------
c97a3l4vp01.ppd.pok.ibm.com:
     MAC_address=6220e0001002
     adapter_duplex=full
     adapter_speed=100
     cable_type=N/A
     install_gateway=9.114.113.206
     install_server=9.12.6.76
     location=U9111.520.106D84D-V1-C2-T1
     machine_type=install
     netaddr=
     network_type=en
     ping_status=ok
     subnet_mask=

c97a3l4vp02.ppd.pok.ibm.com:
     MAC_address=6220e0002002
     adapter_duplex=full
     adapter_speed=100
     cable_type=N/A
     install_gateway=9.114.113.206
     install_server=9.12.6.76
     location=U9111.520.106D84D-V2-C2-T1
     machine_type=install
     netaddr=
     network_type=en
     ping_status=ok
     subnet_mask=
```

Now, we feed the CSM database with the stanzafile:

```
csmserver:/:> getadapters -w -f /tmp/p520_stanzafile
```

NIM definition of the nodes

We use the *csm_nimnodes* SMIT panel to input the appropriate information as shown in Example 7-5. This time the CSM node group is used to define all new nodes in one step.

Example 7-5 csm_nimnodes SMIT panel

```
Convert CSM to NIM Nodes

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

                                                      [Entry Fields]
  Setup all nodes                                     no                      +
   OR
  List of nodes                                      []                       +
   OR
  List of groups                                     [p520_nodegroup]         +

  Update existing NIM group definitions               yes                     +
  NIM machine type                                    standalone              +
  NIM network name                                   [9_114_net]              +
  Ring speed (required if token ring)                []                       +
  Cable type (required if ethernet)                  [tp]                     +
  Platform                                            chrp                    +
  Netboot kernel                                      mp                      +
  Display Verbose Messages?                           no                      +

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

As the installp bundles for *openssh* and *openssl* are already created and the fileset names are the same in AIX 5L V5.3, we can use the same bundle definition files and NIM resources to allocate them for the new nodes.

We allocate the *fb_script* for the nodes as well which will customize the name resolution via changing the netsvc.conf file on the installed node.

We use the same application to install for the new nodes as in the POWER4 scenario. For details refer to 6.1.5, “Automatic node customization and application deployment” on page 113.

The node customization does the following after initial node installation:

- Create the .profile file on the installed node
- Create the /etc/hosts file

- Installs the *Apache* application
- Adds the automatic startup of the application to the /etc/inittab file
- Performs *rootvg* volume group mirroring

Next, we allocate the CSM customization script to the p520_nodegroup:

```
csmserver:/tmp:> csmsetupnim -N p520_nodegroup
```

7.2.3 Install the operating system

This section contains the necessary steps to install the operating system on the nodes.

Allocating the NIM resources

We allocate the bundles, *fb_script*, AIX 5L V5.3 lppsource, and SPOT resources to the nodes and set them to install as follows:

```
csmserver:> nim -o bos_inst -a source=rte -a boot_client=no \
-a group=basic_p5_res_group -a fb_script=set_netsvc_script \
-a installp_bundle=openssl -a installp_bundle=openssh \
c97a3l4vp01 c97a3l4vp02<
```

Network boot the nodes

We initiate network boot and installation of the nodes by issuing the following command.

```
csmserver:> netboot -N p520_nodegroup
```

Trying the CSM command **netboot** will fail because of lack of shared Ethernet adapter support. The netboot log file contains the following:

```
16:52:13: Nodecond Status: power on the node to Open Firmware
16:52:15: Nodecond Status: wait for power on
16:52:34: Nodecond Status: power on complete
16:52:34: Nodecond Status: waiting for RS/6000 logo
16:52:35: Nodecond Status: at RS/6000 logo
16:52:37: Nodecond Status: at ok prompt
16:52:37: Nodecond Status: Starting to get full device name and phandle
16:52:37: Nodecond Status: sending dev / command
16:52:39: Nodecond Status: at root
16:52:39: Nodecond Status: sending ls command
16:52:40: Nodecond Status: Parsing device tree
16:52:41: No network adapters found
16:52:41: return code -1 from get_phandle. Quitting.
16:52:41: Nodecond Status: clean-exit called with code 1
```

Because of this we have to do a manual netboot via the HMC as shown in Figure 7-2. The process is the following:

- Open an IBM Web-based System Manager connection to the HMC
- Shut down the partition if it is running
- Start the partition in SMS mode

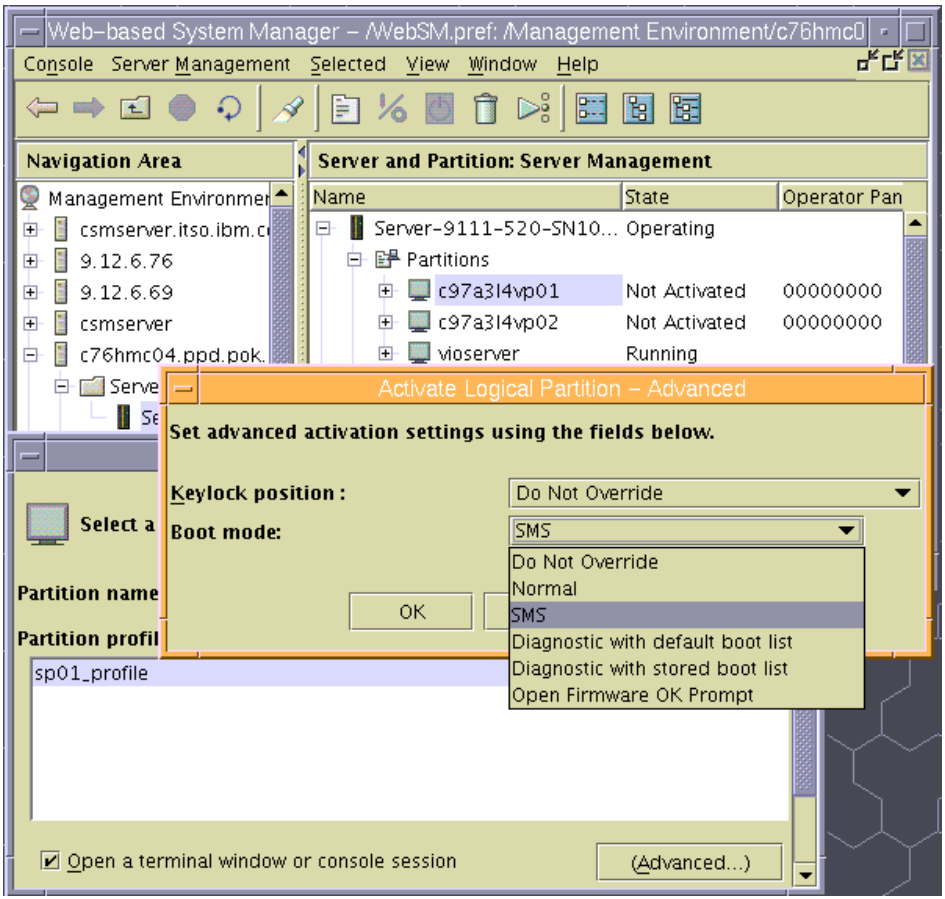


Figure 7-2 Starting POWER5 partition

- In SMS menu setup the remote IPL as shown in Example 7-6.

Example 7-6 `Setup remote IPL in SMS`

```
Version SF220_012
SMS 1.5 (c) Copyright IBM Corp. 2000,2003 All rights reserved.
-------------------------------------------------------------------------------
IP Parameters
```

```
Interpartition Logical LAN: U9111.520.106D84D-V1-C2-T1
 1.  Client IP Address                  [9.114.113.202]
 2.  Server IP Address                  [9.12.6.76]
 3.  Gateway IP Address                 [9.114.113.206]
 4.  Subnet Mask                        [255.255.255.240]

-------------------------------------------------------------------------------
Navigation keys:
M = return to Main Menu
ESC key = return to previous screen         X = eXit System Management Services
-------------------------------------------------------------------------------
Type the number of the menu item and press Enter or select Navigation Key:
```

- Execute a ping test to make sure the network connection exists between the node and the management server.
- To initiate the network boot use the *Select Boot Options -> Select Install/Boot Device -> Network* panel, and then select the install adapter as shown in Example 7-7. In our scenario, we have only one virtual adapter assigned to the LPAR. This adapter is mapped to a physical interface in the Virtual I/O server.

Example 7-7 Select install adapter

```
Version SF220_012
SMS 1.5 (c) Copyright IBM Corp. 2000,2003 All rights reserved.
-------------------------------------------------------------------------------
Select Device
Device  Current  Device
Number  Position  Name
 1.        -      Virtual Ethernet
                  ( loc=U9111.520.106D84D-V1-C2-T1 )

-------------------------------------------------------------------------------
Navigation keys:
M = return to Main Menu
ESC key = return to previous screen         X = eXit System Management Services
-------------------------------------------------------------------------------
Type the number of the menu item and press Enter or select Navigation Key:
```

7.3 Virtual I/O devices

In this section, we dynamically create a Virtual Ethernet device in the virtual LPAR, assign the IP address, do a ping test, and then dynamically remove the virtual adapter.

7.3.1 Step 1. Verify the list of Ethernet devices

Verify the list of Ethernet devices by issuing the **lsdev** command:

```
c97a3l4vp01.ppd.pok.ibm.com:/:> lsdev -C | grep Ethernet
   en0        Available       Standard Ethernet Network Interface
   ent0       Available       Virtual I/O Ethernet Adapter (l-lan)
   et0        Defined         IEEE 802.3 Ethernet Network Interface
```

7.3.2 Step 2. Create the virtual Ethernet device

We select the *Dynamic LPAR -> Virtual Adapter Resources -> Add/Remove* menu as shown in Figure 7-3.

