

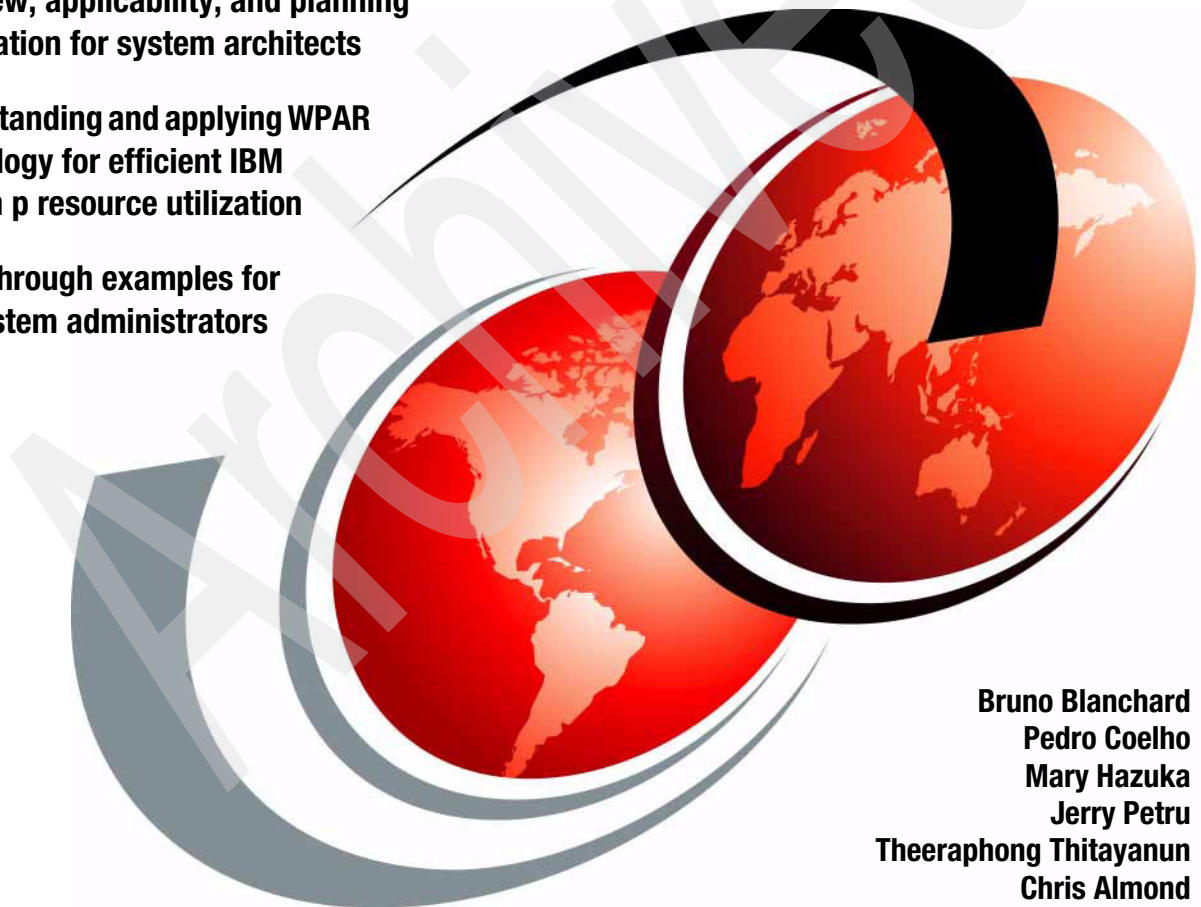


Introduction to Workload Partition Management in IBM AIX Version 6.1

Overview, applicability, and planning information for system architects

Understanding and applying WPAR technology for efficient IBM System p resource utilization

Walk-through examples for AIX system administrators



Bruno Blanchard
Pedro Coelho
Mary Hazuka
Jerry Petru
Theeraphong Thitayanun
Chris Almond



International Technical Support Organization

**Introduction to Workload Partition Management in
IBM AIX Version 6.1**

November 2007

Archived

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

First Edition (November 2007)

This edition applies to IBM AIX Version 6.1.

Note: This book is based on a pre-GA version of a product and may not apply when the product becomes generally available. We recommend that you consult the product documentation or follow-on versions of this IBM Redbooks publication for more current information.

© Copyright International Business Machines Corporation 2007. All rights reserved.

Note to U.S. Government Users Restricted Rights -- Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

Notices	ix
Trademarks	x
Preface	xi
The team that wrote this book	xii
Acknowledgements	xiii
Become a published author	xiv
Comments welcome	xv
Part 1. Introduction and planning	1
Chapter 1. Introduction to workload partition technology in AIX V6	3
1.1 Overview of partitioning and virtualization capabilities prior to AIX V6	4
1.2 AIX V6 and WPAR-based system virtualization	7
1.3 What is a workload partition	8
1.3.1 The global environment in an LPAR	10
1.3.2 WPARs	10
1.3.3 System WPARs	11
1.3.4 Application WPARs	11
1.4 Live application mobility	11
1.5 When to use workload partitions	13
1.5.1 Improvement of service level agreements	13
1.5.2 Protection of existing hardware investment	14
1.5.3 Optimization of resource usage	14
1.5.4 Highly granular control of resource allocation	15
1.5.5 Control of security and privilege command	15
1.5.6 Simplified handling of software stack	16
1.5.7 Simplified handling of application OS environment	17
1.5.8 Business continuity: Disaster or failure recovery solution	18
1.5.9 Supporting “Green” computing strategies	18
Chapter 2. Understanding and planning for WPARs	19
2.1 High-level planning information	20
2.2 General considerations	21
2.2.1 Networking	21
2.2.2 Deployment of the Workload Partition Manager	21
2.2.3 Software prerequisites	24
2.2.4 File system considerations	25

2.3	Global environment considerations	26
2.4	Application WPARs	27
2.5	System WPARs	28
2.6	WPAR mobility	31
2.7	WPAR and LPAR comparison	36
Part 2. Using WPARs: management and operation		39
Chapter 3. Overview of WPAR management tools		41
3.1	Comparison of application and system WPARs	42
3.2	Overview of tools	44
3.3	AIX command-line interface	45
3.4	WPAR Manager interface	48
3.5	Workload Manager (WLM) considerations	49
3.6	WPAR description database	50
3.6.1	WPAR representation	50
3.6.2	WPAR definition	50
3.6.3	WPAR specification	52
3.6.4	WPAR checkpoint state	53
3.6.5	Other global environment considerations	54
Chapter 4. State management		55
4.1	Overview of WPAR states	56
4.2	State transitions	57
Chapter 5. System administration tasks		61
5.1	Workload partition administration	62
5.1.1	Create a simple WPAR	62
5.1.2	Start, stop, and reboot a system WPAR	66
5.1.3	Change a WPAR	69
5.1.4	Creating WPARs: Advanced options	74
5.1.5	Recover from a broken state	84
5.1.6	Remove a WPAR	84
5.2	Software installation and maintenance	86
5.2.1	Software availability	86
5.2.2	Using installp	88
5.2.3	Using NIM	94
5.2.4	Updating software	94
5.2.5	Alternate disk installation	95
5.3	File system considerations	95
5.3.1	Creating a file system for a running WPAR	96
5.3.2	Changing a file system	97
5.3.3	Shared /usr with writable file system	98
5.3.4	Enabling EFS in WPAR	100

5.3.5 Network File System (NFS) considerations	102
5.4 Backup, restore, and cloning	102
5.4.1 Backup	102
5.4.2 Restore	103
5.4.3 Cloning	106
5.5 Users and groups	111
5.5.1 Default access: Users and root	111
5.5.2 Recover root password	112
5.5.3 Role-based access control (RBAC)	113
5.6 Print spooling considerations	115
5.7 System environments	116
5.8 Processes and subsystems	117
5.9 Performance and resource schedule	120
Chapter 6. IBM WPAR Manager for AIX	125
6.1 Overview of WPAR Manager for AIX	126
6.2 WPAR monitoring	127
6.3 WPAR relocation	128
6.3.1 Various WPAR states during relocation	130
6.3.2 WPAR relocation flow	130
6.4 WPAR load balancing	131
6.4.1 WPAR group	131
6.4.2 Server group	132
6.4.3 WPAR Manager workload balancer component	132
6.4.4 WPAR relocation policy	132
6.5 WPAR Manager component overview	135
6.5.1 Common agent services (CAS) framework	136
6.5.2 Common agent	137
6.5.3 Agent manager	138
6.5.4 Resource manager	139
6.6 WPAR Manager installation overview	140
6.7 WPAR Manager installation and configuration	141
6.7.1 Check prerequisites and prepare for installation	142
6.7.2 Install WPAR Manager and CAS Agent Manager	142
6.7.3 Install and configure DB2	146
6.7.4 Configure WPAR Manager and CAS Agent Manager	148
6.7.5 Verify WPAR Manager installation	153
6.7.6 Uninstall WPAR Manager and CAS Agent Manager	154
6.7.7 Uninstall DB2	155
6.8 WPAR Agent installation and configuration	155
6.8.1 Check prerequisites	155
6.8.2 Install WPAR Agent	155
6.8.3 Configure WPAR Agent	160

6.8.4	Verify WPAR Agent installation	161
6.8.5	Uninstall WPAR Agent	162
6.9	Creating a System WPAR	162
6.10	Creating an application WPAR	181
6.11	Migrating an application WPAR	192
6.11.1	Create a WPAR group	202
6.11.2	Resource Views	218
6.11.3	Deploy application WPAR	219
6.11.4	Task Activity	227
6.11.5	Application configuration	227
6.11.6	Logging and tracing	228
6.12	Introduction to mobility command-line interface	230
6.13	Explanation of mobility CLI commands	231
6.14	Comparison of CLI commands	232
6.15	Step-by-step relocation of a workload partition	233
Chapter 7.	Resource control	237
7.1	Workload partition resource control overview	238
7.1.1	CPU and memory allocation to a WPAR	238
7.1.2	Processes and threads in a WPAR	239
7.1.3	Virtual memory of a single process within a WPAR	239
7.1.4	Resource sets	239
7.2	Workload partition resource control attributes	240
7.3	Default values of resource control attributes	241
7.4	Practical examples of resource allocation	242
7.4.1	CPU	243
7.4.2	Memory	246
7.4.3	Processes and threads	247
7.4.4	Process virtual memory	248
7.5	Use case scenario	249
7.6	WPAR resource control changes to WLM	249
7.6.1	Number of user-defined superclasses	250
7.6.2	Resource limit granularity	250
7.6.3	Changes to WLM commands	250
7.7	Questions about WLM and WPAR resource control	251
7.7.1	The tier	251
7.7.2	Superclass and subclass	251
7.7.3	Class assignment rule	251
7.7.4	Enabling resource control when creating a WPAR	252
7.7.5	Enabling resource control for selected WPARs	252
7.7.6	Using WLM commands directly for resource control	253
7.7.7	Using WLM to manage other workloads	253

Chapter 8. Device management	255
8.1 Device visibility in the workload partition	256
8.2 Supported device categories in a WPAR environments	259
Part 3. Appendixes	261
Appendix A. Modified AIX commands	263
AIX modified commands	264
Related publications	269
IBM Redbooks	269
Other publications	269
Online resources	269
How to get IBM Redbooks publications	270
Help from IBM	270
Index	271

Archived

Notices

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

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

Trademarks

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

AIX 5L™	POWER™	Redbooks (logo)  ®
AIX®	POWER4™	Sequent®
BladeCenter®	POWER5™	System p™
DB2 Universal Database™	POWER5+™	Tivoli®
DB2®	POWER6™	WebSphere®
HACMP™	pSeries®	Workload Partitions Manager™
IBM®	Redbooks®	

The following terms are trademarks of other companies:

Adobe, and Portable Document Format (PDF) are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

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

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

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

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

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

Preface

This IBM® Redbooks® publication presents workload partitions (WPARs), a set of completely new features introduced in IBM AIX Version 6.1.

Adding WPARs to the AIX® operating system provides a new level of system virtualization capability. WPARs complement the already existing AIX and System p™ virtualization features, by allowing:

- ▶ The partitioning of an AIX instance into multiple environments, each of them hosting applications and providing isolation from applications executing in the other environments
- ▶ The ability to checkpoint and restart the execution of applications without modification of the application code
- ▶ The capability to relocate a live application into a different AIX instance, whether it is hosted in a logical partition (LPAR) of the same physical server or executing on a different physical server
- ▶ Highly granulated control of the resource utilization by each application executing within an AIX instance

This book is written for multiple audiences. You can read only the sections that fit your needs, without having to read the whole book, cover to cover:

- ▶ Part 1 presents the high level concepts and architecture related to WPARs. The goal in Part 1 is to explain the benefits of using WPAR in different business contexts. This part is intended to be read by Solution Architects and Planners.
- ▶ Part 2 explains the technology supporting WPARs. It details the various components of the WPAR technology, and how they relate to other AIX concepts. Part 2 also presents examples of scenarios that take advantage of WPARs in conjunction with other AIX and pSeries features. This part is intended to be read by the designers of IT solutions, who need to include this technology in their IT environments, and help system administrators looking for walk-through examples of WPARs deployments.

Note: This book refers to the first version of functions and features related to WPARs in AIX V6. WPAR is a strategic component of the AIX architecture that will evolve and improve with following versions of AIX. Although this book presents a great number of new features and functions, you can consider it as only a preview of a larger set of features that will be announced later.

The team that wrote this book

This IBM Redbooks publication was produced by a team of specialists that came from around the world to work together at the IBM International Technical Support Organization in Austin, Texas, in April and May of 2007.

Bruno Blanchard is a Certified IT Specialist in France, working for the IBM worldwide ITD Technology Integration and Management Competency center. He has been with IBM for 24 years and has approximately 20 years of experience in AIX and IBM pSeries. He has helped to write several other IBM Redbooks publications in addition to this one. He is currently involved as an IT architect in projects deploying System p servers in On Demand data centers, server consolidation environments, and large server farms. His areas of expertise include System p virtualization and OS provisioning.

Pedro Coelho is an advisory IT Specialist with IBM Global Services in Portugal. He has eight years of experience in AIX and Linux® in the area of post-sale support and services. He holds a Masters degree in Computer Science from COCITE, Lisbon. His areas of expertise include HACMP™ and performance analysis and tuning. He is also working with IBM Learning Services to teach beginning and advanced classes on AIX 5L™ and Linux.

Mary Hazuka is an IBM System p Field Technical Sales Specialist (FTSS) in the Chicago area, a role she has held for the past seven years. As an FTSS, Mary has experienced first hand how clients implement IBM products in their data centers. Prior to IBM, she worked for five years at Sequent®, Inc., becoming an IBM employee when Sequent was acquired by IBM. During the transition, Mary learned Sequent's Dynix/PTX operating system for the NUMA systems, followed by IBM AIX. She credits the AIX System Management Interface Tool (commonly called SMIT) and IBM Redbooks publications with easing her switch to AIX. Mary holds two degrees in Music Education, a Bachelors degree from Mundelein College and a Masters degree from Illinois State University.

Jerry Petru is a Senior Technical Advocate and works in the AIX Collaboration Center based out of Austin, Texas. He has over twenty years of experience in the technical industry. He has worked extensively in support of large corporate UNIX® environments and has designed and implemented system security, high availability, and disaster recovery plans for some of the world's largest companies. Mr. Petru has written and taught more than twenty courses based on AIX, DNS, Internet Security, Linux, LPAR, and micro-partition technology. Jerry developed the Regatta Installation cook book on POWER4™ and led creation of the Advanced Power Virtualization workshop for POWER5™.

Theeraphong Thitayanun is a Certified Consulting IT Specialist for IBM Thailand. His main responsibility is to provide services and support in all areas of the System p product set. His areas of expertise include AIX/PSSP, LPAR/HMC, PLM, GPFS/AFS, HACMP, HACMP/XD for Metro Mirror and DB2® Universal Database™. He holds a Bachelors degree in Computer Engineering from Chulalongkorn University and, as a Monbusho student, a Masters degree in Information Technology from Nagoya Institute of Technology, Japan.

Production of this IBM Redbooks publication was managed by:

Chris Almond, an ITSO Project Leader and IT Architect based at the ITSO Center in Austin, Texas. In his current role, he specializes in managing technical content development projects focused on Linux and AIX 5L systems engineering. He has a total of 17 years IT industry experience, including the last eight with IBM.

Acknowledgements

This IBM Redbooks publication would not have been possible to write without the generous support of many IBM employees around the world. For their contributions, guidance, patience, and technical feedback in support of this project, we gratefully acknowledge the following IBM employees:

John Banchy

Frederic Barrat

Richard Bassemir

Laurent Dufour

Thierry Fauck

Khalid Filali-Adib

Eric Fried

Nigel Griffiths

Rick Houlihan

Vinit Jain

Rene Martinez

Rick Mason Jr

Rajeev Mishra

Jorge Rodriguez

Lance Russell

Edward Shvartzman

Marc Stephenson

Marina Stotz

Marcos Villarreal

Philip Warren

The team would also like to acknowledge the support for this project provided by Jay Kruemcke, IBM AIX Offering Manager, Satya Sharma, IBM Distinguished Engineer, and Scott Vetter, ITSO Project Leader.

Become a published author

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

Your efforts will help increase product acceptance and customer satisfaction. As a bonus, you will 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 books to be as helpful as possible. Send us your comments about this book or other IBM Redbooks publications in one of the following ways:

- ▶ Use the online **Contact us** review IBM Redbooks publications form found at:
ibm.com/redbooks

- ▶ Send your comments in an e-mail to:
redbooks@us.ibm.com

- ▶ Mail your comments to:

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

Archived



Part 1

Introduction and planning

In Part 1 of this book, we present detailed conceptual and architectural descriptions of WPAR technology. The goal of the chapters in Part 1 is to explain the benefits of using WPARs in different business contexts. This part of the book is intended to be read by Solution Architects and Planners. Part 1 contains the following chapters:

- ▶ Chapter 1, “Introduction to workload partition technology in AIX V6” on page 3
- ▶ Chapter 2, “Understanding and planning for WPARs” on page 19

Archived

Introduction to workload partition technology in AIX V6

Workload partitions (WPARs) provide a software solution for creating virtualized operating-system environments for managing multiple workloads. This chapter describes the IBM technology that has influenced this software solution, the terminology used to describe this new virtual environment, and situations in which you would want to use WPARs. This chapter includes the following sections:

- ▶ 1.1, “Overview of partitioning and virtualization capabilities prior to AIX V6” on page 4
- ▶ 1.2, “AIX V6 and WPAR-based system virtualization” on page 7
- ▶ 1.3, “What is a workload partition” on page 8
- ▶ 1.4, “Live application mobility” on page 11
- ▶ 1.5, “When to use workload partitions” on page 13

1.1 Overview of partitioning and virtualization capabilities prior to AIX V6

Today's competitive corporate environment requires nimble IT departments with the ability to respond quickly to changes in the capacity and the use of innovative methods to reduce time to market for new applications and systems. Escalating costs of power, raised floor capacity, and administrative costs also drive the need to utilize technology in new ways to maximize a company's IT investment.

Figure 1-1 on page 5 presents the various partitioning and virtualization technologies that have been integrated within AIX. Of the technologies presented in Figure 1-1 on page 5, Workload Manager (WLM) is the only one that is software-based.

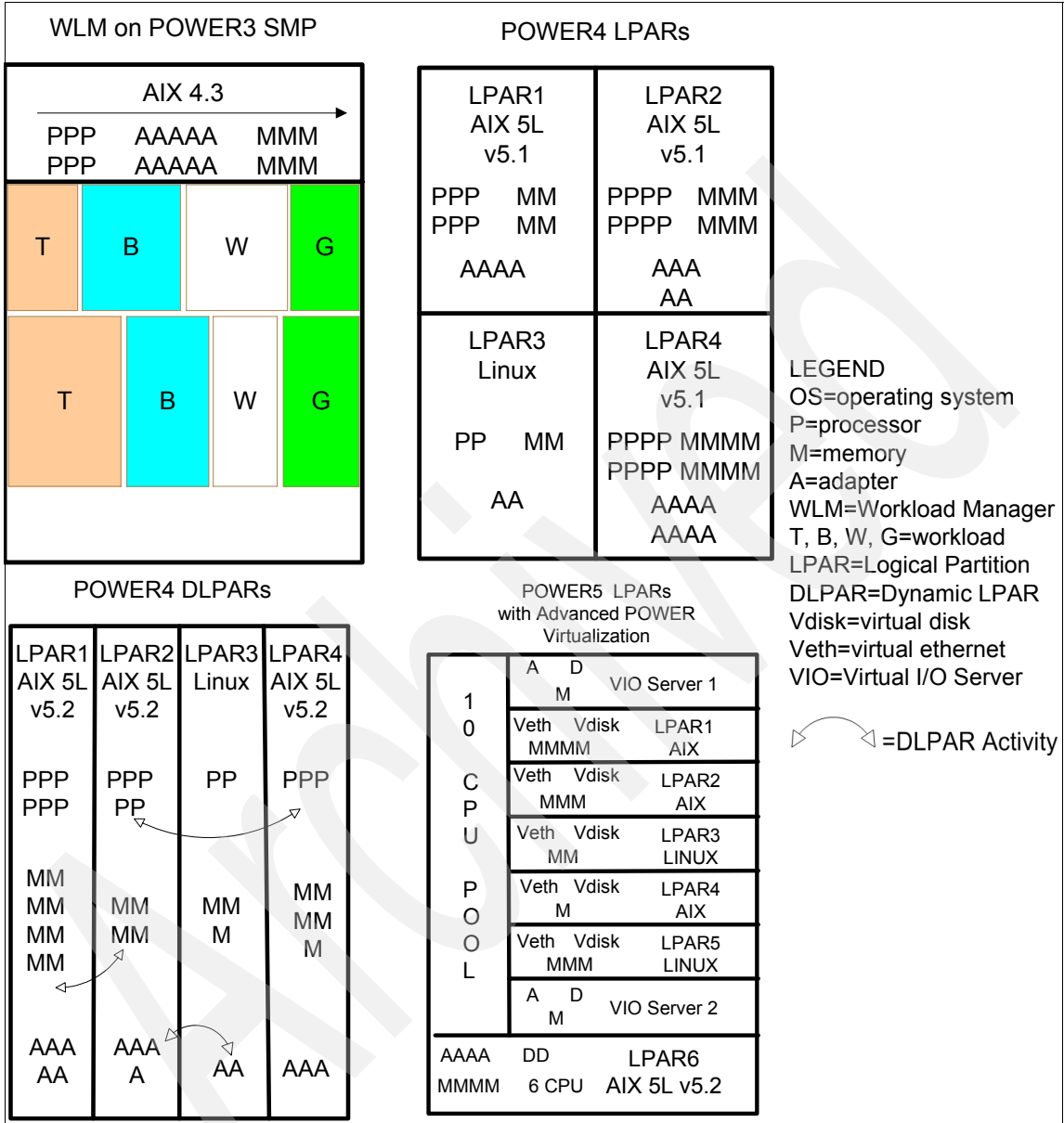


Figure 1-1 Overview: Advanced POWER™ virtualization technologies

AIX Workload Manager

Within AIX, Workload Manager has been part of the operating system since Version 4.3. It allows multiple workloads to run under one AIX instance. The system administrator builds rules based upon a user, process, or workload. Based upon these rules, shares of CPU and memory adjust to the workload with peak demand.

If you have used Workload Manager in the past, Chapter 7, “Resource control” on page 237 will be of interest to you to see the relationship between Workload Manager and Workload Partitions.

Logical partitions

With AIX 5.1 and POWER4 technology, IBM announced logical partitions (LPARs) as a way to provide greater flexibility and better utilization of large systems. Now, systems can run AIX and Linux in separate partitions starting at a minimum of one CPU, 1 GB of memory, and one Ethernet adapter. However, a reboot was required to move resources between LPARs.

Dynamic logical partitions

AIX 5.2 added more systems flexibility by being able to move CPU, I/O adapters, and memory dynamically without rebooting the LPARs. This innovation is the result of combining the technologies of firmware, the hypervisor, and AIX. It allowed IT environments to become more adaptable to their clients’ needs.

Advanced POWER Virtualization

The AIX 5.3 and POWER5 ability to virtualize CPUs, share Ethernet adapters, and virtually slice disks for provisioning to client LPARs allows IT staff to impress their customers and upper management. Virtualization is an excellent vehicle to address business needs while controlling costs. IBM System p Advanced Power Virtualization (APV) offers advanced technology to facilitate server consolidation, reduce costs, provide redundancy, and adapt capacity to quickly meet demand. APV can be used to reduce the need for static adapters, can rapidly respond to increasing capacity demands, and generally allows companies to utilize their purchasing dollars more effectively.

From this history of improving flexibility of system resources, we now have AIX V6. AIX V6 is capable of running on POWER4, POWER5, POWER5+™, PPC970, and POWER6™-based servers.

1.2 AIX V6 and WPAR-based system virtualization

With the release of AIX V6.1 in the last part of 2007, IBM will introduce a new virtualization capability called *workload partition (WPAR)*. WPAR is a purely software partitioning solution that is provided by the operating system. It has no dependencies on hardware features.

AIX V6 is available for POWER4, POWER5, POWER5+, and POWER6. WPAR can be created in all these hardware environments.

WPAR provides a solution for partitioning one AIX operating instance in multiple environments: each environment, called a *workload partition*, can host applications and isolate them from applications executing within other WPARs.

Figure 1-2 shows that workload partitions can be created within multiple AIX instances of the same physical server, whether they execute in dedicated LPARs or micropartitions.

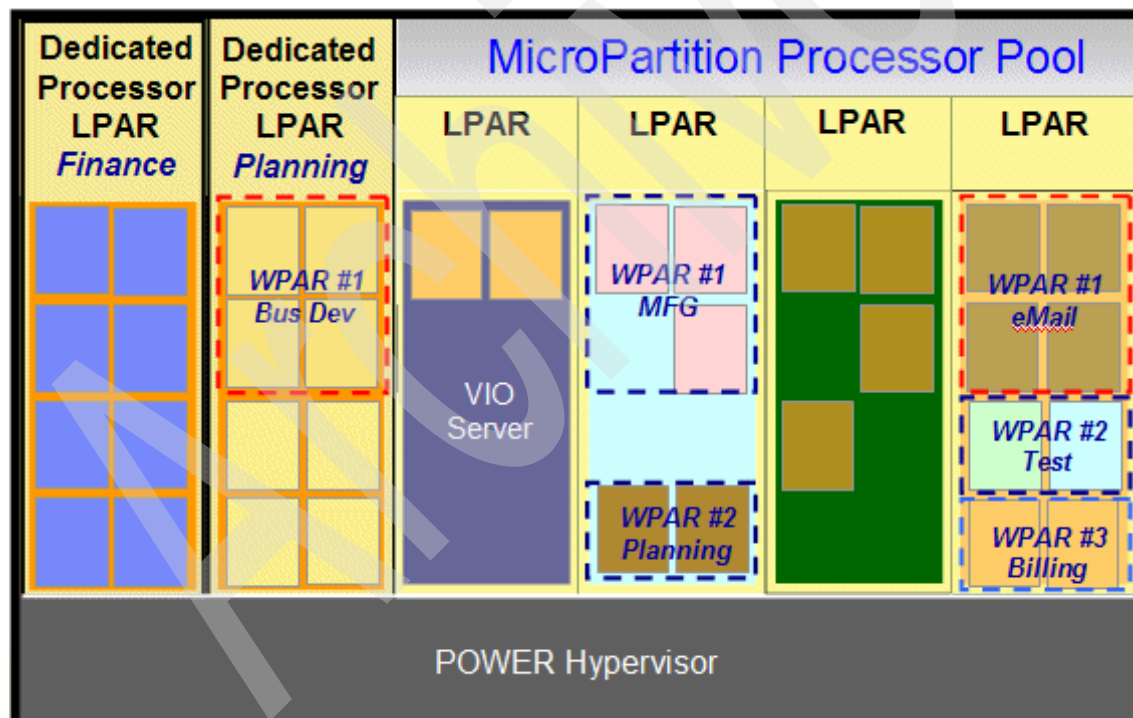


Figure 1-2 WPAR instantiated within dedicated partitions and micropartitions

The WPARs in this example are used to isolate the AIX processes belonging to one application from processes of other applications. For example, in the last LPAR, the eMail processes in WPAR #1 cannot interfere with the processes of the Test or Billing applications.

1.3 What is a workload partition

A WPAR is a software-created, virtualized OS environment within a single AIX V6 image. Each workload partition is a secure and isolated environment for the application it hosts. The application in a WPAR thinks that it is being executed in its own dedicated AIX instance.

Figure 1-3 on page 9 is a graphical overview of workload partitions in an AIX V6 environment. Workload partitions can be created within an AIX6 LPAR. At this point of the book, WPARs can be considered as a boundary around a set of AIX processes.

Note: Throughout this book, we use the term LPAR to refer indifferently to a micropartition or dedicated partition of a POWER-based server, or to a full physical server that is not partitioned (also known as a *full-system partition* in POWER4 terminology).

The term *global environment* is introduced in the AIX terminology to refer to the part of the AIX operating system that hosts workload partitions. Creating WPARs within an LPAR does not restrict the use of the hosting AIX instance. It is possible to log in to the global environment, to launch a program in the global environment, and to perform the exact same actions that are possible on any AIX instance that does not host WPARs.

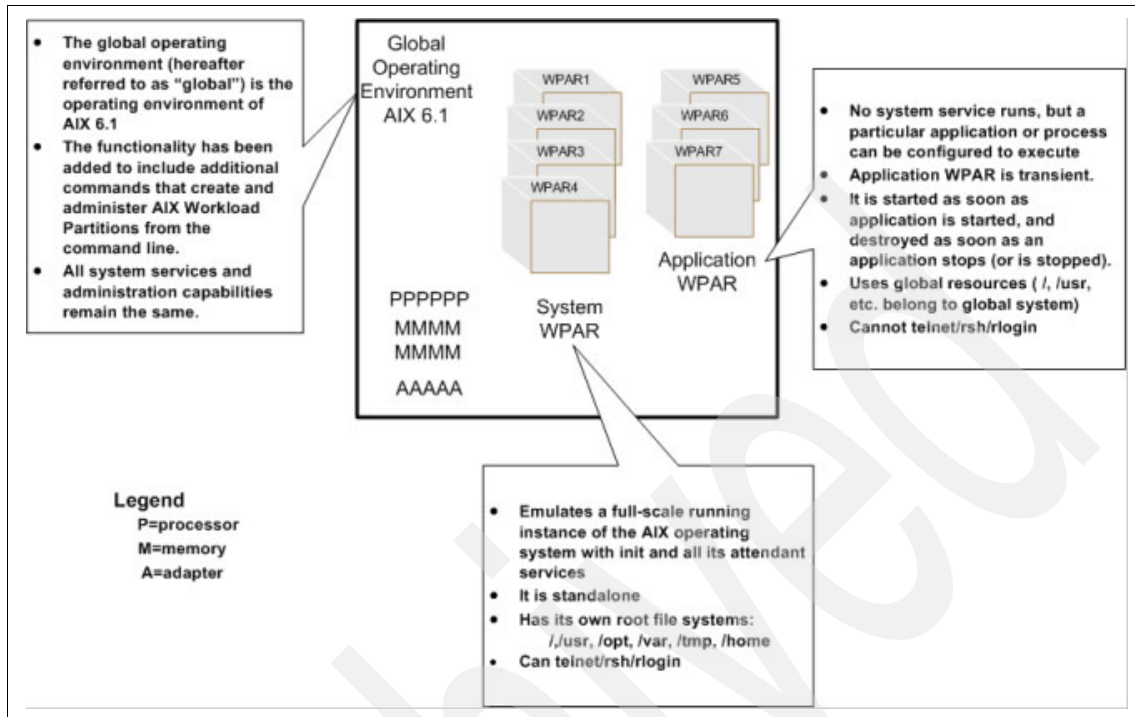


Figure 1-3 Global environment, system WPARs, and application WPARs

Figure 1-3 introduces new concepts, such as *application workload partitions* or *system workload partitions*.

An important feature of workload partitions is their ability to be relocated from LPAR to LPAR, whether these LPARs are hosted on the same physical server or on different physical servers. The most important new concepts described in the following sections include:

- ▶ The global environment
- ▶ The differences between the types of WPARs: *application and system*
- ▶ Live application mobility, which is also referred to as *workload partition mobility* or *workload partition relocation*

1.3.1 The global environment in an LPAR

As mentioned earlier, workload partitions are created within standard AIX V6 instances, and the global environment is the part of an AIX V6 instance that does not belong to any workload partition. The global environment is therefore similar to the operating system environment of earlier versions of AIX. This global environment can be hosted within a dedicated LPAR or a micropartition.