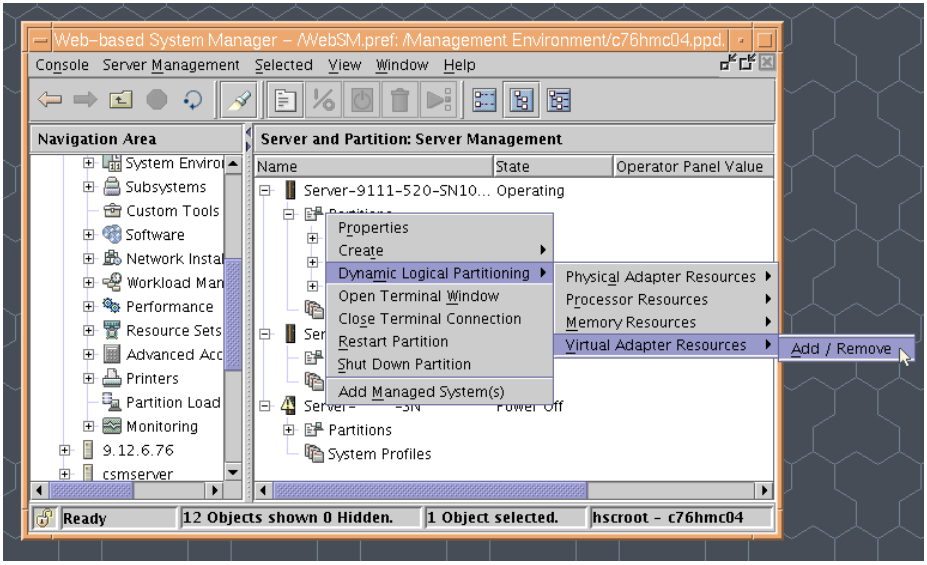


Figure 7-3 Add Virtual adapter menu.

We create the virtual Ethernet device in the LPAR by selecting the free slot and VLAN. VLAN number one is our default VLAN and has connectivity to the outside network via the VIO LPAR. See Figure 7-4 on page 148.

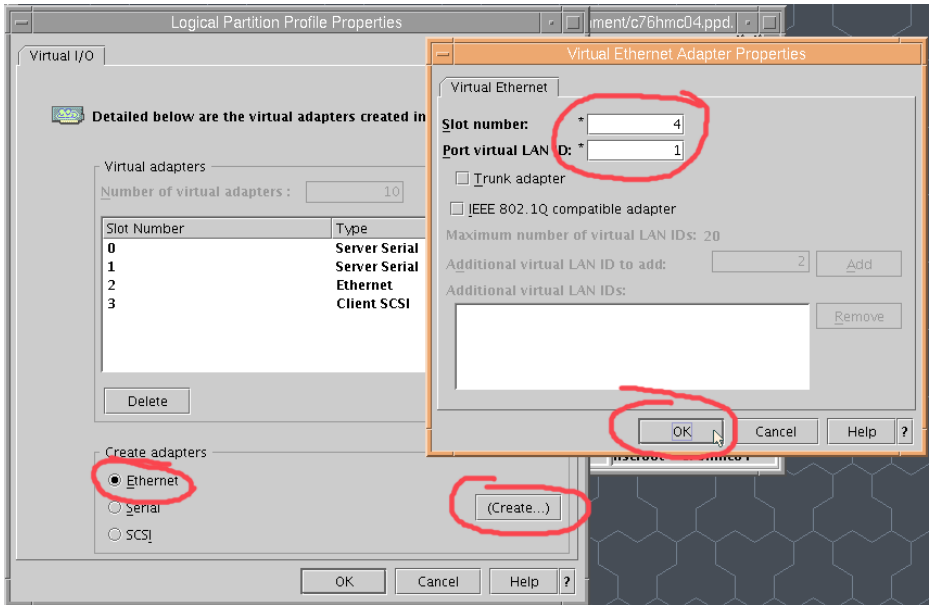


Figure 7-4 Create virtual ethernet device

7.3.3 Configure and verify the Ethernet device

We rescan the bus by issuing the **cfgmgr** command and verify the device list:

```
c97a3l4vp01.ppd.pok.ibm.com:/:> lsdev -C | grep Ethernet
   en0        Available       Standard Ethernet Network Interface
   en1        Defined         Standard Ethernet Network Interface
   ent0       Available       Virtual I/O Ethernet Adapter (l-lan)
   ent1       Available       Virtual I/O Ethernet Adapter (l-lan)
   et0        Defined         IEEE 802.3 Ethernet Network Interface
   et1        Defined         IEEE 802.3 Ethernet Network Interface
```

We assign the IP address:

```
chdev -l 'en1' -a netaddr='9.114.113.204' -a netmask='255.255.255.0' \
-a state='up'
```

And, do a ping test from our management server:

```
csmserver:/:> ping 9.114.113.204
   PING 9.114.113.204: (9.114.113.204): 56 data bytes
   64 bytes from 9.114.113.204: icmp_seq=0 ttl=245 time=1 ms
   64 bytes from 9.114.113.204: icmp_seq=1 ttl=245 time=1 ms
   ...
   ...
```

7.3.4 Dynamically remove the Ethernet device

We remove the *ent1* device from the system by issuing the **rmdev** command:

```
c97a3l4vp01.ppd.pok.ibm.com:/:> ifconfig en1 detach
c97a3l4vp01.ppd.pok.ibm.com:/:> rmdev -dl ent1 -R
   ent1 deleted
c97a3l4vp01.ppd.pok.ibm.com:/:> rmdev -dl et1 -R
   et1 deleted
c97a3l4vp01.ppd.pok.ibm.com:/:> rmdev -dl en1 -R
   en1 deleted
```

And, then we remove the device on the HMC, see Figure 7-5.

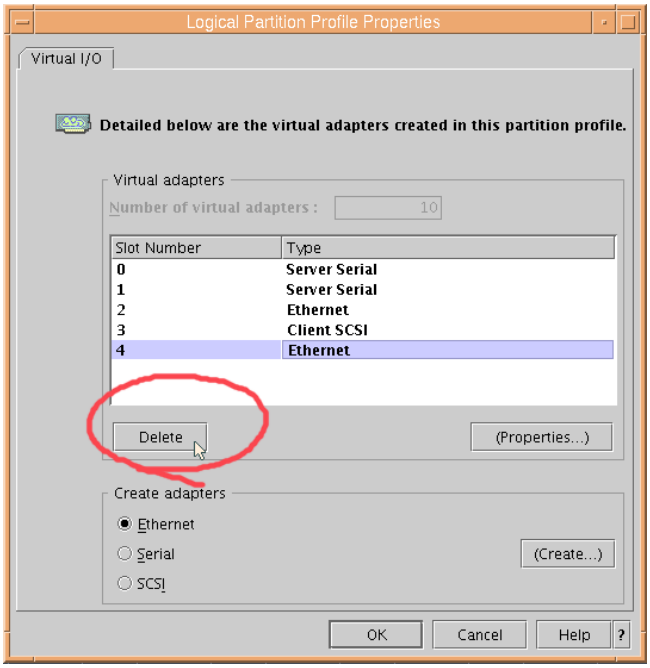


Figure 7-5 Remove the Ethernet device

7.4 Service Update Management Assistant (SUMA)

In this scenario, we configure the SUMA system task to download the latest critical fixes to AIX 5L V5.3. We use our management server to download the fixes, store the filesets in the /export/FIXES directory. The files can then be used to update the management server, lppsource, spot and then applied them on the nodes using NIM. For more information about SUMA refer to 4.2.7, “Service Update Management Assistant (SUMA)” on page 44.

7.4.1 Create new SUMA task

We issue the smitty *suma -> Custom/Automated Downloads (Advanced) -> Create a New SUMA Task* panel as shown in Example 7-8 to create new SUMA task.

Example 7-8 Create new SUMA task

```
Create a New SUMA Task

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

                                                      [Entry Fields]
     Save and Execute Now
  Display name                                       [critical_fixes]
  Action                                             [Download]             +
  Directory for item storage                         [/export/FIXES]        +
  Type of item to request                            [Critical]             +
  Name of item to request                            []
  Level of item to request                           []
  Get prerequisites/corequisites?                     yes                   +
  Get ifrequisites?                                   yes                   +
  Get superseding items?                              no                    +
  Get items which fix regressions?                   [If Available]         +
  Repository to filter against                       [/export/FIXES]
  Maintenance level to filter against                []                     +
  System or lslpp output to filter against           [localhost]
  Maximum total download size (MB)                   [-1]                   +#
  EXTEND file systems if space needed?                yes                   +
  Maximum file system size (MB)                      [-1]                   +#


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

Example 7-8 creates a SUMA task that downloads all latest critical fixes for the base level of AIX 5L V5.3, and stores them in /export/FIXES directory. See Example 7-9.

Example 7-9 Download success

```
COMMAND STATUS

Command: OK            stdout: yes           stderr: no

Before command completion, additional instructions may appear below.
```

```
[MORE...107]
.1.bff
Download SUCCEEDED:
/export/FIXES/installp/ppc/devices.vtdev.scsi.rte.5.3.0.1.bf
f
Download SUCCEEDED: /export/FIXES/installp/ppc/invscout.ldb.2.2.0.1.bff
Download SUCCEEDED: /export/FIXES/installp/ppc/invscout.rte.2.2.0.1.bff
Download SUCCEEDED: /export/FIXES/installp/ppc/sysmgt.websm.apps.5.3.0.1.bff
Download SUCCEEDED:
/export/FIXES/installp/ppc/sysmgt.websm.framework.5.3.0.1.bf
f
Download SUCCEEDED: /export/FIXES/installp/ppc/sysmgt.websm.rte.5.3.0.1.bff
Download SUCCEEDED:
/export/FIXES/installp/ppc/sysmgt.websm.webaccess.5.3.0.1.bf
f
Summary:
        82 downloaded
        0 failed
        0 skipped

[BOTTOM]

F1=Help             F2=Refresh          F3=Cancel           F6=Command
F8=Image            F9=Shell            F10=Exit            /=Find
n=Find Next
```

The fixes are downloaded and ready to install.

7.5 Partition Load Manager (PLM)

Partition Load Manager (PLM) can be used as one of the provisioning tools. In this section, we install, configure, and test the PLM software in our sample environment. The setup and implementation of PLM can be automated using workflows and scripts, although in this scenario we use GUI panels for some tasks.

We install the PLM software on the management server, setup the secure connection to the HMC which is managing two virtual logical partitions, create sample policy file, and manage the dynamic resource migration.

7.5.1 PLM installation and configuration

In this section, we install and setup PLM in our environment.

Step 1. PLM installation

We install PLM version 1.1.0.0 bffs on the management server.

Step 2. Setup the secure connection with the HMC

We add the management server's public *ssh2* key to hscroot's authorized_keys2 file on the c76hmc04.ppd.pok.ibm.com HMC server. Refer to "Step 2. Configure the SSH connection" on page 121.

Step 3: Create the PLM policy file

The PLM policy file can be created either using standard text editor or guided IBM Web-based System Manager panels. Our policy file is presented in the Example 7-10.

Example 7-10 p520_policyfile

```
globals:
        hmc_host_name = c76hmc04.ppd.pok.ibm.com
        hmc_user_name = hscroot
        hmc_cec_name = Server-9111-520-SN106D84D
        hmc_command_wait = 5

tunables:
        cpu_free_unused = yes
        mem_free_unused = yes

p520_plmgroup:
        type = group
        cpu_type = shared
        cpu_maximum = 2
        mem_maximum = 2048
        cpu_free_unused = yes
        mem_free_unused = yes

c97a3l4vp01:
        type = partition
        group = p520_plmgroup
        cpu_guaranteed = 0.2
        cpu_maximum = 1
        cpu_minimum = 0.1
        cpu_shares = 20
        mem_guaranteed = 512
        mem_maximum = 1024
        mem_minimum = 256
        mem_shares = 20

c97a3l4vp02:
        type = partition
```

```
group = p520_plmgroup
cpu_guaranteed = 0.3
cpu_maximum = 1
cpu_minimum = 0.2
cpu_shares = 60
mem_guaranteed = 512
mem_maximum = 1024
mem_minimum = 256
mem_shares = 60
```

Step 4. Setup management of logical partitions

We use IBM Web-based System Manager panel to allow root user on the management server to connect to the client LPARs via RMC. See Example 7-6.

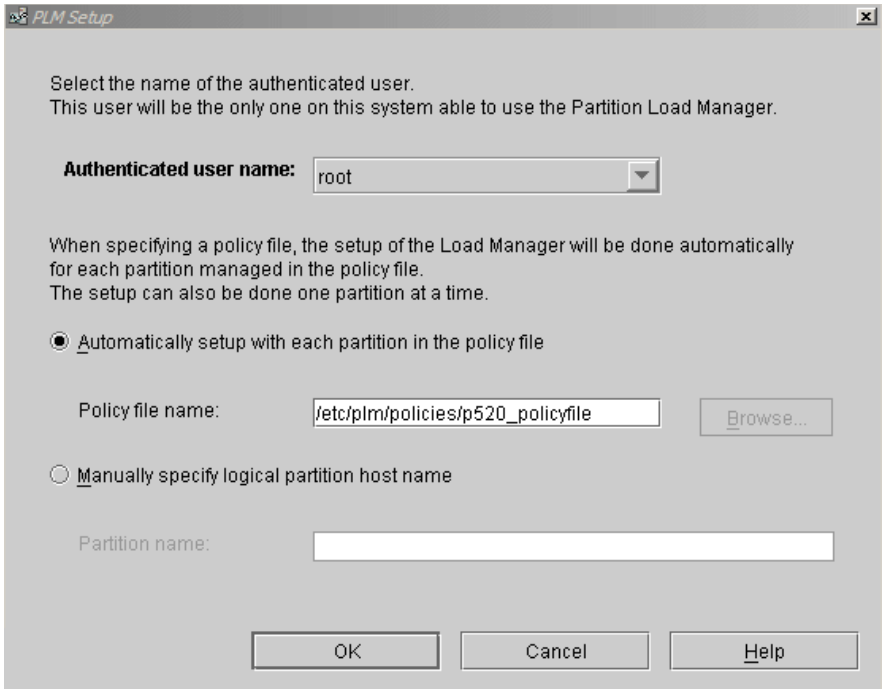


Figure 7-6 Setup management of logical partitions

Step 5. Start the PLM server

We start the PLM instance using the Web-based System Manager panel. See Example 7-7 on page 154.

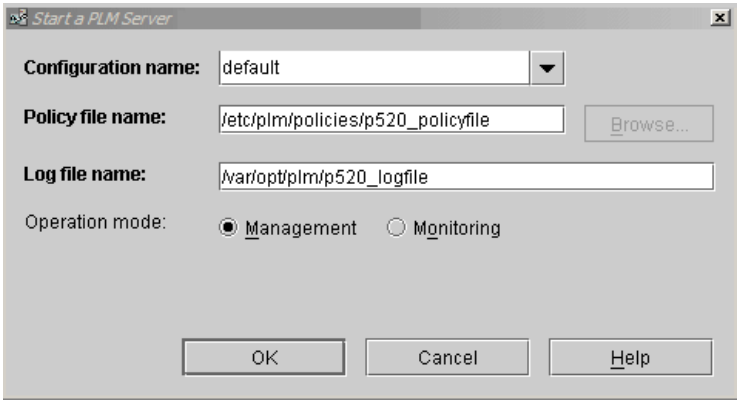


Figure 7-7 Start PLM server

Step6. Verify the RMC on the nodes

We verify the RMC settings on one of the nodes by issuing the following command:

```
c97a3l4vp02.ppd.pok.ibm.com:/:> lsrsrc "IBM.LPAR"
   Resource Persistent Attributes for IBM.LPAR
   resource 1:
           Name              = "c97a3l4vp02"
           LPARFlags         = 7
           MaxCPU            = 2
           MinCPU            = 1
           CurrentCPUs       = 1
           MinEntCapacity    = 0.1
           MaxEntCapacity    = 1
           CurEntCapacity    = 0.21
           MinEntPerVP       = 0.1
           SharedPoolCount   = 0
           MaxMemory         = 1024
           MinMemory         = 512
           CurrentMemory     = 512
           CapacityIncrement = 0.01
           LMBSize           = 16
           VarWeight         = 178
           CPUIntvl          = 6
           MemIntvl          = 6
           CPULoadMax        = 1
           CPULoadMin        = 0.4
           MemLoadMax        = 90
           MemLoadMin        = 50
           MemPgStealMax     = 0
           ActivePeerDomain  = ""
           NodeNameList      = {"c97a3l4vp02.ppd.pok.ibm.com"}
```

7.5.2 Dynamic system reconfiguration with PLM

When our PLM server is started, we can see the partition statistics as shown in Example 7-8.

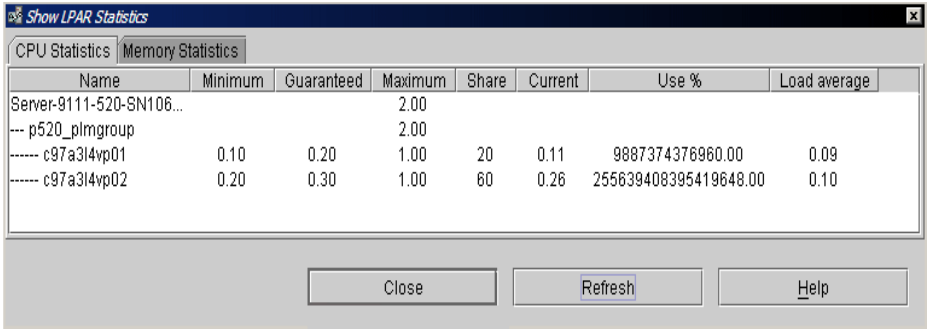


Figure 7-8 LPAR statistics

As you can see in Example 7-8, c98a3l4vp02 has 0.26 CPU shares now. We run the CPU stress script on the node c98a3l4vp01 and observe the effect shown in Example 7-11.

Example 7-11 Effects of running a CPU stress script

```
c97a3l4vp01.ppd.pok.ibm.com:/home/cpuStress:> ./cpuStress.ksh
tail -f p520_logfile
   <10/01/04 14:08:19> <PLM_TRC> Added 0 virtual CPUs and 0.02 units of CPU
   capacity for c97a3l4vp02 .
   <10/01/04 14:08:19> <PLM_TRC> Event notification of ConfigChanged for
   c97a3l4vp02 .
   <10/01/04 14:08:19> <PLM_TRC> Current number of CPUs is 1 for c97a3l4vp02 .
   <10/01/04 14:08:19> <PLM_TRC> Current CPU entitlement is 0.27 for
   c97a3l4vp02 .
   <10/01/04 14:08:19> <PLM_TRC> Current memory is 512 MBs for c97a3l4vp02 .
   <10/01/04 14:08:26> <PLM_TRC> Command received: QUERY
   <10/01/04 14:08:26> <PLM_TRC> Command received: QUERY
   ...
   <<10/01/04 14:23:32> <PLM_TRC> Added 1 virtual CPUs and 0.07 units of CPU
   capacity for c97a3l4vp02 .
   <10/01/04 14:23:32> <PLM_TRC> Event notification of ConfigChanged for
   c97a3l4vp02 .
   <10/01/04 14:23:32> <PLM_TRC> Current number of CPUs is 1 for c97a3l4vp02 .
   <10/01/04 14:23:32> <PLM_TRC> Current CPU entitlement is 0.81 for
   c97a3l4vp02 .
   <10/01/04 14:23:32> <PLM_TRC> Current memory is 512 MBs for c97a3l4vp02 .
   <10/01/04 14:23:32> <PLM_TRC> Event notification of ConfigChanged for
   c97a3l4vp02 .
   <10/01/04 14:23:32> <PLM_TRC> Current number of CPUs is 2 for c97a3l4vp02 .
```