A system administrator must be logged in to the global environment to create, activate, and manage workload partitions. Workload partitions cannot be created within other workload partitions.

The global environment owns all physical resources of the LPAR: network adapters, disk adapters, disks, processors, and memory. It allocates CPU and memory resources to the workload partitions. It provides them access to the network and storage devices.

The global environment has visibility into the workload partitions. It is possible from the global environment to see (and control) the processes executing within the WPARs and to see the file systems used by the WPARs.

Most performance monitoring and tuning activities are performed from the global environment.

1.3.2 WPARs

To most applications, the WPAR appears as a booted instance of AIX. In general, applications can run without modification in a WPAR.

Inside the WPAR, the applications:

- ▶ Have private execution environments
- ▶ Are isolated from other processes outside the WPAR, signals, and file systems (file system isolation only applies to system WPARs)
- ▶ Might have dedicated network addresses
- ▶ Have interprocess communication that is restricted to processes executing in the same workload partition.

There are two types of workload partitions that can reside in a global environment:

- ▶ System WPAR: almost a full AIX environment.
- ▶ Application WPAR: a light environment suitable for execution of one or more processes.

1.3.3 System WPARs

A system WPAR is similar to a typical AIX environment. Each System WPAR has dedicated writable file systems, although it can share the global environment /usr and /opt file systems in read only mode. When a system WPAR is started, an *init process* is created for this WPAR, which in turns spawns other processes and daemons. For example, a system WPAR contains an *inetd* daemon to allow complete networking capacity, making it possible to remotely log in to a system WPAR. It also runs a *cron* daemon, so that execution of processes can be scheduled.

1.3.4 Application WPARs

If an application or group of applications can be started with one command of the AIX command-line interface, it is a candidate to be hosted by an application WPAR. This command is passed as an argument to the `wparexec` command that will create an application WPAR. As soon as the passed command exits, the workload partition is terminated.

An application partition shares the file system of the global environment. It does not own any dedicated storage.

An application partition can run daemons. But application partitions will not run any of the system service daemons, such as *inetd*, *srcmstr*, and so forth. It is not possible to remotely log in to an application partition or remotely execute an action into an application WPAR.

1.4 Live application mobility

Both types of workload partitions, the system WPAR and the application WPAR, are capable of being configured to support mobility, or *relocation*.

Distinction: In 2007, IBM System p6 and AIX V6 have two features that seem similar, but are different: WPAR mobility and live partition mobility:

- ▶ WPAR mobility, which is discussed in this book, is a feature of AIX V6 and WPAR Manager. It is available on POWER4, POWER5, and POWER6 systems.
- ▶ Live partition mobility relies on the POWER6 hardware and hypervisor technology (Advance Power Virtualization). It is available on POWER6 systems only. *This feature is also available to AIX 5.3 LPARs.*

The capability to move one WPAR from one LPAR to another, possibly from one physical system to another, can be executed on active partitions. In this case, the application undergoes active relocation (it is *hot-migrated*) without stopping the application. The only visible effect for a user of the application is a slightly longer response time while the application is migrating.

Workload partition mobility uses checkpoint and restart features to move workload partitions. The checkpoint saves the current status of the application and then restarts it on a new system or OS instance at the previously saved state.

Important: Workload partition mobility is a software solution that is dependent on AIX V6 for execution. When used for the migration of a WPAR from one LPAR to another or between physical systems, then *hardware and software compatibility is required*.

Partition mobility is not a replacement for a high availability solution. The premise allows for planned migrations of workloads from one system to another so that the application is uninterrupted, for example, during hardware maintenance or a firmware installation on the server. The workload does not need to be aware of the migration for the most part. But proper planning and testing are always recommended before moving anything into a production environment.

Figure 1-4 depicts the use of WPAR relocation for workload balancing where two applications are moved between two servers to balance the load of these servers. This figure also introduces the concept of WPAR Manager that is described in Chapter 2, “Understanding and planning for WPARs” on page 19.

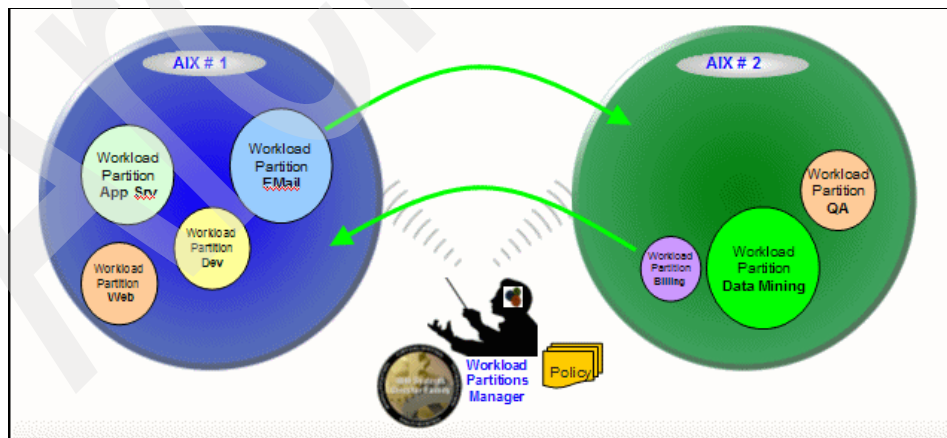


Figure 1-4 WPAR migration

1.5 When to use workload partitions

Workload partitions offer new possibilities for managing AIX environments. They complement other virtualization solutions available for System p6 platforms. The following scenarios show the benefit of using WPARs.

1.5.1 Improvement of service level agreements

Hardware components of an IT infrastructure might need to undergo maintenance operations requiring the component to be powered off. If an application is not part of a cluster of servers providing continuous availability, either for technical, organizational, or cost reasons, WPARs can help to reduce the application downtime. Using the live partition mobility feature, the applications that are executing on a physical server can be temporarily moved to another server without an application blackout period during the period of time required to perform the server physical maintenance operations.

Long running jobs can take advantage of the checkpoint/restart feature of WPARs. It can be used to protect them against a failure that requires restarting all computations from the beginning. The checkpoint feature can be used to regularly capture a snapshot of the application runtime environment, without having to instrument the code. In the case where the job needs to be stopped before reaching completion of the computation, the job can be resumed in the state it was when the last checkpoint was saved.

The checkpoint/restart feature can also be used to execute long lasting batch jobs on a system with limited resources. This job can be run at nighttime, be paused during the daytime, when the computer resources have to be dedicated to other applications, such as transaction handling or Web serving, and then resumed at the beginning of the next night.

The workload partition technology can also help in an environment where an application needs to be started often, on-demand, and quickly. This might apply, for example, in test environments where resources are too scarce to keep multiple applications executing concurrently when not in use. Using WPARs, many applications can be defined on a server, but not activated. Activation of the workload partitions executing each of these applications can be performed only when needed for a test.

1.5.2 Protection of existing hardware investment

Although customers using POWER4 IBM pSeries® servers cannot take advantage of physical or hypervisor-based virtualization technology, the WPAR technology relies only on IBM AIX Version 6.1 with no dependency on the underlying hardware. It can be used on POWER4, POWER5, and POWER6-based servers.

Clients having many applications, each running a dedicated POWER-based server or dedicated partition and requiring only a fraction of the available processing power, can, thanks to the WPAR technology, consolidate these applications within one LPAR. Each application can be executed within one WPAR, providing a dedicated environment isolated from the other applications' environments, while all WPARs share the physical resource of one LPAR.

1.5.3 Optimization of resource usage

The IBM System p family offers many ways to optimize resource utilization through virtualization technologies, such as LPARs, DLPARs, and micropartitions. The WPAR technology complements the existing solution offerings thanks to its unique characteristics.

The WPAR technology gives you additional flexibility in system capacity planning as part of a strategy for maximizing system utilization and provisioning efficiency. Due to the static allocation of partitions in physical servers, in a typical IT environment, each server is sized with spare capacity to allow for resource consumption increase of all applications executing within this server. Thanks to the mobility feature of WPARs, the server sizing and planning can be based on the overall resources of a group of servers, rather than being performed server per server. It is possible to allocate applications to one server up to 100% of its resources. When an application grows and requires resources that can no longer be provided by the server, the application can be moved to a different server with spare capacity.

The same mobility feature, combined with the policy-based relocation functions of the WPAR Manager, allows you to size a set of servers to handle the peak load, based on the overall resource capacity of the set of servers, and not for each server. In a classical environment, each server must be able to support the peak load of all partitions hosted within that server. Thanks to the WPAR mobility, it is possible to take advantage of free resources in one physical server to offload another physical server hosting applications that require more resources than are locally available.

AIX V6 provides highly granulated control of CPU and memory resource allocation to workload partitions (down to 0.01% increments). This technology is

therefore suitable for server consolidation of very small workloads. This can be particularly interesting for the replacement of old servers, for which even 10% of one POWER5 or POWER6 processor (the smallest micropartition) exceeds the application needs.

The *theoretical* upper limit on the number of workload partitions that can be executed within one LPAR is 8192. In actual practice, your application environment will probably require far less than 8192 WPARs running within a single LPAR. And in practice, we expect that you will encounter other AIX system limitations preventing you from actually approaching this theoretical limit.

Note: In practice, the number of WPARs, which can be created and made active in an LPAR, depends upon the capacity of the system, the configuration of the WPARs, and the characteristics of the applications being run in those WPARs.

1.5.4 Highly granular control of resource allocation

When multiple applications are executing within the same AIX instance, the system administrator might want to control the amount of CPU and memory resources used by each application. One way to perform this control is to set up the Workload Manager (WLM) functions that is part of standard AIX features.

The WPAR technology provides a new way to perform this resource control. The WPAR resource control reuses the WLM technology, but encapsulates it in a way that WLM is invisible to the system administrator. There is no need for the system administrator to know about WLM. The resource control is available through options of the WPAR command line and SMIT interfaces.

The WPAR resource control feature allows the system administrator to arbitrate between applications competing for CPU and memory resources. This guarantees that each application receives a share of the CPU and memory resource available from the global environment. These resources are separate from the requirements of the other applications executing in WPARs within the same operating system instance.

1.5.5 Control of security and privilege command

In large AIX environments, where a partition hosts many applications, it is not unusual to have multiple people acting as system administrators. However, all of them might not need root or superuser privileges in all domains of system administration. These people can be specialized for activities, such as user administration, network control, storage control, or software maintenance.

The WPAR technology supports this specialization of roles and can help restrict the privileges given to one person to just the scope that person needs to control. System workload partitions have their own user set, independent from the user set defined at the global environment level. An individual, who is using root within a system workload partition, only has superuser privileges for the resources visible within this WPAR. This user cannot control global environment resources, such as network adapter or physical devices, and cannot act on resources belonging to other workload partitions. Many applications need the application administrator to use the root user to control the application, even if this person does not need to manage the operating system. The WPAR technology allows you to delegate the superuser privileges to one individual and limit them to an application environment without jeopardizing the global environment.

The separation of user sets (or security domains) between different system workload partitions also enables the system administrators to isolate groups of users logging on in AIX environments according to their application access control requirements. Users defined in one system WPAR are unaware of the applications executing in the global environment or in other WPARs. They cannot see the list of users or processes outside their WPAR.

IBM AIX Version 6.1 provides improvement over the previous AIX 5L Version 5.3 for role-based control of user privileges. This feature is known as *Role-Based Access Control* (RBAC). An exhaustive description of these new features is available in *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430.

WPAR integrates the use of RBAC features for controlling privileges. A default RBAC setting is provided with each WPAR, but the system administrator can also further customize the RBAC configuration used in a WPAR context.

1.5.6 Simplified handling of software stack

The WPAR technology can help the system administrator simplify the way that the system administrator maintains the operating systems and application software stacks.

For a long time, the traditional approach to application deployment has been to dedicate one server to one application. With the advent of virtualization and partitioning technologies, it has been possible to host multiple applications within partitions of a physical server. But this solution still implies that the system administrator needs to maintain one operating system instance for each application. The WPAR technology allows you to share an AIX instance between multiple applications, while still running each application within its own environment, providing isolation between applications. In this case, the more applications that are consolidated within one AIX instance, the less the system

administrator has to perform OS fix applications, backups, migration, and other OS maintenance tasks. However, note that this type of consolidation requires that all applications can run under the same version and maintenance level of the OS.

In addition to sharing the operating system, the system administrator can take advantage of the WPAR technology to share application code. In a traditional AIX environment, if several Apache Web servers are needed, they each need to be deployed in a dedicated server or LPAR. In a WPAR environment, it is possible to install Apache in one LPAR and then execute multiple instances of the Apache server within this LPAR, by starting multiple WPARs. Each WPAR runs its own Apache server with its own data in dedicated disk space, but shares the Apache code with all other WPARs. This type of a configuration optimizes memory utilization by eliminating duplication of code and reduces administration maintenance of the Apache code, which only needs to be updated once for all server instances.

IBM AIX Version 6.1 introduces a new concept in software installation and management: *relocatable software packages*. A relocatable application is an application where the files can be installed relative to a base directory that is different from the / root directory of the AIX environment. Using this feature, it is possible to deploy multiple versions of the same application within one AIX instance. The system administrator can take advantage of relocatable applications by starting each version of the application in a specific WPAR, therefore providing multiple servers with different server code versions from one LPAR.

1.5.7 Simplified handling of application OS environment

The workload partition configuration can be stored in human-readable specification files. These specification files can be generated by the operating system from preexisting workload partitions or can be edited, created, or modified manually. In an environment where a system administrator has to manage several application environments, the WPAR technology can help the system administrator quickly clone and define new application environments. These specification files can be used as input to WPAR creation commands, allowing the system administrator to automate through scripts and programs the startup and handling of multiple workload partitions.

1.5.8 Business continuity: Disaster or failure recovery solution

The WPAR technology can be integrated as one element of a solution to provide a business continuity plan.

The checkpointing feature of WPAR allows you to capture a snapshot of an executing application without having to instrument the code. The application checkpoint image is then saved to a file that can later be used to resume execution of an application. Combined with a backup of the application data, the WPAR checkpoint feature can provide an alternate disaster or failure recovery solution for applications that do not use other solutions, such as HACMP or server clusters.

1.5.9 Supporting “Green” computing strategies

Using WPAR relocation features for live application mobility means that you have the flexibility to consolidate workloads during periods of low usage onto smaller numbers of operating server platforms. In this strategy, you still provide continuous application availability, but you do so using a smaller number of powered up servers. As you approach normal high usage periods, you can then power up additional peak demand server resources and relocate cyclical workloads back to those machines during those peak demand periods. For example, if your data center peak workload periods are 12 hours per day, 5 days per week, peak load systems only need to be powered up approximately 35% of the time.

Understanding and planning for WPARs

This chapter describes workload partitioning (WPAR) technology and how it can be implemented to work in your IT environment. The chapter is designed to provide system architects and system administrators the level of knowledge required to plan the deployment of WPARs in their IT infrastructure.

This chapter includes the following sections:

- ▶ 2.1, “High-level planning information” on page 20
- ▶ 2.2, “General considerations” on page 21
- ▶ 2.3, “Global environment considerations” on page 26
- ▶ 2.4, “Application WPARs” on page 27
- ▶ 2.5, “System WPARs” on page 28
- ▶ 2.6, “WPAR mobility” on page 31
- ▶ 2.7, “WPAR and LPAR comparison” on page 36

2.1 High-level planning information

The WPAR technology is purely software-based. It can therefore be deployed on any hardware platform that supports AIX V6:

- ▶ IBM pSeries with POWER4 processors
- ▶ IBM System p
- ▶ IBM BladeCenter® JS21 with PPC 970 processors

The WPAR offering consists of two parts:

- ▶ IBM AIX Version 6.1 contains the base support for WPAR technology. This includes creation and management of both application and system workload partitions within the LPAR where AIX V6 is installed. AIX provide WPAR support and management through the AIX command-line interface and SMIT menus.
- ▶ IBM Workload Partitions Manager™ for AIX is a optional separately installable licensed program product that supports more advanced features:
 - Graphical User Interface including wizards for most management activity
 - Management of multiple WPARs on multiple servers from a single point of control
 - Enablement for WPAR mobility
 - Automated and policy-based WPAR mobility

The decision to use WPAR technology depends on the potential benefits that this technology can yield to a specific user environment. We described these benefits in 1.5, “When to use workload partitions” on page 13.