```
<10/01/04 14:23:32> <PLM_TRC> Current CPU entitlement is 0.81 for
c97a3l4vp02 .
<10/01/04 14:23:32> <PLM_TRC> Current memory is 512 MBs for c97a3l4vp02 .
<10/01/04 14:23:32> <PLM_TRC> Event notification of ConfigChanged for
c97a3l4vp02 .
<10/01/04 14:23:32> <PLM_TRC> Current number of CPUs is 2 for c97a3l4vp02 .
<10/01/04 14:23:32> <PLM_TRC> Current CPU entitlement is 0.81 for
c97a3l4vp02 .
<10/01/04 14:23:32> <PLM_TRC> Current memory is 512 MBs for c97a3l4vp02 .
```

The PLM adds resources from the free pool to the node.

Tip:

- When we generate extremely high CPU loads, the RMC subsystem on the node seems to fail to send events to the PLM server, and in effect no resources are added.
- Dynamic LPAR removes do not occur on busy nodes when there is contention, even if the shares indicate that a remove should be run.

8

pSeries provisioning in an on demand world

This chapter discusses pSeries provisioning. We present possible directions and new technologies which can be used in pSeries provisioning, in these topics:

8.1 Open standards for provisioning

Here, we discuss the possible solutions and tools that can be used for provisioning and implemented using open standards.

Some of them are already integrated into off the shelf products, but with limited functionality at this time. We can see a lot of development in this area, and all the development is going in the direction of more automated provisioning and management solutions.

8.1.1 openPegasus and openCIMOM

First, we provide some explanations about the terms used throughout this chapter:

Common Information Model (CIM) is the data description and organization standard of the Desktop Management Task Force (DMTF). It uses object-oriented paradigm to describe how computer hardware and software resources are represented and managed.

CIM defines the information model while Web-Based Enterprise Management (WBEM) defines the protocols used to communicate with a particular CIM implementation that uses CIM servers. Managed Object Format (MOF) is an object description language defined by DMTF. MOF enables exchange of information between management applications.

The CIM provider is a software that links the CIM server and the system interfaces to allow the CIM server to access and manage the resources. A CIM provider implements a particular portion of a WBEM profile.

A **WBEM** profile is a collection of CIM elements and behavior rules that represent a specific area of management. The purpose of a WBEM profile is to ensure interoperability. Two WBEM profiles are provided by IBM for the management of IBM @server operating systems:

- @server operating system (OS) management

  Gives the possibility of accessing the operating systems properties. The function of it is read access only to core IT resources and their relationship.

- @server operating system monitoring

  Provides monitoring of the IT resources managed by the @server operating system management profile.

OpenPegasus

@server operating system CIM instrumentation includes the OpenPegasus CIM server and CIM providers that implement the WBEM profiles for @server operating system management and @server operating system monitoring.

The providers in the OpenPegasus package were developed using the Common Manageability Programming Interface (CMPI) specification.

Figure 8-1 shows the architecture of OpenPegasus.

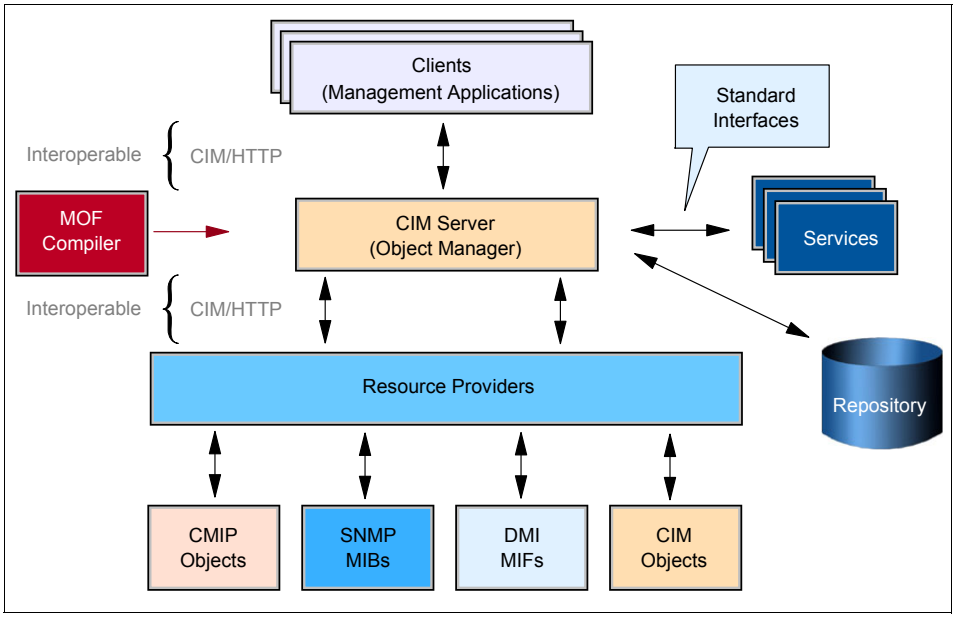


Figure 8-1 OpenPegasus architecture

The clients which could be a browser or a management application communicate with the CIM server via CIM operations over HTTP protocol. The communication is XML encoded. The client could run on the CIM server as well.

The CIM server in OpenPegasus provides registration services for providers and service extensions. It makes the routing between the other components in the management environment. It also maintains security with the possibility of modular extension.

The providers makes the connection between the CIM environment and the managed resources. The managed resources can be remote objects as well. OpenPegasus aims to support provider/server interoperatibility. This enables using providers from other CIM implementations with OpenPegasus CIM server, and using OpenPegasus providers with other CIM servers.

MOF files representing the CIM schema and class extensions can be compiled and loaded to the repository.

OpenPegasus provides a modular CIM environment what enables the extension with new services, providers and functionality.

In the following sections, we show some examples of OpenPegasus on AIX. Example 8-1 shows Pegasus CIM server from the AIX 5L V5.3 expansion package.

Example 8-1 smitty screen for OpenPegasus installation

```
Install Software

Ty+--------------------------------------------------------------------------+
Pr|                           SOFTWARE to install                            |
  |                                                                          |
  | Move cursor to desired item and press F7. Use arrow keys to scroll.      |
* |     ONE OR MORE items can be selected.                                   |
* | Press Enter AFTER making all selections.                                 |
  |                                                                          |
  | [MORE...112]                                                             |
  |                                                                          |
  | > sysmgt.pegasus.cimserver                                          ALL  |
  |    + 2.3.1.0  Pegasus CIM Server Runtime Environment                     |
  |                                                                          |
  | > sysmgt.pegasus.osbaseproviders                                    ALL  |
  |    + 1.2.3.0  Base Providers for AIX OS                                  |
  |                                                                          |
  |   sysmgt.websm.security                                             ALL  |
  | [MORE...13]                                                              |
  |                                                                          |
  | F1=Help                 F2=Refresh              F3=Cancel                |
F1| F7=Select               F8=Image                F10=Exit                 |
F5| Enter=Do                /=Find                  n=Find Next              |
F9+--------------------------------------------------------------------------+
```

After the installation, we can start the CIM server by running the command **cimserver**, as shown in Example 8-2.

Example 8-2 OpenPegasus base directory content and CIM server starting

```
csmserver:/opt/freeware/cimom/pegasus:> ls -l
total 40
-r-xr-xr-x   1 root     system        12759 Feb 20 2004  README
dr-xr-xr-x   2 root     system          256 Feb 20 2004  bin
dr-xr-xr-x   4 root     system          256 Sep 09 11:08 etc
dr-xr-xr-x   2 root     system         4096 Feb 20 2004  lib
drwxr-xr-x   2 root     system          256 Sep 08 08:22 logs
```

```
dr-xr-xr-x   3 root     system          256 Feb 20 2004  samples
.
.
.
csmserver:/opt/freeware/cimom/pegasus/samples/clients:> cimserver
Logs Directory = /opt/freeware/cimom/pegasus/logs
CIM Server 2.3.1
--- CMPI Provider Manager activated
Listening on HTTP port 5988.
Started.
```

In the directory /opt/freeware/cimom/pegasus/samples/clients are sample programs to try the functionality of OpenPegasus. See Example 8-3 for output when starting **osinfo**.

Example 8-3 Sample query programs for OpenPegasus

```
csmserver:/opt/freeware/cimom/pegasus/samples/clients:> ls
EnumerateInstanceNames  README wbemexec                wbemexec_ei.xml
EnumerateInstances      osinfo wbemexec_ai.xml
csmserver:/opt/freeware/cimom/pegasus/samples/clients:> ./osinfo
OperatingSystem Information
  Host: csmserver
  Name: AIX
   ( 5 3 )
  Version: 3
  UserLicense: Unknown
  Number of Users: 10 users
  Number of Processes: Unknown
  OSCapability: 64 bit
  LastBootTime: Unknown
  LocalDateTime: Sep 9, 2004  18:24:30 (-0600)
  SystemUpTime: Unknown
```

We tried to connect the OpenPegasus CIM server from the HMC using the CIMOM server installed there. See Example 8-4 for the output of the started query.

Example 8-4 Running cimomtest on HMC to query the CSM server

```
[hscroot@hmcitso hscroot]$ cimomtest clientEnum csmserver lniesz root/cimv2
IBMAIX_ComputerSystem
Password:

  *** Show Namespace, CIMClient and ObjectPath ***
 NameSpace: CIMNameSpace=root/cimv2 [host csmserver]
 CIMClient: org.snia.wbem.client.CIMClient@251b4ce3
 ObjectPath: IBMAIX_ComputerSystem
```

```
+++++
IBMAIX_ComputerSystem.CreationClassName="IBMAIX_ComputerSystem",Name="csmserver
"
+++++
instance of IBMAIX_ComputerSystem {
Caption = Computer System;
Description = A class derived from ComputerSystem that represents the single
node container of the AIX OS.;
ElementName = csmserver;
InstallDate = null;
OperationalStatus = null;
StatusDescriptions = null;
Status = null;
EnabledState = 2;
OtherEnabledState = null;
RequestedState = 2;
EnabledDefault = 2;
CreationClassName = IBMAIX_ComputerSystem;
Name = csmserver;
PrimaryOwnerName = root;
PrimaryOwnerContact = root@csmserver;
Roles = null;
NameFormat = IP;
OtherIdentifyingInfo = null;
IdentifyingDescriptions = null;
Dedicated = [0];
OtherDedicatedDescriptions = null;
ResetCapability = null;
PowerManagementCapabilities = null;
}

+++++
 !!! Close session !!!
```

For more information regarding OpenPegasus and CIM implementation on AIX see the *AIX Common Information Model Guide,* SC23-4942 and the OpenPegasus Web site at:

http://www.openpegasus.org

A good overview and more documentation about WBEM and OpenPegasus can be found on the Open Group Web site at:

http://www.opengroup.org

Search for the documents *Pegasus Technical Workshop - Tutorial - WBEM Overview*, and *Pegasus Technical Workshop Overview Presentation*s on the Open Group documentation Web site:

http://www.openpegasus.org/documents.tpl?CALLER=doc.tpl&grouped=Y

OpenCIMOM

The openCIMOM is a Red Hat Package Manager (RPM) package. This RPM contains the Storage Networking Industry Association (SNIA) openCIMOM suite.

OpenCIMOM is needed for hardware management functions initiated from the CSM server. It contacts the Hardware Management Console (HMC) when hardware control functions are initiated from the CSM server.

Example 8-5 shows the **cimomtest** output started on the HMC to query the pSeries systems connected.

Example 8-5 Running cimomtest on HMC to query the connected pSeries systems

```
[hscroot@hmcitso hscroot]$ cimomtest clientEnum hmcitso hscroot root/ibmhscV3_2
ibmhsc_computersystem|more
Password:

  *** Show Namespace, CIMClient and ObjectPath ***
 NameSpace: CIMNameSpace=root/ibmhscV3_2 [host hmcitso]
 CIMClient: org.snia.wbem.client.CIMClient@21faa9dc
 ObjectPath: ibmhsc_computersystem
IBMHSC_ComputerSystem
+++++
IBMHSC_ComputerSystem.Name="7040-681*022BE2A",CreationClassName="IBMHSC_Compute
rSystem"
+++++
instance of IBMHSC_ComputerSystem {
Name = 7040-681*022BE2A;
Mode = 255;
PowerOffPolicy = false;
CspSurveillancePolicy = 20;
State = 1;
Capability = 195;
AffinityCapability = 2;
RuntimeCapability = 24;
CUoDCapabilities = 0;
UserDefinedName = itso_p690;
SerialNumber = 022BE2A;
Model = 7040-681;
InstalledCPUCount = 8;
IODrawerCount = 5;
InstalledMemory = 16384;
OpPanelWindowCount = 0;
VirtualTTYWindowCount = 0;
OpPanelValue = LPAR...;
CSPVersion = V4.0;
```

```
LMBSize = 256;
LparMinMemorySize = 256;
LPARSlotCount = 16;
CUoDActivateStatus = 0;
CUoDResourceInitStatus = 0;
OnOffResourceInitStatus = 0;
AsyncMsg = null;
CageNumber = null;
InitialLoadInfo = null;
LastLoadInfo = null;
ResetCapability = null;
PowerManagementSupported = null;
PowerManagementCapabilities = null;
PowerState = null;
WakeUpType = null;
NameFormat = null;
OtherIdentifyingInfo = null;
IdentifyingDescriptions = null;
Dedicated = null;
CreationClassName = IBMHSC_ComputerSystem;
PrimaryOwnerContact = null;
PrimaryOwnerName = null;
Roles = null;
InstallDate = null;
Status = null;
Caption = null;
Description = null;
}

+++++
 !!! Close session !!!
```

Future opportunities

The SNIA Open Source Java CIMOM project has moved to the Open Group, where it will join with the Open Group's Pegasus project in creating a comprehensive, open-source, WBEM-based environment. The SNIA Web site can be found at:

http://www.snia.org

It is possible to create extensions for the present openCIMOM package installed on Cluster Systems Management (CSM) servers, to collect more information from the connected systems.

If in the future, the CIM providers on the HMC can change the hardware configuration, then the initiation of this hardware change could be done from the provisioning toolset. This can be based on the customers requirements and on

the discovered free resources, and will be started as a step of the provisioning workflow.

8.1.2 Web services

In a Service Oriented Architecture (SOA) all entities look as services, which can provide either computational, storage, business or other functions. These services are communicating with each other via standards based protocols exchanging well structured information.

The services capabilities are defined by interfaces, declaring the type of information they can request and provide, the protocols they can be reached, and functions they can do. The implementation of the service functionality is hidden from the client which initiate any action on them. This can be seen as another level of virtualization.

Web services uses a program-to-program communications model built on existing standards, such as Hyper Text Transmission Protocol (HTTP), Extensible Markup Language (XML), Simple Object Access Protocol (SOAP), Web Services Description Language (WSDL), and Universal Description Discovery, and Integration (UDDI).

The on demand operating environment can be seen as the layers of these services where the layers are built upon each other.

How can be the provisioning tools provided by the pSeries servers and by AIX, be integrated into this service oriented world? They have to provide interfaces to describe their functionality, and the way a client communicate with them. This description has to be based on standards used by other services. They have to understand the protocols used in an operating environment based on the service oriented architecture. In some case the development of these interfaces is done by the team who provide the base application or tool. In other cases the team which integrates the separate services into an automatic infrastructure has to create the interfaces used between the separated elements. To enable this up to date normalizing, deep knowledge is necessary about the service. The developer of the service has to publish the APIs for its product. Version control and following the changes can require a lot of effort.

8.1.3 GRID computing

GRID computing can be seen as the next level of IT resource virtualization and distributed groups of cluster systems. For the user or a client which can be an application or data it seems as a single huge computing system. It integrates computing, storage and networking resources.

Building on existing Web services standards, the Open Grid Services Architecture (OGSA) defines a GRID service as a Web service that conforms to a particular set of conventions. The specifications in the architecture are under continuous development by members of the Global Grid Forum:

http://www.ggf.org

For more information about OGSA visit the following IBM Web site:

http://www-106.ibm.com/developerworks/grid/library/gr-visual

A very important aspect of any GRID environment is how can the resource discovery and allocation be automated. In other words, how is the automated provisioning enabled in GRID computing. IBM @server pSeries servers and the operating system running on them can provide the physical resources and the logical resources as well by built in virtualization features.

At this time many steps of the resource allocation are automatized, however the base infrastructure has to be build in advance. Servers have to be installed with GRID enabling middleware, filesystems have to be created before they can receive a request from resource managers. Next step could be the automation of these steps. For example a resource pool manager could integrate a new pSeries server into a GRID, by creating shared processor LPARs and installing the operating system and middleware. The amount of time needed for this will not allow the prompt requests but could be good to serve scheduled ones.

Many tools used in GRID computing are developed as open software and distributed as a Red Hat Package Manager (RPM) set of files. The installation manager in AIX can handle these packages, so it is easy to install and use them on pSeries machines. General Parallel File System (GPFS) can be used for storage virtualization as a high performance and reliable distributed file system. For batch jobs and workload management, AIX and POWER5 systems provide LoadLeveler (LL), Workload Manager (WLM), and Partition Load Manager (PLM).

Globus Toolkit

Globus Toolkit 3.0 is an implementation of the Open Grid Services Infrastructure (OGSI) Version 1.0 which is a new specification that the Globus Project plays a key role in defining. More information about the Globus project can be found at:

http://www.globus.org

A good explanation on GRID computing, and the Globus Toolkit can be found in the *Introduction to Grid Computing with Globus,* SG24-6895.

8.2 Storage virtualization for provisioning

It is a trend that the size of a physical disk is growing rapidly year after year. Today the average application, and the data the application works on, consumes much more space on the permanent storage area that the operating system. Storage virtualization technologies can help in the resource allocation that enables optimal utilization of resources.

Keeping the operating system, the application code and the application data on separate disks can increase the level of availability and shorten the recovering time of the elements of an application complex. Here again, storage virtualization can help with separation of different kind of data and in site or either geographical mirroring.

The solutions chosen for storage virtualization can be separated based on the main requirement of which level we want to raise:

- **Performance**: Could be reached by choosing faster I/O paths and storage devices, raising the number of concurrent I/O operations.
- **Availability**: Could be reached by increasing the copies of the stored information, using multiple path to storage devices.
- **Scalability**: Could be reached by increasing the possible routes to storage devices, sharing the data over some network.

All the mentioned requirements already has solutions which are addressing not only one but maybe all the three. Redundant Array of Independent Disks (RAID), Storage Area Network (SAN), Network Accessed Storage (NAS), General Parallel File System (GPFS), Network File System (NFS) is widely used as storage for the pSeries servers. It is also possible to place the operating system on external storage devices.

With the new features of the POWER5 such as Advanced Virtualization, we can do even more fine tuning on the granularity and the utilization of our storage devices.

For provisioning, storage virtualization provides the possibility that we do not have to care where and how these storage devices are build. We just use the solution, based on the requirements. The environment has to be set up, of course, and since we want to lower the cost of ownership there is no need to implement maximum capacity from the beginning, which means the storage environment has to be scalable.

This can be handled by the provisioning toolset if the solution for storage virtualization provides registration and discovery functionality.

Registration

Virtualized storage providers has to be able to register themselves to a resource database. This feature enables the separation of the resource management from the provisioning process. As the free resource pool is already updated with the new available storage elements, no discovery is needed, so the provisioning process can be faster.

The provisioning toolset needs information about the connection path, the available space, the performance and data availability.

Discovery

The storage virtualization has to provide an interface towards the monitoring and provisioning toolset. This interface enables the provisioning toolset to run a discovery for the storage solutions required by the customer.

8.3 The role of RSCT in provisioning

Reliable Scalable Cluster Technology (RSCT) is a set of software components that together provide a comprehensive clustering environment for AIX and Linux.

The components of RSCT are:

- **Resource Monitoring and Control (RMC) subsystem**: It provides common view of the resources for individual systems or clusters of nodes.
- **Core resource managers**: This software layer provides the link between RMC and the commands of a resource. All resource managers interacts the same way with each other and with the higher layers in the RSCT. They show a standard interface to their clients but perform their actions in a resource-specific way. Each resource manager is implemented as a process in AIX and each can manage several resource classes. A resource class is a collection of resources with similar characteristic.
- **Topology Services**: Provides services for node and network failure detections on some cluster configurations.
- **Group Services**: Provides coordination of cluster events between nodes of some type of cluster configuration.

For more details on RSCT components see RSCT manuals and Redbooks:

- *IBM Reliable Scalable Cluster Technology Technical Reference,* SA22-7890
- *IBM Reliable Scalable Cluster Technology Administration Guide,* SA22-7889
- *A Practical Guide for Resource Monitoring and Control,* SG24-6615

8.3.1 Resource managers for provisioning

At the moment there are two core resource managers which can be used or are used for provisioning purposes: CIM resource manager, hardware control resource manager.

CIM resource manager

One of the core resource managers in RSCT implementation is available only on Linux at this time. This is the CIM resource manager which provides the connection between RSCT and CIM. It is possible to register and query CIM classes with RMC.

Hardware control resource manager

The IBM.HWCTRLRM resource manager running on CSM is responsible for the hardware control of the defined CSM nodes. It controls the following RMC resource classes:

- IBM.NodeHwCtrl node hardware control
- IBM.HwCtrlPoint hardware control point
- IBM.DeviceHwCtrl hardware device
- IBM.DeviceGroup hardware device group

Example 8-6 shows information about the running hardware control resource manager, which is under the control of the system resource controller daemon.

Example 8-6 lssrc -ls IBM.HWCTRLRM