After the decision to use WPAR has been made, the planning activity consists of deciding:

- ▶ Which is the best-suited workload partition type: application or system WPARs?
- ▶ Is application mobility required?

The answers to these questions have technical consequences that are described in the following sections.

2.2 General considerations

The WPAR provides isolation of software services, applications, and administration, utilizing flexible software-defined boundaries within a single instance of the AIX V6.1 operating system (global environment). When building a WPAR from the command line, you can configure and start it within a few minutes.

This technology presents system administrators with new planning considerations related to network, file systems, and OS versions. Network aliases, shared or Network File Systems (NFS), and kernel unicity require a different approach to the planning of application deployment.

2.2.1 Networking

When planning for networks, you must understand how to get the most out of this technology. Using aliases decreases the number of adapters needed for communications but requires careful planning of bandwidth utilization, because several WPARs can share the same adapter.

NFS is a prerequisite to the WPAR mobility functionality. Three components are involved in NFS communications for WPAR mobility:

- ▶ The name and IP address of the global environment
- ▶ The name and IP address of the WPAR
- ▶ The name and IP address of the NFS server

Because they all play a role in this communication, they all must know each other. Preferably put them all in the same subnet. For more detailed explanation, refer to Chapter 6, “IBM WPAR Manager for AIX” on page 125.

2.2.2 Deployment of the Workload Partition Manager

The Workload Partition (WPAR) Manager provides a central systems management solution by providing a set of Web-based system management tools and tasks that simplify the management of a client’s server and WPAR infrastructure.

The WPAR Manager offers infrastructure resource optimization, service performance optimization, and service availability optimization tools. Specific features of the Workload Partition Manager include a centralized, single point of administrative control for managing both system and application WPARs, a browser-based GUI for dynamic resource management, system and application level workload management through WPARs, role-based views and tasks,

dynamic allocation/reallocation and configuration of virtual servers, storage, and network, and uninterrupted maintenance zero downtime for server fixes and upgrades through virtual server relocation.

Using the WPAR Manager involves three roles:

- ▶ A WPAR Management Server is a Java™ application running in an AIX server. This can be a stand-alone server or an LPAR in a shared physical server (dedicated or micropartition).
- ▶ The WPAR Management clients are installed in each LPAR where you plan to deploy WPARs and communicate with the WPAR Management Server.
- ▶ The WPAR Manager User Interface is a lightweight browser-supported interface to the WPAR Management Server. The interface can be provided by any Web browser with an IP connection to the WPAR Management Server. The UI allows for the display of information that has been collected through the agents and also provides management capability, such as the creation, deletion, and relocation of WPARs.

Figure 2-1 on page 23 shows where the components of the WPAR Manager execute.

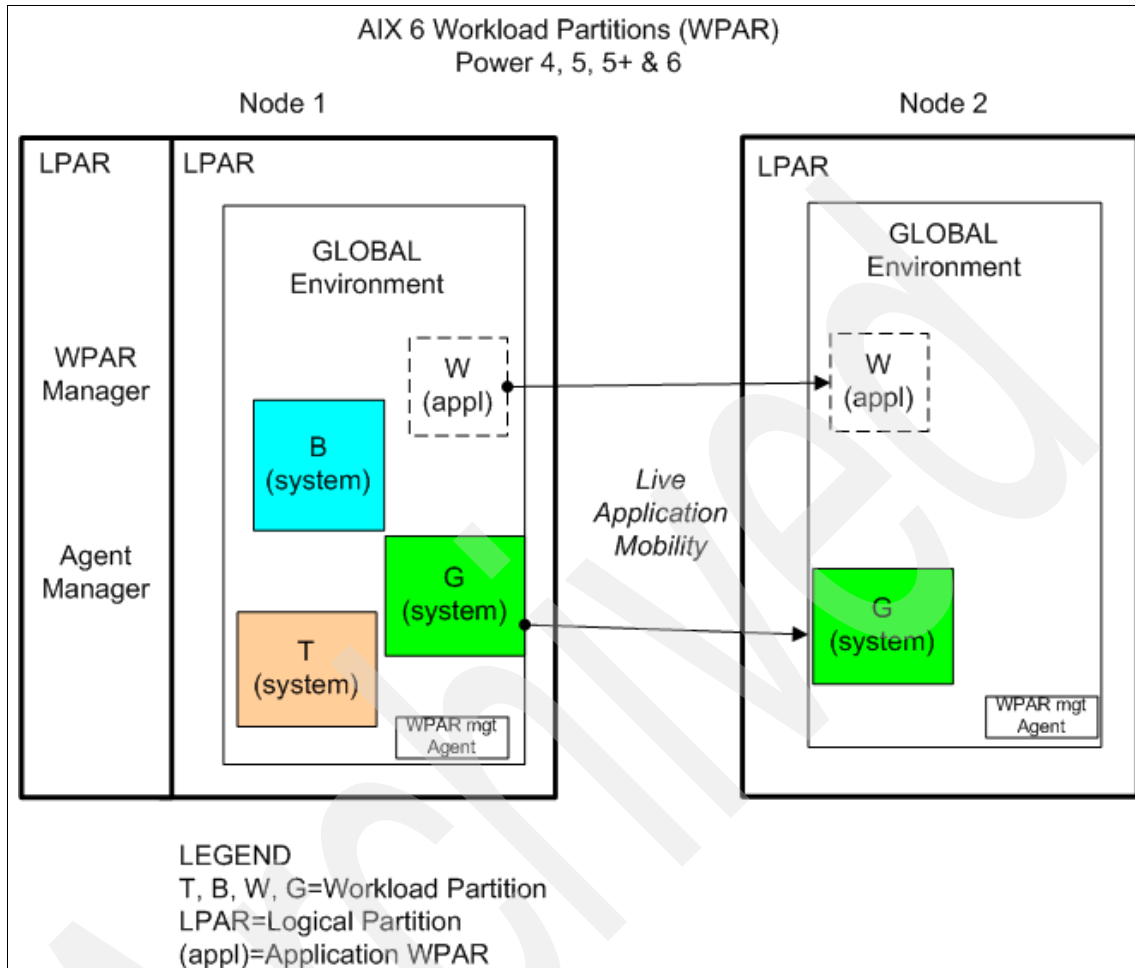


Figure 2-1 Workload partitions (WPARs) running in POWER4, 5, 5+, or 6 nodes

When planning the deployment of WPAR Manager components on different LPARS and workstations, the network firewalls must be configured to allow traffic to the specific ports listed on Figure 2-2 on page 24.

Note: Figure 2-1 contains the default values of the ports used by the WPAR Manager. The system administrator can modify these values when configuring the WPAR Manager.

Ports 14080 and 14443 are used for communication between the system administrator workstation and the WPAR Manager.

Ports 9510, 9511, 9512, and 9513 are used for communications among the agents and managers.

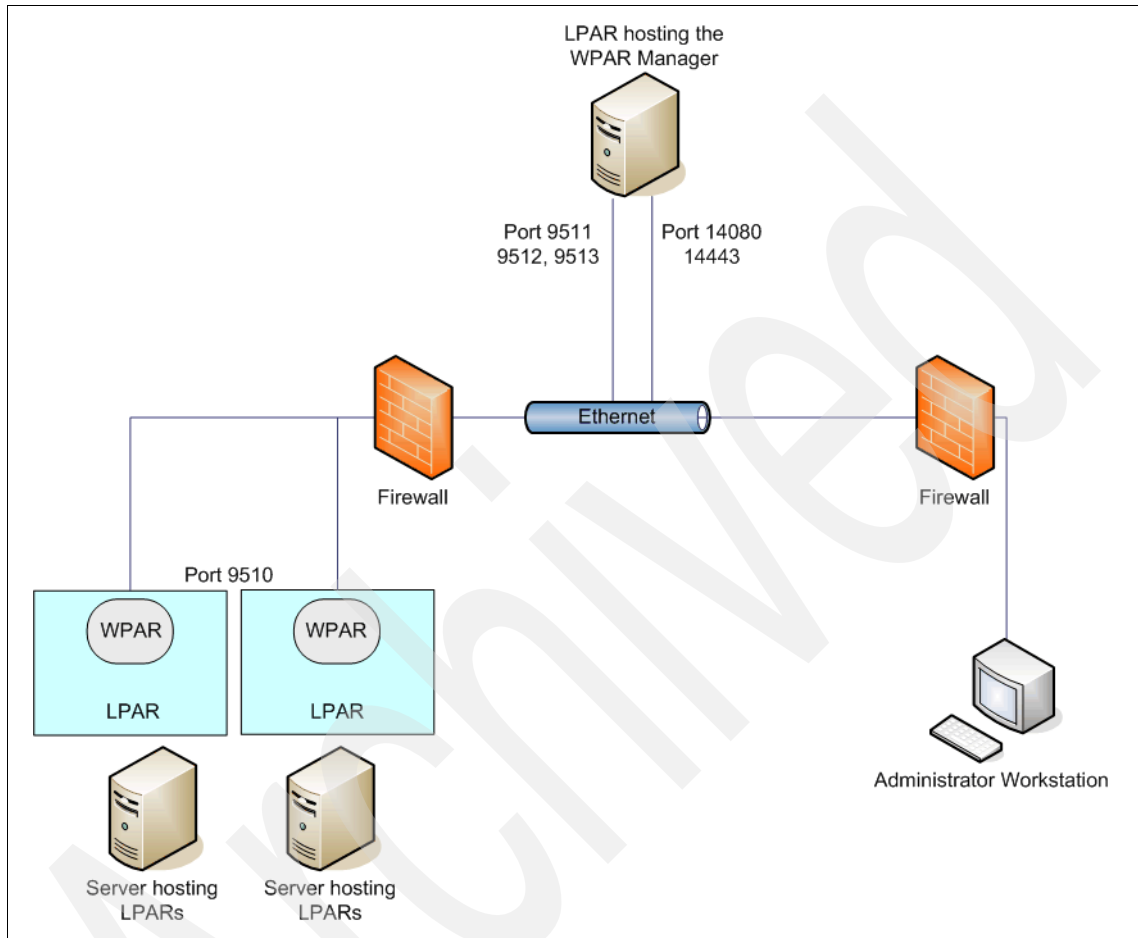


Figure 2-2 TCP/IP ports to configure on the firewall to use the WPAR Manager

2.2.3 Software prerequisites

Having a single instance of AIX simplifies the installation and general administration of the WPARs. Software is installed once and used many times in many WPARs. Although totally isolated from each other, these WPARs use the same AIX kernel instance. This means that all WPARs use the exact same level of AIX. When planning for WPARs, you must make sure that all application software supports the level of AIX of the global environment. *It is even more important that you plan for the future.* Updating or upgrading AIX in the global

environment means updating or upgrading AIX in all hosted WPAR environments. If you have an application that needs a specific version of AIX and cannot be updated, move it to a different LPAR so that it does not prevent the other WPARs from updating.

2.2.4 File system considerations

System WPARs created with the default options have shared read-only /usr and /opt file systems. This speeds up the creation, installation, and updating of WPARs and also prevents the accidental removal of system software that is shared with other WPARs. Having the read-only shared /usr and /opt file systems might not suit every application; certain applications are designed to write in the /usr or /opt file systems. One solution is to define the needed application's writable directory as a different file system and link it to the mount point that the application needs. Refer to 5.3.3, "Shared /usr with writable file system" on page 98 for an explanation of how a WPAR can have a writable directory under a read-only /usr or /opt. Another solution is for the application to not use the global environment shared /usr or /opt file systems. This solution requires extra disk space, because it duplicates the global environment's /usr or /opt to the WPAR's private and fully writable file systems.

Consolidating many applications within one global environment changes the way the system administrator manages file systems. Instead of managing multiple LPARs, each with a few file systems, the system administrator now manages only one LPAR with many file systems. In both cases, the overall number of file systems remains in the same order of magnitude (although using WPARs slightly reduces this number), but they are controlled within a single system. By default, a system WPAR has four dedicated file systems, two shared read-only file systems (/usr/ and /opt), and the /proc file system. For example, deploying 200 system WPARs in one global environment will result by default in a global environment with 800 separate file systems and 1200 mount points in the /proc pseudo-file systems. The WPAR technology provides an option to reduce this number of file systems. Instead of using the default file system creation option, the system administrator can choose to create one single file system per WPAR, as described in 5.1.4, "Creating WPARs: Advanced options" on page 74. This solution creates only one real file system (the root "/" file system) for the WPAR, and subtrees /var, /tmp, and /home are just created as subdirectories of the "/" file system, instead of real file systems as they usually are in AIX instances and the default system WPAR.

File systems of each system WPAR are created in the global environment directory tree and are mounted under the WPAR *base directory*. One base directory is defined per WPAR. The default path of the base directory is /wpars/<partition-name>. When planning to deploy several system partitions, the

system administrator might want to organize the base directory in a different directory tree organization.

2.4, “Application WPARs” on page 27 to 2.6, “WPAR mobility” on page 31 explain in more detail the file system considerations for application WPARs, system WPARs, and when mobility is used.

2.3 Global environment considerations

As described in Figure 2-1 on page 23, an LPAR (server, logical partition, or micropartition) hosts an AIX V6 instance. The AIX instance can contain one or more workload partitions. It is possible to mix application and system WPARs within the same AIX instance. The part of AIX outside of the workload partitions is referred to as the *global environment*.

If the WPAR Manager is used, the global environment also contains a WPAR Manager agent.

The global environment, just like any classical AIX instance, has one or more dedicated networks, IP addresses, and disks along with unique users and groups. The global environment can use physical or virtual adapters.

The hosted workload partitions have no control of hardware devices. The global environment therefore also owns all physical I/O adapters needed by the workload partitions:

- ▶ Enough adapters must be configured on the global environment to support the combined I/O throughput of all hosted partitions.
- ▶ The global environment must have access to all disks that will contain the file systems used by each hosted WPAR.
- ▶ If WPARs need to have IP connectivity, they will have an IP address that needs to be configured as an alias on one of the physical network adapters of the global environment.

It is theoretically possible to create up to 8192 WPARs within one physical AIX system. Thorough planning of the I/O configuration of the global environment is therefore mandatory.

For the best utilization and consolidation of the system, a combination of virtualization from the Advanced Power Virtualization package (VIOS) is recommended with WPAR systems. This allows for the quick deployment of systems and resources.

2.4 Application WPARs

There are two types of workload partitions. The simplest is an *application WPAR*. It can be viewed as a shell that spawns an application and can be launched from the global environment. This is a lightweight application resource; it does not provide remote login capabilities for users. It only contains a small number of processes, which are all related to the application, and uses the services of the global environment daemons and processes.

The application WPAR shares the operating system file systems with the global environment. It can be set up to receive its application file system resources from disks owned by the hosting AIX instance or from an NFS server.

Figure 2-3 on page 28 shows the relationship of an application WPAR's file systems to the default global environment file systems. The file systems that are visible to processes executing within the application WPAR are the global environment file systems shown by the relationships in the figure.

If an application WPAR accesses data on an NFS mounted file system, this file system must be mounted in the global environment directory tree. The mount point is the same when viewed from within the WPAR as it is when viewed from the global environment. The system administrator of the NFS server must configure the `/etc/exports` file so that file systems are exported to both the global environment IP address and to the application WPAR IP address.

Processes executing within an application WPAR can only *see* processes that are executing within the same WPAR. In other words, the use of Inter Process Communication (IPC) by application software is limited to the set of processes within the boundary of the WPAR.

Application WPARs are temporary objects. The life span of an application WPAR is the life span of the application that it hosts. An application WPAR is created at the time that the application process is instantiated. The application WPAR is destroyed when the last process running within the application partition exits. An application WPAR is a candidate for mobility. It can be started in one LPAR and relocated to other LPARs during the life of its hosted application process.