```
csmserver:/:> lssrc -ls IBM.HWCTRLRM
Subsystem         : IBM.HWCTRLRM
PID               : 348334
Cluster Name      : IW
Node Number       : 1
Daemon start time : Wed Sep  8 13:28:20 CDT 2004

Information from malloc about memory use:
   Total Space    : 0x00720280 (7471744)
   Allocated Space: 0x006777d8 (6780888)
   Unused Space   : 0x000a6000 (679936)
   Freeable Space : 0x00000000 (0)
```

This resource manager is using the openCIMOM package to communicate the CIM server running on the HMC. The CIM server on HMC does the work by utilizing the local providers on HMC to manage the connected pSeries systems.

8.3.2 Extending RSCT

A possible direction towards automated provisioning could be to extend some of the present functionality of RSCT and create new resource managers.

Hardware management

At the present, we have hardware control only, via the HMC. We cannot create LPARs, as a step of a new node installation, initiated from the management server (MS). The CIM providers on HMC should be extended to receive the LPAR management information from the resource manager, and start LPAR or dynamic LPAR activity. This operation has to take into account the running applications and services, if the amended LPAR is already in production. As the MS is the central focal point of domain management, this can be solved as it is already managing the node.

The direction of the development of HMC and RSCT (which is part of AIX now) however shows further separation of the hardware, software control, and management. For a centralized provisioning environment, there should be a possibility to connect the tools we have now, or integrate them in an automated workflow. This integrated solution will be represented in a higher layer of the on demand environment. An existing solution is Tivoli Provisioning Manager.

Software installation and management

For software installation and version control, AIX has a built in feature, the Network Installation Manager (NIM). This is a continuously evolving environment with new possibilities in every new release of AIX.

NIM can be configured to automatically install AIX on a node or pre configured LPAR and run customizing programs after the install. It is possible to define network interfaces which will be managed by NIM. HA NIM is a new feature to keep the installation environment highly available, by moving the NIM functionality to a standby machine in the case of failure on the primary NIM server.

CSM is using NIM to install the managed nodes as well, but leaving the possibility that we configure the NIM for other machines or LPARs. At the time of writing this book, the scripts are working only for NIM master which is on the same machine as the CSM server.

To have the same monitoring and management interface as the clustering and the hardware control resource managers it would be possible to create a NIM resource manager.

Note: IBM plans to externalize the APIs for Resource Monitoring and Control and the resource managers.

A

CPU resource distribution by Hypervisor and PLM

In this appendix, we discuss in detail, the distribution of CPU cycles by POWER Hypervisor (PHYP) and Partition Load Manager (PLM). PHYP is the firmware layer that runs underneath AIX, Linux and i5/OS on p5 processors. Its main function in this context is to schedule the partitions' virtual CPUs on the physical CPUs of the machine. A description of PLM is given in 4.2.4, "Partition Load Manager (PLM)" on page 41.

This appendix contains the following topics:

A.1 Entitlement in POWER Hypervisor

When a partition using Micro-Partitioning technology is defined by the administrator, it is given a desired number of processing units - effectively CPU cycles. These are defined in 1/100ths of a CPU. Assuming there are sufficient unallocated CPU cycles at boot time, the desired units will become the “entitlement” of the LPAR. An LPAR’s entitlement is always available to it if required, irrespective of the load on the system. Unallocated CPU cycles are those which are not part of the entitlement of another LPAR.

If there are fewer unallocated units available at boot time, an LPAR can be given a lower entitlement. However, an LPAR is also defined with a minimum requirement. If there are not enough unallocated units to meet the LPAR’s minimum requirement, the LPAR will not boot.

Micro-Partitioning technology can be defined as capped. Capped Micro-Partitioning technology can only use its entitlement.

A partition using Micro-Partitioning technology can use less than its entitlement, donating the additional cycles back to PHYP. PHYP may therefore have more cycles available (free) than the sum of the LPAR entitlements would suggest. However, these available but allocated cycles cannot be used to get an LPAR to boot.

PHYP will give additional cycles to uncapped Micro-Partitioning technology if those cycles are available and the Micro-Partitioning technology can use them. In this case, the LPAR’s usage will be higher than its entitlement. The additional cycles are known as the “excess”.

The separation of entitlement and excess is critical to understanding PHYP and PLM. PHYP does not change the entitlement of an LPAR - it just controls the number of cycles an LPAR can use. However, an LPAR’s entitlement can be changed by a Dynamic Logical Partitioning operation, which can be initiated by an administrator or by PLM.

The definition of Micro-Partitioning technology also includes a maximum number of processing units. This is only used as a limit on the range of dynamic LPAR operations, not on the excess. For example, a partition with a current entitlement of 1.5 CPUs and maximum entitlement of 2.0 CPUs can have its entitlement increased by a dynamic LPAR operation to 2.0 CPUs and no further. However, the Micro-Partitioning technology could use all the CPU cycles on the system if those cycles are available. The minimum processing units defined for the partition also sets a limit on dynamic LPAR operations.

A.2 Distribution of the excess

As previously stated, PHYP can give uncapped Micro-Partitioning technology additional CPU cycles if there are any available. It determines which Micro-Partitioning technology should receive the additional cycles by looking at the weight setting, which is part of the Micro-Partitioning technology definition. The weight setting is a number between 0 and 255 and the default is 128. If Micro-Partitioning technology is given weight of 0 it is treated as a capped partition and can use no excess cycles.

Somewhat confusingly, other Redbooks describe how weight works in two ways, although they are mathematically equivalent. For completeness, both descriptions are given here.

If only one LPAR can use additional capacity, and there are CPU cycles free, then that LPAR will be given those resources. The simplest interesting case is where there are two competing LPARs.

A.2.1 Description 1

A partition's priority is determined by the excess cycles it is using divided by its weight. The lower this number is, the higher its priority. Higher priority LPARs can take resources from lower priority LPARs.

For example, a system has 6 CPUs and two LPARs, A and B:

- LPAR A: entitlement 1.0 CPUs, weight 10, uncapped
- LPAR B: entitlement 2.0 CPUs, weight 20, uncapped

If both partitions are now loaded with a CPU-intensive application, so they use as many cycles as they can, PHYP will allocate the excess capacity of the system (3.0 CPUs) such that the priority of the LPARs is equal.

- Partition A will be given excess E, and its priority P will equal E/10.
- Partition B will be given excess F, and its priority P will equal F/20.

Since the system should allocate the resources such that P is the same for A and B, E/10=F/20. Multiply both sides by 20 and we have F=2E.

The total excess, E+F, equals 3. Substituting 2E for F, we get E+2E=3.

Each LPAR will use its entitlement plus its excess, so:

- LPAR A will use 2.0 CPUs (1.0 entitlement + 1.0 excess)
- LPAR B will use 4.0 CPUs (2.0 entitlement + 2.0 excess)

A.2.2 Description 2

A partition's share of the excess is computed by dividing its variable capacity weight by the sum of the variable capacity weights for all uncapped partitions.

Taking our two LPARs again, the total number of shares (weights) is 10+20=30. Total excess is 3, so each share is 3/30=0.1, and we end up with the same distribution.

A.3 Entitlement in PLM

In PLM, each Micro-Partitioning technology is again given a minimum and maximum entitlement, plus a guaranteed entitlement and a delta value.