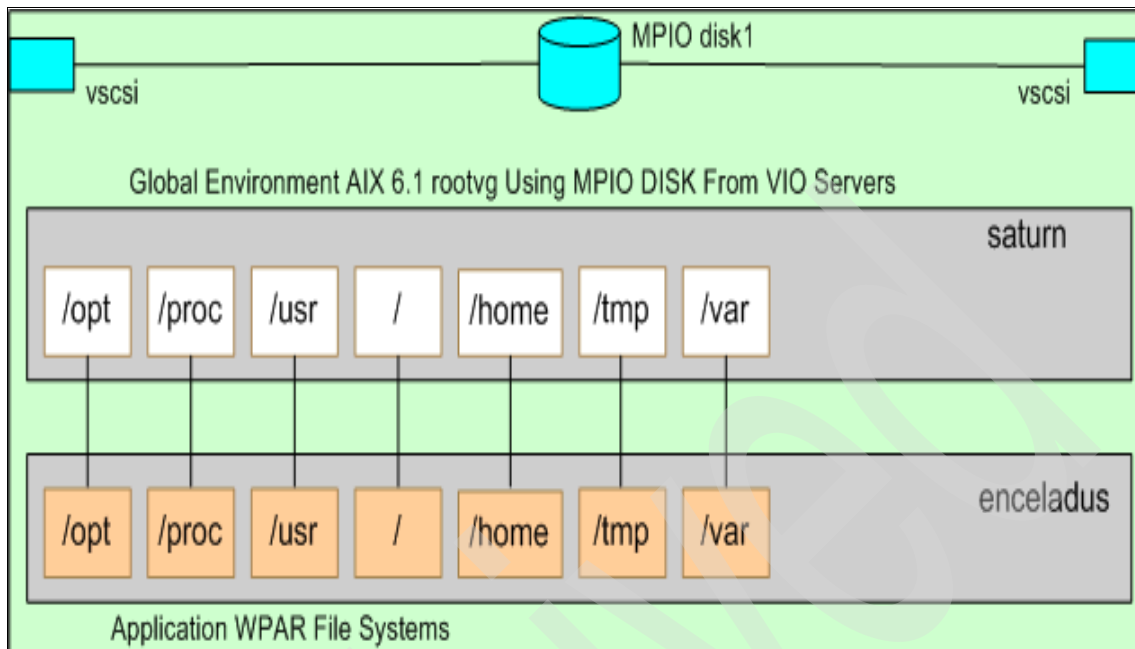


Figure 2-3 File system relationships from the global environment to the application WPAR

2.5 System WPARs

The second type of WPAR is a *system WPAR*. A system WPAR provides a typical AIX environment for executing applications with certain restrictions. A system WPAR has its own runtime resources. It contains an init process that can spawn daemons. For example, it has its own inetd daemon to provide networking services and its own System Resource Control (SRC).

Every system WPAR has its own unique set of users, groups, and network interface addresses. The users and groups defined within a system WPAR are completely independent from the users and groups defined at the global environment level. In particular, the root user of the WPAR only has superuser privileges within this WPAR, and has no privilege in the global environment (in fact, the root and other users defined within the WPAR cannot even access the global environment). In the case of a system partition hosting a database server, the DB administrator can, for example, be given root privilege within the DB WPARs without giving the DB administrator any global environment privileges.

The environment provided by a system WPAR to its hosted application and processes is a *chroot* complete AIX environment with access to all AIX system

files that are available in a native AIX environment. The creation of a system WPAR includes the creation of a *base directory* (see the WPAR documentation). This base directory is the root of the chroot system WPAR environment. By default, the path to this base directory is `/wpar/<name_of_wpar>` in the global environment.

By default, the base directory contains seven file systems:

- ▶ The `/`, `/home`, `/tmp`, and `/var` file systems are real file systems that are dedicated to the system partition's use.
- ▶ The `/opt` and `/usr` file systems are read-only namefs mounts over the global environment's `/usr` and `/opt`.
- ▶ The `/proc` pseudo-file system maps to the global environment `/proc` pseudo-file system (`/proc` in a WPAR only makes available process information for that WPAR).

Figure 2-4 depicts an overview of these file systems, viewed from the global environment and from within the system WPAR. In this example, a WPAR called `titian` is hosted in an LPAR called `saturn`. Although the diagram shows the global environment utilizing VIOs with two `vscsi` adapters along with virtual disk and using AIX native MPIO for a highly available rootvg, the system can be set up and supported with physical adapters and disk.

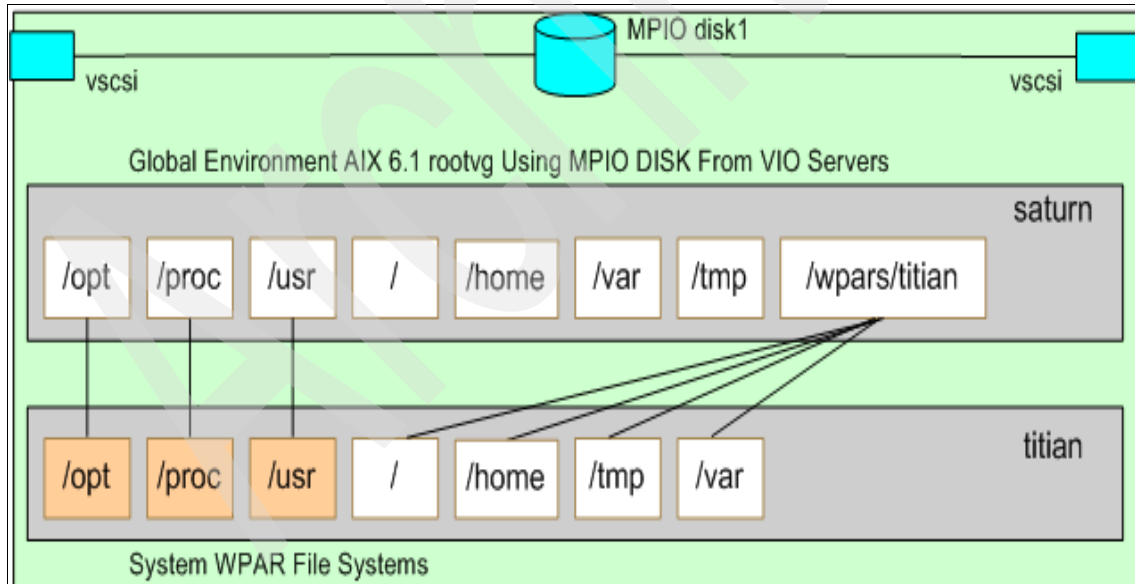


Figure 2-4 File system relationships from the Global Environment to the System WPAR

In Figure 2-4 on page 29, boxes with a white background symbolize real file systems, while boxes with orange backgrounds symbolize links. The gray box labeled `titian` shows the path name of the file systems as they appear to processes executing within the system WPAR. The gray box labeled `saturn` shows the path name to the file systems used within the global environments, as well as the `basedir` mount point below which the system WPARs are created.

Example 2-1 shows the `/wpars` directory created within the global environment file system that will be used to contain the system WPAR file systems.

Example 2-1 Listing files in the global environment

```
root: saturn:/ --> ls -ald /wpars
drwx-----  5 root      system          256 May 15 14:40 /wpars
root: saturn:/ -->
```

Now, within the `/wpars` directory, there is a `titian` directory as shown in Example 2-2.

Example 2-2 Listing /wpars in the global environment

```
root: saturn:/wpars --> ls -al /wpars
drwx-----  3 root      system          512 May  1 16:36 .
drwxr-xr-x  23 root      system         1024 May  3 18:06 ..
drwxr-xr-x  17 root      system         4096 May  3 18:01 titian
```

In Example 2-3, we see the mount points for the file system of the operating system of `titian` as created from `saturn` to generate this system WPAR.

Example 2-3 Listing the contents of /wpars/titian in the global environment

```
root: epp182:/wpars/titian --> ls -al /wpars/titian
drwxr-xr-x  17 root      system         4096 May  3 18:01 .
drwx-----  3 root      system          512 May  1 16:36 ..
-rw-----  1 root      system          654 May  3 18:18 .sh_history
drwxr-xr-x  2 root      audit           256 Mar 28 17:52 audit
lrwxrwxrwx  1 bin       bin              8 Apr 30 21:20 bin -> /usr/bin
drwxrwxr-x  5 root      system         4096 May  3 16:41 dev
drwxr-xr-x  28 root      system         8192 May  2 23:26 etc
drwxr-xr-x  4 bin       bin             256 Apr 30 21:20 home
lrwxrwxrwx  1 bin       bin              8 Apr 30 21:20 lib -> /usr/lib
drwx-----  2 root      system          256 Apr 30 21:20 lost+found
drwxr-xr-x 142 bin       bin            8192 Apr 30 21:23 lpp
drwxr-xr-x  2 bin       bin             256 Mar 28 17:52 mnt
drwxr-xr-x  14 root     system          512 Apr 10 20:22 opt
dr-xr-xr-x  1 root     system           0 May  7 14:46 proc
drwxr-xr-x  3 bin       bin             256 Mar 28 17:52 sbin
drwxrwxr-x  2 root     system          256 Apr 30 21:22 tftpboot
```


drwxrwxrwt	3	bin	bin	4096	May	7	14:30	tmp
lrwxrwxrwx	1	bin	bin	5	Apr	30	21:20	u -> /home
lrwxrwxrwx	1	root	system	21	May	2	23:26	unix -> /usr/lib/boot/unix_64
drwxr-xr-x	43	bin	bin	1024	Apr	27	14:31	usr
drwxr-xr-x	24	bin	bin	4096	Apr	30	21:24	var
drwxr-xr-x	2	root	system	256	Apr	30	21:20	wpars

Example 2-4 shows the output of the **df** executed from the saturn global environment. It shows that one system WPAR is hosted within saturn, with its file systems mounted under the /wpars/titian base directory. The example shows that the /, /home/ /tmp, and /var file systems of the system WPAR are created on logical volumes of the global environments. It also shows that the /opt and /usr file systems of the WPAR are namefs mounts over the global environment /opt and /usr.

Example 2-4 Listing mounted file system in the global environment

```
root: saturn:/wpars/titan --> df
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
/dev/hd4         131072          66376   50%     1858    6% /
/dev/hd2         3801088         646624   83%    32033    7% /usr
/dev/hd9var      524288          155432   71%     4933    8% /var
/dev/hd3         917504          233904   75%      476    1% /tmp
/dev/hd1         2621440         2145648  19%      263    1% /home
/proc            -                -         -         -        - /proc
/dev/hd10opt     1572864         254888   84%     7510    4% /opt
glear.austin.ibm.com:/demofs/sfs 2097152  1489272  29% 551 1% /sfs
/dev/fs1v00     131072          81528   38%     1631   16% /wpars/titian
/dev/fs1v01     131072          128312   3%         5    1% /wpars/titian/home
/opt            1572864         254888   84%     7510    4% /wpars/titian/opt
/proc          -                -         -         -        - /wpars/titian/proc
/dev/fs1v02     262144          256832   3%         12    1% /wpars/titian/tmp
/usr            3801088         646624   83%    32033    7% /wpars/titian/usr
/dev/fs1v03     262144          229496  13%     1216    5% /wpars/titian/var
```

2.6 WPAR mobility

Live Application Mobility is the newest virtualization technology from IBM. This is a software approach that enhances the current line of technology. Live Application Mobility is a complement to the IBM line of virtualization packages. Live Application Mobility allows for planned migrations of workloads from one system to another without interrupting the application. You can use this technology to perform a planned firmware installation on the server. Most

workloads do not need to be aware of the WPAR relocation. But, we always recommend proper planning and testing before moving anything into a production environment.

WPAR mobility, also referred to as *relocation*, applies to both types of WPARs: application and system. The relocation of a WPAR consists of moving its executable code from one LPAR to another one while keeping the application data on the same storage devices. It is therefore mandatory that these storage devices are accessible from both the source and target LPARs hosting the WPAR.

In the initial version of AIX V6, NFS provides this dual access to the storage area. As mentioned previously, the hosting global environment hides the physical and logical device implementations from the hosted WPARs. The WPAR only works with data storage at the file system level. All files that need to be written by the application must be hosted on an NFS file system. All other files, including the AIX operating system files, can be stored in file systems local to the hosting global environment. Table 2-1 helps you plan the creation of the file systems for an application that requires WPAR mobility, when hosted in an application or system workload partition, for an application that only writes in file systems dedicated to the application. Cells in bold font explain the high level difference.

Table 2-1 Default file system location to enable partition mobility

File system	Application WPAR	System WPAR
/	Global environment	NFS-mounted
/tmp	Global environment	NFS-mounted
/home	Global environment	NFS-mounted
/var	Global environment	NFS-mounted
/usr	Global environment	Global environment
/opt	Global environment	Global environment
Application specific	NFS-mounted	NFS-mounted

Figure 2-5 on page 33 shows an example of a complete environment in which to deploy LPARs and WPARs on two p595 systems.

The first global environment is called *saturn* and is hosted in an LPAR on the first p595. It is a client of the NFS server, as well as is *titan*, the system WPAR inside of it. The second system is also a p595, but it can be any of the same class of systems from the p505 or higher. One of its LPARs hosts a global environment called *jupiter*, which is also a client of the NFS server.

There a utility server, which in this example, is a p550. On this system, there is an NFS server, a NIM server, and a WPAR Manager for AIX to provide the single management point needed for all the WPARs. The NIM server represents how to load AIX images into the frame, which can have a large number of LPARs. The NFS server provides an external file system solution to the WPARs and provides the vehicle to move them dynamically from one system to another without disrupting the application.

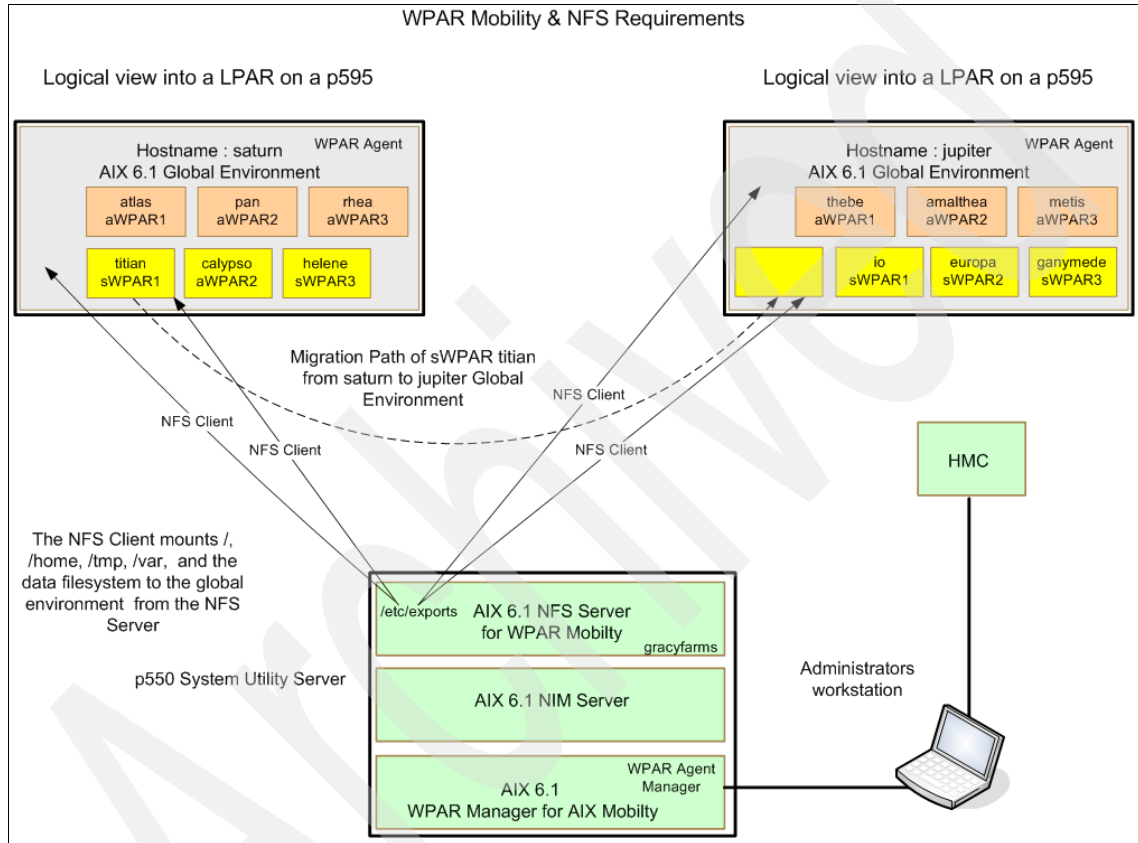


Figure 2-5 Overview of the topology requirements in a mobile WPAR solution

The NFS server is a standard configuration and utilizes either NFS protocol Version 3 or Version 4. You can use command line editing or SMIT to configure the /etc/exports.

Figure 2-6 on page 35 represents the relationship among the different views of the same file systems as seen:

- ▶ From the NFS server where they are physically located
- ▶ From the global environments on which they are NFS-mounted
- ▶ From the system WPAR that uses them

In the WPAR, the /opt, /proc, and /usr are set up as namefs with read-only permissions (exception: /proc is always read-write) mapping on the global environments /opt, /proc, and /usr. The rest of the file systems (/ , /home, /tmp, and /var) are set up as standard NFS. The /etc/exports file on the NFS server must have permissions set for both the global environment (jupiter) and system WPAR (ganymede) for the mobility to work.

Important: The NFS server must provide access to both the global environment and the WPAR in order for the WPAR to work at all. In a mobility scenario, access must be provided to the WPAR and all global environments to which the WPAR might be moved. Furthermore, any time that / , /var, /usr, or /opt are configured as NFS mounts, the NFS server must provide root access (for example, via the `-r` option to `mkntfsxp`) to all of the relevant host names.

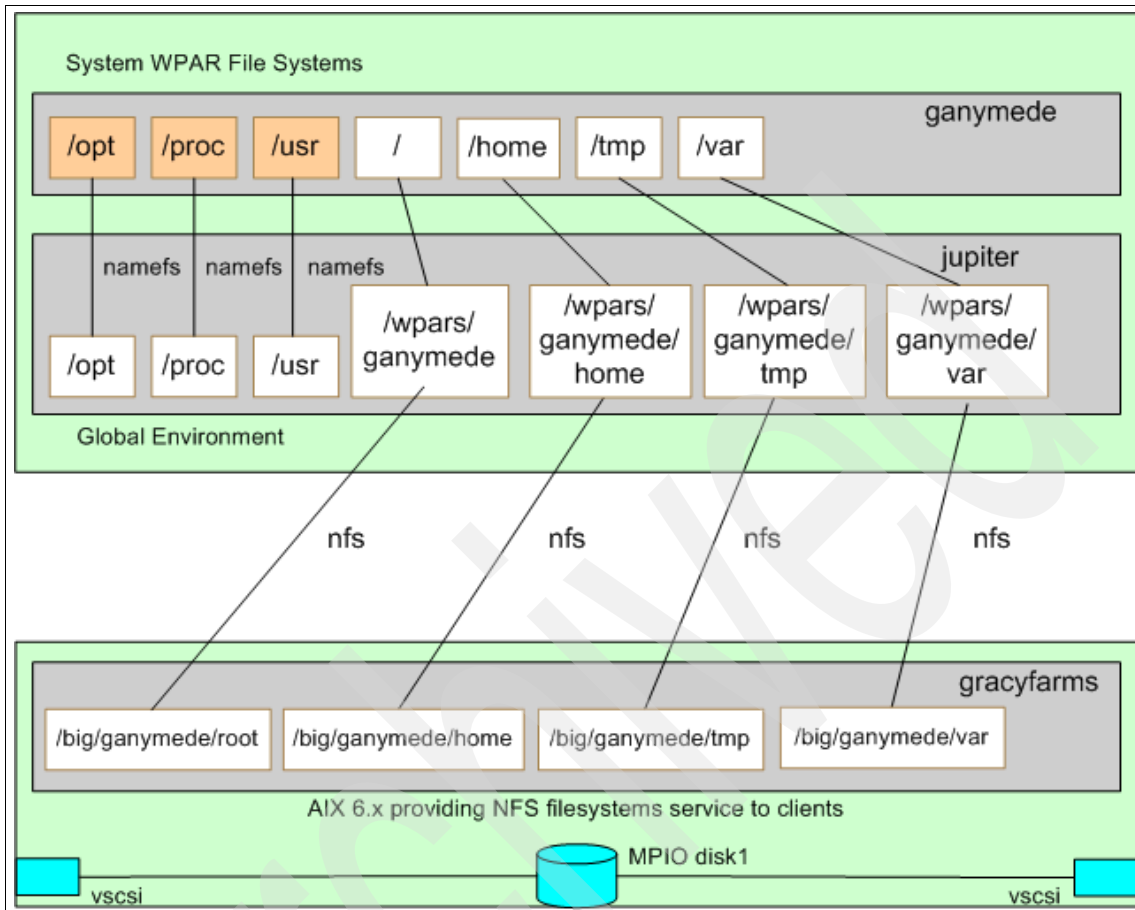


Figure 2-6 File systems from the NFS for a mobile system WPAR

Using the **df** command as in Example 2-5 shows that the global environment **jupiter** has its own file systems hosted on locally attached disks, as well as NFS file systems mounted from the **gracyfarms** NFS server, for use by the **ganymede** system **WPAR**.

Example 2-5 NFS server mount points for ganymede WPAR

```

root: jupiter:/wpars/ganymede --> df
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
/dev/hd4         131072          66376  50%     1858   6% /
/dev/hd2         3801088         646624  83%    32033  7% /usr
/dev/hd9var      524288          155432  71%     4933   8% /var
/dev/hd3         917504          233904  75%      476   1% /tmp
/dev/hd1         2621440         2145648 19%       263   1% /home
/proc            -                -      -       -     - /proc
/dev/hd10opt     1572864         254888  84%     7510  4% /opt
gracyfarms:/big/ganymede/root 131072      81528  38%     1631  16% /wpars/ganymede
gracyfarms:/big/ganymede/home 131072     128312  3%        5   1% /wpars/ganymede/home
/opt            1572864         254888  84%     7510  4% /wpars/ganymede/opt
/proc           -                -      -       -     - /wpars/ganymede/proc
gracyfarms:/big/ganymede/tmp 262144      256832  3%       12   1% /wpars/ganymede/tmp
/usr            3801088         646624  83%    32033  7% /wpars/ganymede/usr
gracyfarms:/big/ganymede/var 262144      229496 13%     1216   5% /wpars/ganymede/var

```

2.7 WPAR and LPAR comparison

IBM has taken a leadership role in innovation over the past fourteen years and has been number one in the patent technology race. Out of this has come a plethora of new and innovative products. In 2001, IBM announced the LPAR feature on IBM pSeries, and then in 2004, Advanced Power Virtualization provided the micropartitioning feature. In 2007, IBM announces WPAR mobility.

WPARs are not a replacement for LPARs. These technologies are both key components of the IBM virtualization strategy. The technologies are complementary and can be used together to extend their individual value.

Providing both LPAR and WPAR technology offers a broad range of virtualization choices to meet the ever changing needs in the IT world. Table 2-2 on page 37 compares and contrasts the benefits of the technologies.

Table 2-2 Comparing WPAR and LPAR

Workload partitions	Logical partitions
Share OS images	Execute OS images
Finer-grained resource management, per workload	Resource management per LPAR Capacity on demand.
Security isolation	Stronger security isolation
Easily shared files and applications	Support multiple OSs, tunable to applications
Lower administrative costs: <ul style="list-style-type: none"> ▶ One OS to manage ▶ Easy create, destroy, and configure ▶ Integrated management tools 	OS fault isolation

Important: When considering the information in Table 2-2, keep in mind the following guidelines:

- ▶ In general, when compared to WPARs, LPARs will provide greater flexibility in supporting your system virtualization strategies. After you have designed an optimal LPAR resource strategy, then within that strategy you design your WPAR strategy to further optimize your overall system virtualization strategy in support of AIX6 applications. See Figure 2-7 for an example of this strategy where multiple LPARs are defined to support different OS and application hosting requirements, while a subset of those LPARs running AIX6 are set up specifically to provide a global environment for hosting WPARs.
- ▶ Because LPAR provisioning is hardware and firmware-based, consider LPARs as a more secure starting point for meeting system isolation requirements than WPARs.

Figure 2-7 shows how LPARs and WPARs can be combined within the same physical server, which also hosts the WPAR Manager and NFS Server required to support partition mobility.

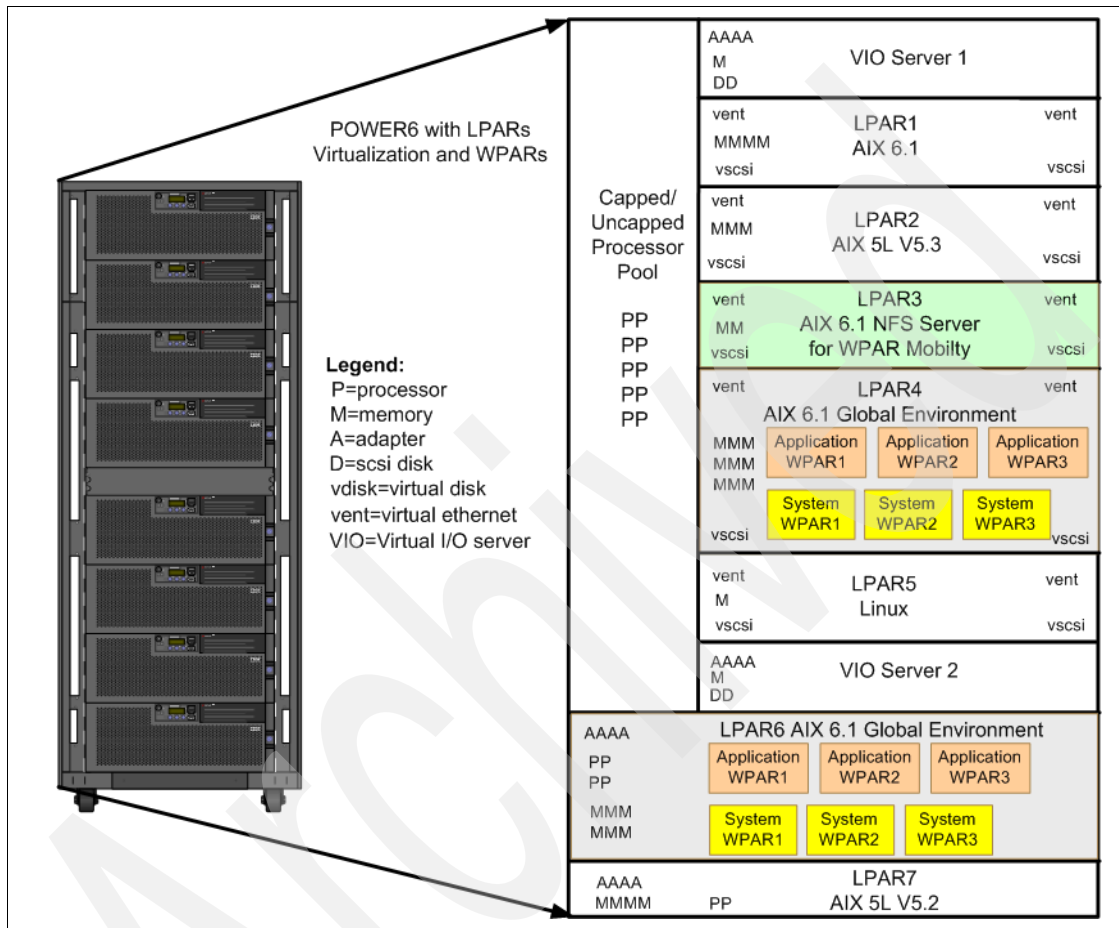


Figure 2-7 Overview of a POWER6-based server implementation

Using WPARs: management and operation

This part of the book details the various components of the WPAR technology, and how they relate to other AIX concepts. In this part, we present example scenarios that take advantage of WPARs in conjunction with other AIX and System p features. This part of the book is intended to be read by system architects, and system administrators looking for walk-through examples of WPAR deployment scenarios. This part contains the following chapters:

- ▶ Chapter 3, “Overview of WPAR management tools” on page 41
- ▶ Chapter 4, “State management” on page 55
- ▶ Chapter 5, “System administration tasks” on page 61
- ▶ Chapter 6, “IBM WPAR Manager for AIX” on page 125
- ▶ Chapter 7, “Resource control” on page 237
- ▶ Chapter 8, “Device management” on page 255

Archived

Overview of WPAR management tools

Workload partitions (WPARs) can be managed in many different ways, through a command-line interface, graphical user interfaces, base AIX commands, or additional program products. This chapter provides an overview of all commands and tools that can be used to manage WPARs and explains in which context they can be used.

This chapter introduces general technical information that you must understand as a prerequisite to reading the other chapters in Part 2, “Using WPARs: management and operation” ; this chapter includes:

- ▶ 3.1, “Comparison of application and system WPARs” on page 42
- ▶ 3.2, “Overview of tools” on page 44
- ▶ 3.3, “AIX command-line interface” on page 45
- ▶ 3.4, “WPAR Manager interface” on page 48
- ▶ 3.5, “Workload Manager (WLM) considerations” on page 49
- ▶ 3.6, “WPAR description database” on page 50

3.1 Comparison of application and system WPARs

Before administering WPARs, it is important to understand all of the differences between the types of WPARs: application workload partition and system workload partition. This first section presents a side by side comparison of these types of workload partitions. Table 3-1 summarizes the differences between the types of WPARs. Cells in bold font explain the high-level difference, and the indented cells provide more detail about that characteristic.

Table 3-1 Side by side comparison of application and system WPARs

Application workload partition	System workload partition
Is used to isolate an individual service or application	Provides a complete virtualized OS environment to host multiple processes, services, and applications
Provides a runtime environment for executing either a single application or a group of related applications. It does not contain running AIX services or daemons normally present in a complete instance of AIX.	Provides a complete running instance of AIX, including system init and other AIX services and daemons
A set of applications or processes are started from a single command or script	Much like a typical AIX environment, applications can be started in many ways: cli, remote command, crontab, and so forth
Lightweight, requiring only one process plus the application itself	Potentially consisting of a large number of processes
Can be created and started in a matter of seconds within the same action	Created and started in two distinct phases. Takes longer to start, stop, checkpoint, and so forth
Transient environment	Persistent environment
Started as soon as an application is started and is destroyed as soon as all the processes in the hosted application exit.	The wpar can be booted, shut down, and rebooted.
If startwpar fails, the WPAR is deleted.	If startwpar fails, the WPAR returns to the Defined state.
The command lswpar can only list active application partitions.	The command lswpar lists all system partitions independent of their state.

Application workload partition	System workload partition
Uses global resources	Owns dedicated resources
Uses the users and groups defined in the global environment	Has its own set of users and groups
Shares the file system namespace of the global environment	Has its own writable file systems and only shares read-only file systems with the global environment and other workload partitions
Has a vinit process, with PID 1, which performs no actions but stands as the parent process of the other processes executing within the WPAR	Has its own init process, which spawns other AIX environment processes and daemons
Provides limited user access	Supports full user access services
Cannot be remotely "logged in" (telnet, rsh, and so forth)	Supports remote login: telnet, rsh, rlogin, and ssh, depending on daemons and services executing within the workload partitions
Does not provide RBAC service specific to the workload partition	Integrated with RBAC to provide granular privilege and security control within the context of the WPAR

The following items emphasize similarities between application and system workload partitions:

- ▶ Application and system workload partitions can only be created and administered by the root user or a user with the RBAC `aix.wpar` authorization.
- ▶ Each WPAR can have its own isolated network environment with IP addresses and a unique host name.
- ▶ Both application and system workload partitions can be created using a predefined specification file rather than using options and arguments of the `mkwpar` command.
- ▶ Checkpoint and mobility features are supported by both application and system workload partitions.
- ▶ Application and system workload partitions support resource control (Workload Manager (WLM)), so that they can be assigned specific fractions of CPU and memory.
- ▶ Application and system workload partitions can be accessed from the global partition using the `clogin` command.

- ▶ Both system and application WPARs can provide accounting information for each WPAR

3.2 Overview of tools

WPAR functions are included as part of the base AIX6 offering. Note in Table 3-2 that certain tooling features require an additional product: IBM Workload Partitions Manager for AIX.

Table 3-2 WPAR management options

Tool or function	Part of	Usage
AIX command-line interface	AIX Base	Creation, activation, modification, and deletion of WPARs
SMIT/smitty	AIX Base	Identical to CLI usage
WLM	AIX Base	WLM provides the underlying technology for WPAR resource management, but is not directly used by system administrators to manage WPARs
WPAR checkpoint and relocation command-line interface	IBM Workload Partitions Manager for AIX	A checkpoint of the runtime status of a WPAR that can be used to resume a workload at a specific point of its execution and optionally move it to a different server
WPAR Manager GUI	IBM Workload Partitions Manager for AIX	Automation of WPAR relocation, load balancing, metering, inventory, performance data collection, and policy-based mobility

The WPAR features, which are part of AIX, have a scope that is limited to the LPAR in which the AIX instance is executing.