PLM uses dynamic LPAR operations to change the entitlement of partitions depending on the load within those partitions. These dynamic LPAR operations are limited by the minimum and maximum settings within PLM. Thus PLM will not reduce a partition’s entitlement below its minimum or above its maximum. If the minimum entitlement in PHYP is set above that in PLM, the PHYP value will be used. Similarly, if the PHYP maximum is lower than that in PLM, the PHYP value will be used. PHYP maxima and minima cannot be dynamically changed, whereas those in PLM can be, so you may want to configure PHYP with broad values and use PLM to set narrower limits.

The PLM delta value is the percentage of the current entitlement to add or remove in each dynamic LPAR operation, and is typically 10%. If adding 10% of current capacity will take an LPAR above its maximum, no dynamic LPAR operation will take place - it will not add a lower percentage. The same applies to removing capacity below the minimum.

The guaranteed entitlement of an LPAR is precisely that - the LPAR will always be able to have this entitlement if it needs it, irrespective of the load on the system and the PLM shares discussed below. The PHYP entitlement is a point in time guarantee of CPU cycles. The PLM guaranteed entitlement is a long-term guarantee of entitlement that then determines CPU cycles. In both cases, the LPAR can only go below the entitlement “voluntarily” by not using the resources. It cannot have PHYP CPU cycles or PLM entitlement taken away by another LPAR.

If the Micro-Partitioning technology is configured in PHYP with more desired processing units than are subsequently configured as PLM guaranteed units, the PLM guaranteed units will be increased to the PHYP desired value.

PLM makes dynamic LPAR changes when a partition reaches a trigger point a certain number of times. Both the trigger point and the number of times are

configured within PLM. If a partition is, for example, 95% CPU busy for six intervals (of ten seconds), it will generate an RMC request for more resources. If those resources are available, PLM will issue a dynamic LPAR request to increase that partition's entitlement by the delta value.

Partitions that use fewer cycles than a minimum trigger point for a period of time will also generate an RMC request. PLM can be configured to reduce the partition's entitlement immediately, or only if another partition requests the resources.

A.4 Resource distribution in PLM

Each LPAR is configured with PLM shares. These are analogous to the PHYP weight and are calculated in the same way. However, the results (in terms of usage) are different because PLM shares affect entitlement, not the excess.

In the scenario where two LPARs are both running CPU-intensive loads, both partitions will request additional resources. PLM then allocates those resources in accordance with the share settings. Returning to our example:

Our system has 6 CPUs and two LPARs, A and B:

- LPAR A: guaranteed entitlement 1.0 CPUs, shares 10, uncapped
- LPAR B: guaranteed entitlement 2.0 CPUs, shares 20, uncapped

We have 30 shares (of 0.2 CPUs each) and 6.0 CPUs. In an ideal situation where both partitions acquire the resources they are entitled to, we will eventually reach a point where LPAR A has (10/30)*0.2=2.0 CPUs and LPAR B has (20/30)*0.2=4.0 CPUs. All of this is entitlement and there will be no excess capacity remaining. In practice, the size of the delta values will often mean that the "final" dynamic LPAR operation would take the partitions above their share entitlements, so some excess remains.

PLM does not control the distribution of the excess directly. However, if only one LPAR is loaded and the other is idle, the loaded LPAR will reach its maximum entitlement. If we examine the PHYP weights of the partitions at this point, we find that the loaded LPAR has weight 0, so it is effectively capped at its full entitlement. An idle LPAR (with entitlement between its minimum and its guaranteed) has weight 255, and can therefore use a high proportion of the excess if its load increases. Note that the loaded partition can use fewer cycles than it could if PLM was not running, because it cannot use all the excess.

In our example above, the actual usage of the two partitions was the same with PHYP and PLM. This is because the shares are multiples of the guaranteed

entitlement. If we look at an example where this is not the case, we can see the differences between PHYP and PLM.

Our system has 6 CPUs and two LPARs, A and B:

- LPAR A: guaranteed entitlement 1.0 CPUs, PHYP weight 20, PLM shares 20, uncapped
- LPAR B: guaranteed entitlement 2.0 CPUs, PHYP weight 10, PLM shares 10, uncapped

We might use this configuration if A has a variable but important workload, while B has a normally stable but less important workload.

If both partitions are loaded, we will see the resource distribution shown in Figure A-1.

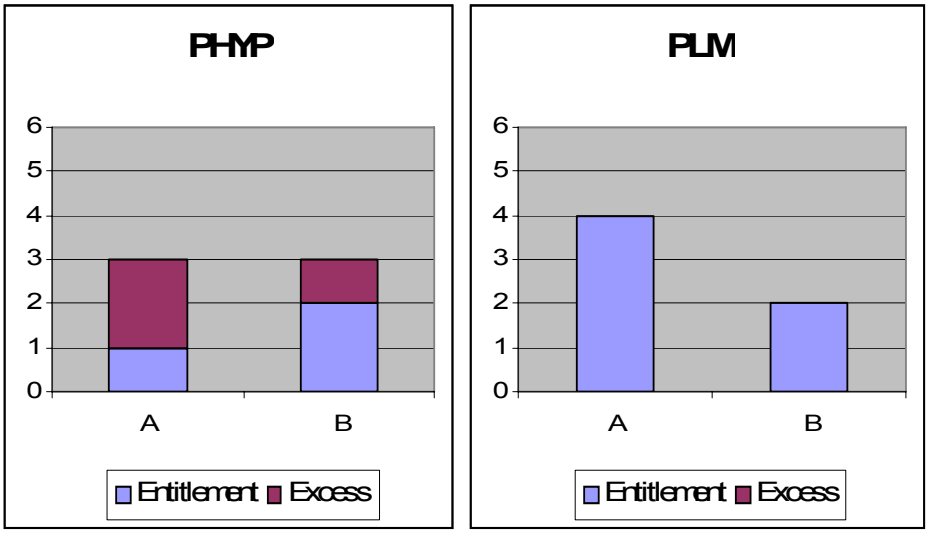


Figure A-1 Resource distribution by PHYP and PLM

Because these are changes in time, it is possible that PLM could give LPAR B 3.0 CPUs entitlement while it is the only loaded partition. If LPAR A is loaded later, PLM should take entitlement from B and give it to A, as long as the dynamic LPAR remove operation (from B) completes.

Batch jobs have the CPU-intensive characteristics that could drive this long-term increase in entitlement, particularly if there are no other active LPARs on the system. You may therefore want to limit systems that run such jobs using maximum entitlement or a relatively low number of shares. Bear in mind that if

such a partition increases its entitlement enough, it could reduce the unallocated cycles sufficiently to stop another LPAR being booted.

PLM manages memory in the same way as CPUs. Each partition has shares for memory (which are separate from those for CPUs). There is no concept of “excess” for memory because it cannot be given to partitions on a point in time basis.

Dedicated processor partitions can also be managed by PLM, but the delta values will have to be larger (1.0 CPUs). The trigger points should be set accordingly.

Abbreviations and acronyms

AIX	Advanced interactive Executive
APAR	Authorized Problem Analysis Report
API	Application Programming Interface
ARM	Application Response Measurement
AS	Advanced Server
ASCII	American Standard Code for Information Interchange
ASMI	Advanced System Management Interface
BFF	Backup File Format
BOS	Basic operating system
CD	compact disk
CD-ROM	compact disk - read only media
CFM	Configuration File Manager
CHRP	Common Hardware Reference Platform
CIM	Common Information Model
CIMOM	Common Information Model Object Manager
CMPI	Common Manageability Programming Interface
CoD	Capacity on Demand
CPU	central processing unit
CSM	Cluster Systems Management
C-SPOC	Cluster Single Point of Control
CUoD	Capacity Upgrade on Demand
DCE	Distributed Computing Environment
DCEM	Distributed Command Execution Manager
DCM	Data Center Model
DMTF	Desktop Management Task Force
DR	dynamic reconfiguration
DRM	Dynamic Reconfiguration Manager
EIM	Enterprise Identity Mapping
EJB	Enterprise Java Beans
eWLM	Enterprise Workload Manager
FC	Feature Code
Gb	Gigabit
GHz	Gigahertz
GPFS	General Parallel File System
GUI	Graphical user interface
HA	High Availability
HACMP	High Availability Cluster Multiprocessing
HACMP/XD	High Availability Cluster Multiprocessing eXtended Distance
HAGEO	High Availability Geographic Cluster for AIX
HBA	Host Bus Adapter
HMC	Hardware Management Console
HPC	High Performance Computing
HTML	Hyper-text Markup Language
HTTP	Hyper Text Transmission Protocol
I/O	input/output
i5/OS	The iSeries operating system on p5

IBM	International Business Machines Corporation
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
IPAT	IP Address Takeover
IPL	Initial Program Load
IT	Information Technology
ITITO	IBM Tivoli Intelligent ThinkDynamic Orchestrator
ITPM	IBM Tivoli Provisioning Manager
ITSO	International Technical Support Organization
JFS	Journaled File System
JFS2	Journaled File System 2
LAN	local area network
LDAP	Lightweight Directory Access Protocol
LL	LoadLeveler
LMB	Logical Memory Block
LPAR	Logical Partitioning / Logical Partition
LPP	Licensed Program Product
LTPA	Lightweight Third-Party Authentication
LUN	logical unit
LV	logical volume
LVM	Logical Volume Manager
M/T	Machine Type
MB	megabyte
MOF	Managed Object Format
MPIO	Multi Path I/O
MS	management server
N/A	not applicable