The WPAR Manager additional program product provides management of multiple WPARs executing on a set of LPARs that can be executing on different physical servers.

The WPAR built-in AIX features are provided by the fileset bos.wpars.

The WPAR Manager additional program consists of the filesets:

mcr.rte	The support for WPAR mobility
wparmgt.agent.rte	The WPAR agent executing in all LPARs containing managed WPARs
wparmgt.mgr.rte	The WPAR Manager executing in the management LPAR
wparmgt.cas.agent	The Common Access Service agent executing in all LPARs containing managed WPARs
wparmgt.cas.agentmgr	The Common Access Service agent executing in the management LPAR
Tivoli@tivguid	The Graphical User Interface
lwi.rte	Eclipse-based LightWeight Infrastructure (LWI) runtime

There is no fileset providing the WPAR Agent Console role. The console can be provided by any Web browser executing on any workstation with an IP connection to the WPAR Manager.

3.3 AIX command-line interface

To support WPARs, the AIX CLI has been extended:

- ▶ New commands have been added.
- ▶ Certain existing commands have been modified.

Some of these commands can be used from the global environment hosting the WPARs and some within WPARs themselves. Some commands apply to system WPARs, while others apply to application WPARs.

Table 3-3 on page 46 lists all of the new commands of the AIX CLI that are specific to WPAR management and describes the environment in which they can be used. Commands are grouped by the type of system administration activity. Details about the commands are in Chapter 5, “System administration tasks” on page 61.

For each of the commands in Table 3-3, a SMIT panel exists that provides the same functionality. The starting point to access the SMIT panels depends on the type of partition to manage: `smitty manage_syswpar` or `smitty manage_appwpar`.

Table 3-3 Summary of all AIX CLI WPAR-related commands

	Applies to	Executed from
Partition instance management		
<code>lspar</code>	Application and system WPARs	Global environment
<code>mkwpar</code>	System WPARs only	Global environment
<code>startwpar</code>	System WPARs only	Global environment
<code>stopwpar</code>	Application and system WPARs	Global environment
<code>wparexec</code>	Application WPARs only	Global environment
<code>chwpar</code>	Both	Global environment
<code>rebootwpar</code>	System WPARs only	Global environment
<code>rmwpar</code>	Application and system WPARs	Global environment
<code>rc.wpars</code>	System WPARs only	Global environment
Backup and restore		
<code>savewpar</code>	System WPARs only	Global environment
<code>restwpar</code>	WPAR backup media	Global environment
<code>restwparfiles</code>	WPAR backup media	Global environment
<code>mkwpardata</code>	WPAR backup media	Global environment
<code>lssavewpar</code>	WPAR backup media	Global environment
Software management	Only applies to system workload partitions	
<code>syncwpar</code>	System WPARs only	Global environment
<code>syncroot</code>	System WPARs only	Within System WPAR only

	Applies to	Executed from
Partition access		
clogin	Application and system WPARs	Global environment
telnet	System WPARs only ^a	Anywhere in a network with connectivity to the WPAR
ssh	System WPARs only ^a	Anywhere in a network with connectivity to the WPAR
rsh	System WPARs only ^a	Anywhere in a network with connectivity to the WPAR

a. Application WPARs too, provided you can get the daemon running in the application WPAR to enable this access mode.

In addition, commands that existed in AIX prior to the availability of workload partitions have been enhanced to support this new feature. As a general rule, the @ character is used as the new flag of existing commands for requesting WPAR specific information. For example, `vmstat -@ ALL` reports a summary of virtual memory information for all workload partitions.

Most modified commands are listed in the *AIX Workload Partitions Administration Guide*. Also, a copy of this list is provided in Appendix A, “Modified AIX commands” on page 263.

Table 3-4 provides additional information about modified commands.

Table 3-4 Modified commands

Command	Applies to	New behavior
shutdown	Application and system WPARs	Shuts down the WPAR. No impact on the global environment
halt	Application and system WPARs	Shuts down the WPAR. No impact on the global environment
reboot	Application and system WPARs	When applied to a system WPAR, restarts the WPAR. When applied to an application WPAR, stops the applications, kills the WPAR, but does not restart it
topas	Global environment	Displays new fields, mentioning the number of executing WPARs
uptime	System WPARs only	Displays uptime information from the moment the WPAR was started

3.4 WPAR Manager interface

The WPAR Manager provides four commands as part of its command-line interface (CLI):

- ▶ **chkptwpar**
- ▶ **killwpar**
- ▶ **restartwpar**
- ▶ **resumewpar**

The WPAR Manager also provides one utility (**mcrk_admin** command) to load and unload the kernel extension.

These four commands are to be used from the global environment and they only apply to the workload partitions that are hosted within this global environment. They apply to both application and system WPARs that have been created with the checkpoint ability option (-c) specified.

WPAR management functions that provide remote workload partition management are only available from the graphical user interface.

The WPAR Manager CLI is described in more detail in 6.12, “Introduction to mobility command-line interface” on page 230, and the GUI is described in Chapter 6, “IBM WPAR Manager for AIX” on page 125.

3.5 Workload Manager (WLM) considerations

WPAR management provides support for control of resource allocation to each WPAR. WLM, the workload management part of the AIX operating system, provides the underlying technology that supports WPAR resource controls.

However, the use of WLM is hidden to the system administrator, who does not directly use WLM commands to manage WPARs. The control of WPAR resources is provided through the optional flags and arguments (**-R**) of WPAR commands (**mkwpar**, **chwpar**, **wparexec**, and **lswpar**).

Note: The system administrator must not use WLM commands to control resources associated to workload partitions.

The exceptions to this rule are the **wlmstat** and **mkrset** commands. The **wlmstat** command has been expanded with an **-@** flag to display WPAR specific information. The **mkrset** command can be used to create resource sets that are then passed as an argument to the WPAR commands.

Important: Resource management is available in the global environment so that the system administrator can balance the allocation of AIX resources among the various WPARs that compete for these shared resources. WPAR resource management commands, as well as WLM commands, are not available within one WPAR.

Details of resource management for WPAR are provided in Chapter 7, “Resource control” on page 237.

3.6 WPAR description database

When a workload partition is created, a configuration profile for that partition is stored in the workload partition database. The purpose of this section is to explain how the AIX system uses the WPAR database to store information about workload partitions.

3.6.1 WPAR representation

WPARs, like most UNIX concepts, are described in files. A system administrator needs to be familiar with the three groups of WPAR description files that exist in an AIX instance:

- ▶ The definition of the partition is used by the operating system. It is stored in the WPAR database. The system administrator normally accesses the content of this database by using operating system commands, such as `lswpar`. The files holding the partition definition are described in 3.6.2, “WPAR definition” on page 50.
- ▶ The specification of a WPAR is used by a system administrator to create a partition. Specification files are described in 3.6.3, “WPAR specification” on page 52.
- ▶ The “checkpoint state” of a WPAR is used by the WPAR mobility and relocation feature to hold all of the information that is needed by the operating system to restart a paused WPAR in the same or in a different LPAR, so it can resume execution of all of its processes at the exact point where they were paused. This information is only used by the operating system and is not for direct handling by a system administrator. Checkpoint statefiles are described in 3.6.4, “WPAR checkpoint state” on page 53.

3.6.2 WPAR definition

The definition of the defined partition is stored in the *WPAR database*. This database consists of files stored in the `/etc/wpars` directory, as well as an extra stanza added to the `/etc/filesystems` of the global environment.

Important: The implementation of the WPAR database is subject to change in future versions of AIX. This section is intended to help the system administrator understand the pieces of information used to describe WPARs. However, the system administrator never needs to directly access the information contained in the files described here.

A system WPAR that has been defined but is not active or running is represented in:

- ▶ An index file (*/etc/wpars/index*)
- ▶ A configuration file (*/etc/wpars/wparname.cf*)
- ▶ Stanzas in */etc/filesystems*

The index file contains information regarding the name, the persistent and configured identifiers of all WPARs known to the system, and all defined system partitions, as well as all executing application partitions. It also contains the WPAR type (system or application).

The configuration file and */etc/filesystems* contain the information required to start a WPAR.

For each system, the configuration file contains the full privilege and device sets for the WPAR and all information provided by the system administrator when creating the WPAR, except the file system-related information.

The global environment */etc/filesystem* contains stanzas describing how file systems of the global environment are mapped to file systems of the WPAR. Note that */etc/filesystems* is only used for information related to system WPARs. Application WPARs do not need specific file system information in order to execute.

Because the WPAR database consists of files stored in the global environment *rootvg*, it is saved during any backup of the global partition (for example, during a **mksysb** operation) and can be restored just as any other set of files.

But if the data and file systems for the WPARs do not match the configuration information contained in the WPAR database, then results can be unexpected.

The best way to recover a WPAR is to use the **restwpar** commands to restore a **savewpar** backup image. And the best way to save the definition of an existing WPAR is to create a specification file from the content of the WPAR database (See 3.6.3, “WPAR specification” on page 52).

WPAR backup and restore considerations are further described in 5.4, “Backup, restore, and cloning” on page 102.

3.6.3 WPAR specification

WPAR specification files are optional.

AIX provides a system administrator with a great many configuration choices for defining a workload partition. When you use most of the default settings, the **mkwpar** command or smitty requires very little input from the system administrator. However, the use of the **mkwpar** command or smitty can be cumbersome when the system administrator needs to change many of the default settings or wants to create a large number of WPARs. In these cases, specification files can speed up the system administrator WPAR creation activity.

A *specification file* can be used to create a WPAR by storing in a file with a predefined format all of the information that otherwise must be typed by the system administrator in the **mkwpar** command or the SMIT panels. A specification file is a text file that can be edited using any AIX text editor.

A sample specification file is provided in `/usr/samples/wpars/sample.spec`. This sample file contains examples of configuration stanzas, as well as their detailed descriptions.

When a WPAR has already been defined, the system administrator can create a specification file from the existing WPAR using **mkwpar -e existingWPAR -w -o filename**. It is then possible to create clones of this existing WPAR by renaming and editing this specification file. This method applies to system WPARs only.

The WPAR can then be created by passing the specification filename as an argument to the **-f** flag of the **mkpwar** or **wparexec** commands.

There are two other files that are also involved in creating a WPAR:

- ▶ The `/etc/wpars/secattrs` file describes the default privileges for the WPARs that are being created.
- ▶ The `/etc/wpars/devexports` file describes the default device handling for the WPARs that are being created.

The user can specify alternate paths to files of the same format via command line options. Note that the two files only apply to system workload partitions, not to application workload partitions.

Note: The specification file and the two default files (`secattrs` and `devexports`) are only used while creating the WPAR. Subsequent changes to these files will not affect any already created WPARs that use them.

Specification files can be stored anywhere within the AIX directory structure. Because the `/etc/wpars` directory organization is subject to change and must not be directly used by the system administrator, a good practice is to create a specific WPAR administration directory where all specification files are located, which can be easily backed up.

3.6.4 WPAR checkpoint state

The files in the WPAR database and the specification files describe the static configuration of a workload partition. To enable pausing and relocating WPARs, AIX needs to capture a dynamic representation of the partition. This dynamic representation is stored in a checkpoint statefile.

Note: Just as with the WPAR database, the implementation of the checkpoint statefile is subject to changes in future versions of AIX.

For an application WPAR, the information contained in the specification file is also saved within the statefile so that the `restartwpar` command can recreate the WPAR.

For a system WPAR, the WPAR must be in the Defined state before it can be restarted whether it is on the LPAR where it was running when checkpointed, or whether it is on a different LPAR. The `restartwpar` command therefore does not need the specification file information, which is not included in the checkpoint statefile.

The current implementation of statefile is a subdirectory containing many files. None of these files are to be directly used by the system administrator. The only actions that a system administrator performs on statefiles are to:

- ▶ Create them as an empty directory on an Network file system-mounted file system.
- ▶ Pass the statefile's name as an argument to the `chkptwpar` and `restartwpar` commands.
- ▶ Delete them after the WPAR restart is complete.

More details about WPAR checkpoint and relocation are provided in Chapter 6, "IBM WPAR Manager for AIX" on page 125.

3.6.5 Other global environment considerations

The global environment on which workload partitions are created is a standard AIX instance. There is no special prerequisite on an AIX 6 instance to host workload partitions. The only limitation is that, for system WPARs only, the operating system must not use the Trusted Computing Base feature.

State management

Workload Partitions (WPARs) are objects that can be defined, activated, stopped, and relocated. The status of WPARs can be represented by state machines. This chapter describes these states and explain what commands can make a WPAR change states. This chapter contains the following sections:

- ▶ 4.1, “Overview of WPAR states” on page 56
- ▶ 4.2, “State transitions” on page 57

4.1 Overview of WPAR states

WPARs can be created as static or as a candidate for checkpointing and relocation, depending on whether the `-c` flag is used in the `mkwpar` or `wparexec` command when creating the partition. The number of states that a partition can reach depends on whether it was created as being capable of the checkpoint state.

There are seven states that a WPAR can reach, as presented in Table 4-1.

Table 4-1 WPAR states

State	Displays as	Static partition		Mobility-enabled partition	
		Application	System	Application	System
Defined	D	I	✓	I	✓
Active	A	✓	✓	✓	✓
Paused	P	N/A	N/A	✓	✓
Broken	B	✓	✓	✓	✓
Transitional	T	✓	✓	✓	✓
Frozen	F	N/A	N/A	I	I
Loaded	L	I	I	I	I

For each state listed in the first column, the second column indicates how the state is presented in the output of the `lswpar` command. A check sign (✓) in columns three through six indicates a state that is normally reachable during the life cycle of a WPAR. The letter I (I) in the same columns indicates an internal state that is not visible to a system administrator under normal circumstances.

The definitions for these states are:

Defined

Applies mostly to system WPARs. This is the state of a system WPAR, which has been created but has not yet started the execution of any process.

This state also exists for an application partition, but as an internal state that also does not display in normal conditions.

Active

This is the normal state of either an application or system partition while executing the workload.

Paused	This state only applies to mobile partitions (created with the <code>-c</code> flag). This state is reached when a WPAR has had a successful checkpoint or restore state, at which time it is ready to be either resumed or ended.
Broken	This state is not visible to the system administrator under normal circumstances. A partition only reaches this state when a failure occurs, and the recovery process fails.
Transitional	“Transitional” is not a state in a strict sense of the word. The <code>lswpar</code> command displays a “T” while a WPAR is in the process of changing from one state to another.
Frozen	This state is mostly used in the case of the High Performance Computing (HPC) or distributed type of application. It allows you to check that all parts of the application are ready to be checkpointed without effectively performing the checkpoint. This state is rarely viewed by system administrators of non-HPC applications.
Loaded	The loaded state applies to mobile partitions and is visible on the target global environment after a WPAR relocation. This state is mostly internal to the life cycle of a partition. It is rarely viewed by system administrators under normal circumstances.

4.2 State transitions

Figure 4-1 on page 58 and Figure 4-2 on page 60 present graphical overviews of WPAR state-to-state transitions for system and application partitions.

The state transitions are slightly different for application workload partitions and system workload partitions, because the application partition does not reach the *Defined* state under normal circumstances.

These diagrams require detailed explanations:

- ▶ The figures present the states reachable by a WPAR as boxes with rounded corners, as well as the commands that lead to transitions from one state to the other. The commands are presented as labels on top of arrows between the boxes.