NAS	Network Accessed Storage
NFS	Network File System
NIM	Network Installation Manager
NTP	Network Time Protocol
ODM	Object Data Manager
OGSA	Open Grid Services Architecture
OGSI	Open Grid Services Infrastructure
OS	operating system
p5	POWER 5 (processors)
PC	personal computer
PCI	peripheral component interconnect
PHYP	POWER Hypervisor
PLM	Partition Load Manager
POWER	Performance Optimization with Enhanced RISC
PSSP	Parallel System Support Programs
PVID	Physical Volume Identifier
RAID	Redundant Array of Independent Disks
RAS	reliability, availability, and serviceability
RDMA	Logical Remote Direct Memory Access
RISC	reduced instruction-set computer
RMC	Resource Monitoring and Control
RPM	Red Hat Package Manager
RSCT	Reliable Scalable Cluster Technology
SAN	Storage Area Network
SAP	service access point
SCSI	small computer system interface
SDD	Subsystem Device Driver
SEA	Shared Ethernet Adapter
SLA	Service Level Agreement

SMIT	System Management Interface Tool
SMP	symmetric multiprocessing
SMS	System Management Services
SNIA	Storage Networking Industry Association
SOA	Service Oriented Architecture
SOAP	Simple Object Access Protocol
SP	See PSSP
SPLPAR	Shared Processor Logical Partition
SPOT	shared product object tree
SSH	Secure Shell
SSL	Secure Sockets Layer
SUMA	Service Update Management Assistant
TB	Tie Breaker
TCB	Trusted Computing Base
TCO	total cost of ownership
TCP	Transmission Control Protocol
TSA	Tivoli System Automation Server
UDDI	Universal Description Discovery, and Integration
URL	Universal Resource Locator
VE	Virtual Engine
VEC	Virtual Engine Console
VIO	Virtual I/O
VLAN	Virtual LAN
VP	Virtual Processor
VSCSI	Virtual SCSI
WBEM	Web Based Enterprise Management
WLM	Workload Manager
WSDL	Web Services Description Language
WSM	Web-based Systems Manager
XML	Extensible Markup Language

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks” on page 188. Note that some of the documents referenced here may be available in softcopy only.

- *AIX 5L Differences Guide, Version 5.2 Edition,* SG24-5765-02
- *AIX 5L Differences Guide, Version 5.3 Edition,* SG24-7463-00
- *The Complete Partitioning Guide for IBM @server pSeries Servers,* SG24-7039-01
- *IBM @server pSeries 670 and pSeries 690 System Handbook,* SG24-7040-02
- *AIX 5L Workload Manager (WLM),* SG24-5977-01
- *Enterprise Workload Manager,* SG24-6350-00
- *A Practical Guide for Resource Monitoring and Control,* SG24-6615-00
- *Advanced POWER Virtualization on IBM eServer p5 Servers: Introduction and Basic Configuration*, SG24-7940-00
- *Advanced POWER Virtualization on IBM @server p5 Servers Architecture and Performance Considerations*, SG24-5768-00
- *Introduction to Grid Computing with Globus,* SG24-6895-01
- *Virtualization and the On Demand Business,* REDP-9115-00
- *Exploring Storage Management Efficiencies and Provisioning,* SG24-6373-00
- *Provisioning On Demand Introducing IBM Tivoli Intelligent ThinkDynamic Orchestrator,* SG24-8888-00
- *Server Consolidation on IBM @server pSeries Systems*, SG24-6966-00
- *Server Consolidation on RS/6000*, SG24-5507-00

Other publications

These publications are also relevant as further information sources:

- *AIX Common Information Model Guide,* SC23-4942-00
- *AIX Installation in a Partitioned Environment,* SC23-4382-04
- *IBM Cluster Systems Management for AIX 5L, Administration Guide, Version 1.4,* SA22-7918-07
- *IBM Cluster Systems Management for AIX 5L, Command and Technical Reference, Version 1.4,* SA22-7934-04
- *IBM Cluster Systems Management for AIX 5L, Planning and Installation Guide, Version 1.4,* SA22-7919-07
- *HACMP Administration and Troubleshooting Guide, Version 5.2,* SC23-4862-03
- *HACMP Concepts and Facilities Guide, Version 5.2,* SC23-4864-03
- *HACMP Planning and Installation Guide, Version 5.2,* SC23-4861-03
- *HACMP and New Technologies for Availability,* white paper, June 2004; PDF available from:

  http://www-1.ibm.com/servers/eserver/pseries/software/whitepapers/hacmp_newtech.html
- *Installation and Support Recommendations for Successful High Availability Environments using IBM HACMP V5.1 for AIX 5L*, white paper, July 2004; PDF available from:

  http://www-1.ibm.com/servers/eserver/pseries/software/whitepapers/hacmp_installsupport.html
- *IBM @server Hardware Management Console for pSeries Installation and Operations Guide,* SA38-0590-08
- *IBM @server Hardware Management Console for pSeries Maintenance Guide,* SA38-0603-05
- *IBM Reliable Scalable Cluster Technology Administration Guide,* SA22-7889-05
- *IBM Reliable Scalable Cluster Technology Managing Shared Disks,* SA22-7937-01
- *IBM Reliable Scalable Cluster Technology Technical Reference,* SA22-7890-06
- *Tivoli Provisioning Manager Install Guide,* GC32-1615-00

- *Pegasus Technical Workshop - Tutorial - WBEM Overview*

  http://www.openpegasus.org

- *Pegasus Technical Workshop Overview Presentation*

  http://www.openpegasus.org

Online resources

These Web sites and URLs are also relevant as further information sources:

- IBM AIX Library

  http://www-1.ibm.com/servers/aix/library/

- IBM eServer Information Center

  http://publib.boulder.ibm.com/eserver/

- Hardware Management Console (HMC)

  http://techsupport.services.ibm.com/server/hmc/home.html
  http://www.redbooks.ibm.com/abstracts/tips0280.html?Open

- Capacity on Demand

  http://www-1.ibm.com/servers/eserver/pseries/ondemand/cod/

- Dynamic LPAR

  http://www.redbooks.ibm.com/abstracts/tips0121.html?Open
  http://www.alphaworks.ibm.com/tech/dlpar

- Tivoli Software Information Center

  http://publib.boulder.ibm.com/tividd/td/tdprodlist.html

- IBM On Demand Automation Catalog

  http://www-18.lotus.com/wps/portal/automation

- Open Pegasus

  http://www.openpegasus.org

- AIX Toolbox for Linux Applications home page

  http://www.ibm.com/servers/aix/products/aixos/linux/

- Open Group

  http://www.opengroup.org

- Open Group Documentation Web Site

  http://www.openpegasus.org/documents.tpl?CALLER=doc.tpl&grouped=Y

- OpenSSH homepage

  http://www.openssh.or

- RPM Packages Link

  http://www-1.ibm.com/servers/aix/products/aixos/linux/download.html

- Global GRID Forum

  http://www.ggf.org

- A visual tour of Open Grid Services Architecture

  http://www-106.ibm.com/developerworks/grid/library/gr-visual

- Globus Project

  http://www.globus.org

- Cluster Systems Management

  http://www.redbooks.ibm.com/abstracts/tips0278.html?Open
  http://www.redbooks.ibm.com/abstracts/tips0295.html?Open

- IBM Virtualization Engine

  http://www-1.ibm.com/servers/eserver/about/virtualization/

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

E

F

G

H

I

L

M

N

W

IBM Redbooks

Introduction to pSeries Provisioning

(0.2”spine)
0.17”<->0.473”
90<->249 pages

Introduction to pSeries Provisioning

Redbooks

Overview of pSeries tools for on demand provisioning

pSeries automation recommendations

Practical examples using CSM and NIM on POWER5

pSeries provisioning is a term for effectively enabling automation tools to allocate, de-allocate, and re-allocate resources for users, applications, or even functionality within an application, thereby providing the most cost-effective delivery of computing resources to an organization. Provisioning can be a manual process, but there is a point where automation becomes essential. Although automation and integration of the provisioning process is a custom effort for each company, it can involve the scripted provisioning tools and automation becomes more dynamic, and the environment becomes more complex for IT professional. By using provisioning, you can provide users with an environment where resources are dynamically adjusted, given the demands of the business.

This IBM Redbook summarizes the pSeries provisioning concept. It highlights the IBM on demand business concepts, IBM Tivoli Provisioning Manager and pSeries automation workflows, pSeries provisioning tools and sample scenarios. It also describes how pSeries provisioning is positioned in an on demand world.

This book describes pSeries provisioning components such as Network Installation Manager (for client installation), RMC (for monitoring), dynamic LPAR (for resource allocation, de-allocation, and re-allocation), HACMP (for high availability), AIX 5L (for accounting), CSM (for cluster management), and CUoD (for automated capacity on demand upgrade). In combination with IBM Tivoli Provisioning Manager, they provide tools and workflows for provisioning resources.

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

SG24-6389-00 ISBN 073849089X