- ▶ The states listed in 4.1, “Overview of WPAR states” on page 56 are represented by a box with:

- A solid border for common states:



- A dashed border for internal states:

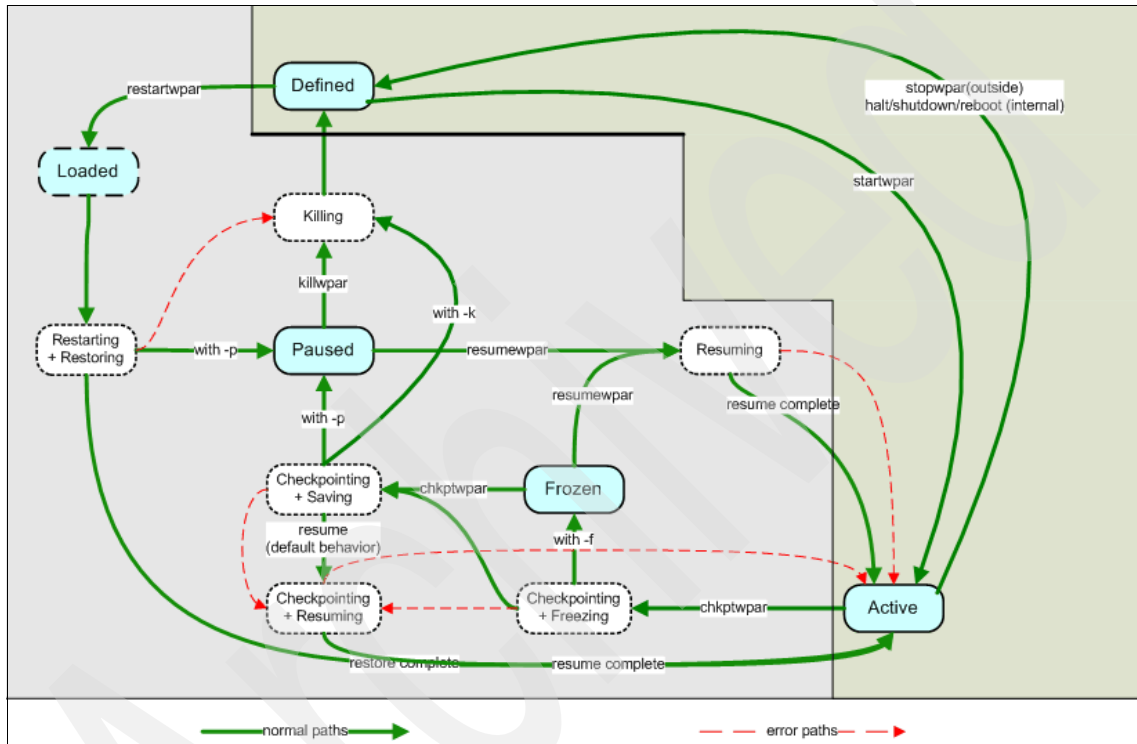


Figure 4-1 System WPAR state transitions

- ▶ Internal states also exist that are not visible to the user with the **lswp** command. Only a subset of these states is presented in Figure 4-1 and Figure 4-2 on page 60. The internal states are presented in boxes with a dashed border and a white background. When in these states, a WPAR is displayed as *Transitional* by the **lswp** command.

These states are presented for the purpose of completeness and so that you can understand the transitions that result from using the various options of the WPAR management commands presented in Figure 4-1.

Note: The descriptions of WPAR internal states are valid for the implementation of the WPAR technology at the time this book was written. This implementation is subject to changes in future releases.

- ▶ The transitions represented by a solid thick line represent normal state changes. Transitions represented by dotted thin lines correspond to exceptions and error recovery paths. The error recovery objective is to get the WPAR back to the state from which it originally came. For example, the error recovery for checkpoint will try to set WPAR back to active, but for restart it will kill the WPAR to get it back to a defined state.
- ▶ There is an implicit error path from every state to the Defined state, which is achieved by calling **stopwpar -F**, which will force the stopping of a WPAR.
- ▶ Within transition labels, *inside* means that a command has been initiated within a WPAR, while *outside* indicates the command was called from the global environment.
- ▶ The upper right area of the diagrams with a light gray background presents the states and transitions that are available to all WPARs. The lower left part with a darker background shows the states and transitions that are specific to mobile partitions.

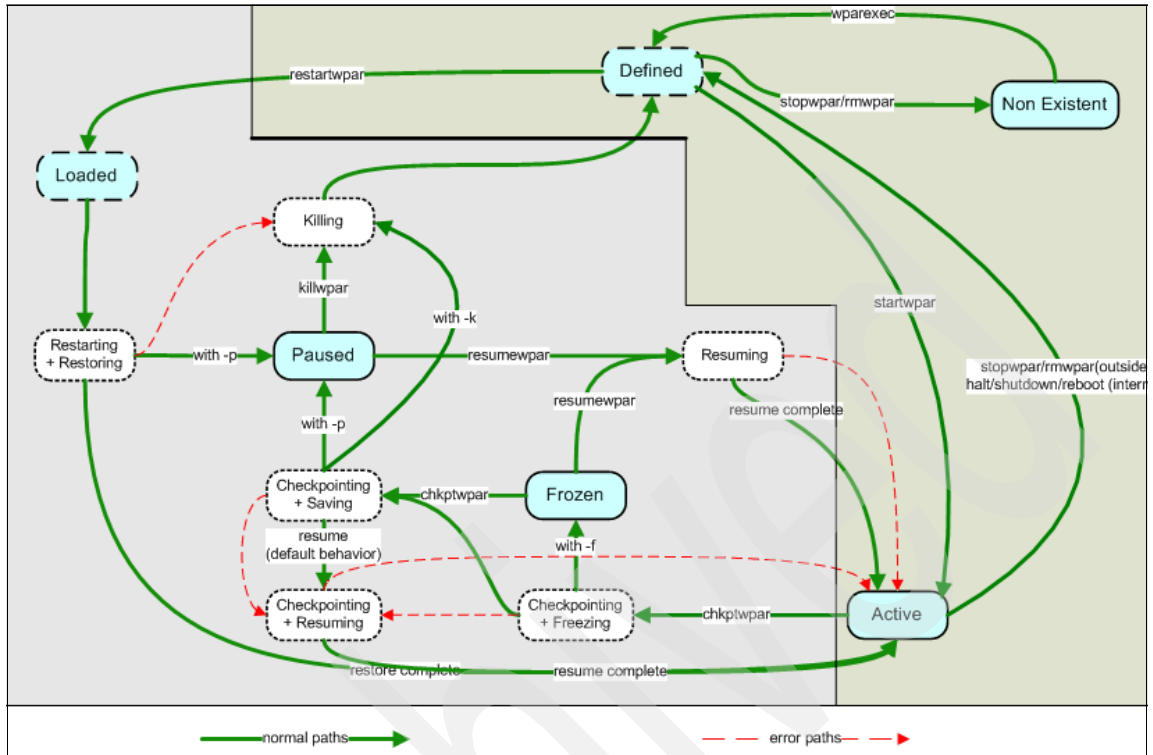


Figure 4-2 Application WPAR state transition

Figure 4-2 contains an extra *state* shown as *Non-existent* to represent the fact that an application workload partition is created from scratch by the `wparexec` command and is destroyed after the completion of the processes that it contains.

System administration tasks

AIX provides support for workload partitions using the command-line interface (CLI) and SMIT. In addition, the Workload Partition Manager (WPARMGR) provides a WPAR management GUI and allows for management of WPAR mobility features. This chapter focuses on the AIX base support. It demonstrates small tasks in a step-by-step approach. You can use this information to build hands-on examples and gain necessary experience handling WPARs. This chapter contains the following sections:

- ▶ 5.1, “Workload partition administration” on page 62
- ▶ 5.2, “Software installation and maintenance” on page 86
- ▶ 5.3, “File system considerations” on page 95
- ▶ 5.4, “Backup, restore, and cloning” on page 102
- ▶ 5.5, “Users and groups” on page 111
- ▶ 5.6, “Print spooling considerations” on page 115
- ▶ 5.7, “System environments” on page 116
- ▶ 5.8, “Processes and subsystems” on page 117
- ▶ 5.9, “Performance and resource schedule” on page 120

For further explanation about how to administer mobile WPARs with the command-line interface, go to 6.12, “Introduction to mobility command-line interface” on page 230 or go to Chapter 6, “IBM WPAR Manager for AIX” on page 125 if you plan to use the graphical user interface.

5.1 Workload partition administration

In this section, we show how to administer a workload partition without the mobility features.

5.1.1 Create a simple WPAR

Workload partitions are divided into system and application WPARs. The commands to create each type of WPAR are different. For system WPARs, we use **mkwpar**, which effectively creates a new virtual AIX. For application WPARs, we execute the application within the WPAR by issuing the **wparexec** command.

System WPAR

The creation of a system workload partition is very simple. AIX provides a default value for each of the available options of the **mkwpar** command. To create a new WPAR, you only supply the name of the partition:

```
# mkwpar -n MySysWpar
```

First, the operating system creates and mounts the WPAR's file systems. Then, it populates them with the necessary system files. And finally, it synchronizes the root part of the installed software. Example 5-1 shows the output of the creation of a simple system WPAR (for clarity purposes, we do not show all of the lines).

Example 5-1 Create a simple system workload partition

```
mkwpar: Creating filesystems...
/
/home
/opt
/proc
/tmp
/usr
/var
populate: Mounting all workload partition file systems
x ./usr
x ./lib
x ./audit
x ./dev
x ./etc
...
x ./etc/security/certificates/certificate_540
x ./home
x ./home/guest
```



```

x ./home/lost+found
syncroot: Processing root part installation status.
syncroot: Synchronizing installp software.
+-----+
                Pre-installation Verification...
+-----+
Verifying selections...done
Verifying requisites...done
Results...

SUCSESSES
-----
Filesets listed in this section passed pre-installation verification
and will be installed.

Selected Filesets
-----
Java14.sdk 1.4.2.125                # Java SDK 32-bit
Java5.sdk 5.0.0.75                 # Java SDK 32-bit
Java5_64.sdk 5.0.0.100             # Java SDK 64-bit
...
mkwpar: Workload partition MySysWpar created successfully.
To start the workload partition, execute the following as root: startwpar [-v]
'MySysWpar'

```

You can check the state of this system WPAR with **lswpar** as shown in Example 5-2.

Example 5-2 List WPARs

```

# lswpar
Name      State  Type  Hostname  Directory
-----
MySysWpar D      S     MySysWpar /wpar/MySysWpar

```

The “D” under “State” means that this system partition is not active; it is in the defined state. The “S” stands for System WPAR.

It is also possible to use SMIT to create a simple system partition:

```
# smit simplewpar_sys
```

Figure 5-1 on page 64 presents an example of a SMIT window.

```

                                Create a System Workload Partition

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

* Workload Partition Name      [Entry Fields]
                               [MySysWpar]
HOSTNAME                       []
Base Directory                 []
Default Volume group name     []      +
Copy Global Name Resolution Settings  no      +
Network INTERFACE              []      +
    Internet ADDRESS (dotted decimal) []
    Network MASK                []
Create the Workload Partition?  yes      +
Specification Output File      []      /
START Workload Partition?     no      +

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

```

Figure 5-1 Simple system WPAR creation using SMIT

Application WPAR

The creation of an application WPAR is also very simple, because the only mandatory parameter is the full path of the executable file to run inside the WPAR. Example 5-3 shows the `wparexec` command starting an application WPAR immediately after creation. This type of WPAR only exists while the application is running. When the application ends, the WPAR also ends and all of its resources are freed. If the application WPAR has a dependency on a file system that is not mounted, it will mount the file system automatically.

Example 5-3 Create a simple application WPAR

```

# wparexec /tmp/MyApp
wparexec: Verifying filesystems...
wparexec: Workload partition MyApp created successfully.
startwpar: COMMAND START, ARGS: MyApp
startwpar: Starting workload partition 'MyApp'
startwpar: Mounting all workload partition file systems
startwpar: Loading workload partition
Weather forecast calculation started...
Done
startwpar: Shutting down all workload partition processes

```

```

rmwpar: Removing workload partition MyApp
rmwpar: Return Status = SUCCESS
startwpar: Return Status = SUCCESS

```

If you do not provide the name of the WPAR, the **wparexec** command gets it from the basename of the application. In this example, the name of the application is **MyApp**, and so is the WPAR's name. Remember that because an application WPAR only exists while its application is running, the **lswpar** command only shows active application WPARs.

Example 5-4 List an application WPAR

```

# lswpar
Name          State  Type  Hostname  Directory
-----
MyApp         A      A     MyApp     /

```

There is also a SMIT window to create and start an application WPAR: **smit simplewpar_app**. The difference though is that the name of the WPAR is mandatory. Figure 5-2 shows this window.

Create an Application Workload Partition

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

		[Entry Fields]	
*	Workload Partition Name	[MyApp]	
	HOSTNAME	[]	
*	Application to run	[/tmp/MyApp]	
	Network INTERFACE		+
	Internet ADDRESS (dotted decimal)	[]	
	Network MASK	[]	
	Create the Workload Partition?	yes	+
	Specification Output File	[]	/

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

Figure 5-2 Simple application WPAR creation using SMIT

5.1.2 Start, stop, and reboot a system WPAR

System WPARs can be started, stopped, and rebooted just as a regular AIX system.

Start

When a system WPAR is in the defined state, you can start it by issuing the **startwpar** command followed by its name. The global environment takes care of the necessary infrastructure, namely, mounting file systems and adding IP addresses (for details, consult “Networks” on page 72). In Example 5-5, **lswpar** shows the defined state (state = “D”), and **startwpar** starts the system WPAR.

Example 5-5 Starting a system WPAR

```
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    D      S     MySysWpar /wpars/MySysWpar
# startwpar MySysWpar
startwpar: COMMAND START, ARGS: MySysWpar
startwpar: Starting workload partition 'MySysWpar'
startwpar: Mounting all workload partition file systems
startwpar: Loading workload partition
startwpar: Exporting workload partition devices
startwpar: Starting workload partition subsystem cor_MySysWpar
0513-059 The cor_MySysWpar Subsystem has been started. Subsystem PID is
372970.
startwpar: Verifying workload partition startup
startwpar: Return Status = SUCCESS
```

In Example 5-6, the state of the WPAR is Active.

Example 5-6 List active system WPAR

```
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    A      S     MySysWpar /wpars/MySysWpar
```

This means that you can now log in to the WPAR and start using it. One way to start using it is from the global environment with the **cllogin** command. This command does not depend on TCP/IP. The command is:

```
# cllogin MySysWpar
```

You can also use SMIT to start a system WPAR:

```
# smit bootwpar_sys
```

Reboot

If you ever need to reboot your WPAR, you can reboot it in the normal AIX way from inside the WPAR: **shutdown -Fr**. Example 5-7 shows the output of this command. During the reboot, the state of the WPAR will be T (Transitional), which means that it is moving from one state to another.

Example 5-7 Reboot a system WPAR from inside

```
# shutdown -Fr
```

```
SHUTDOWN PROGRAM
```

```
Tue Apr 24 17:27:13 CDT 2007
```

```
Wait for 'Rebooting...' before stopping.
```

```
Error reporting has stopped.
```

```
Advanced Accounting has stopped...
```

```
Process accounting has stopped.
```

```
nfs_clean: Stopping NFS/NIS Daemons
```

```
...
```

```
All processes currently running will now be killed...
```

```
Unmounting the file systems...
```

```
Rebooting . . .
```

Another way to reboot a system WPAR is to use the **stopwpar -r** command or the **rebootwpar** from the global environment. In this case, it issues the **shutdown** command in the WPAR and waits up to 600 seconds before ending all the WPAR processes. Example 5-8 shows the output of the **stopwpar** command. You can also force the reboot with the **-F** flag.

Example 5-8 Reboot a system WPAR from the global environment

```
# stopwpar -r MySysWpar
```

```
stopwpar: COMMAND START, ARGS: -r MySysWpar
```

```
stopwpar: Stopping workload partition 'MySysWpar'
```

```
stopwpar: Stopping workload partition subsystem
```

```
0513-044 The cor_MySysWpar Subsystem was requested to stop.
```

```
stopwpar: Waiting up to 600 seconds for workload partition to halt
```

```
Advanced Accounting is not running.
```

```
stopwpar: Shutting down all workload partition processes
```

```
stopwpar: Unmounting all workload partition file systems
```

```
startwpar: COMMAND START, ARGS: MySysWpar
```

```
startwpar: Starting workload partition 'MySysWpar'
```

```
startwpar: Mounting all workload partition file systems
startwpar: Loading workload partition
startwpar: Exporting workload partition devices
startwpar: Starting workload partition subsystem cor_MySysWpar
0513-059 The cor_MySysWpar Subsystem has been started. Subsystem PID is
336078.
startwpar: Verifying workload partition startup
startwpar: Return Status = SUCCESS
```

There is also a SMIT window where you can issue the reboot command:

```
# smit rebootwpar_sys
```

Stop

When you no longer need to work with this WPAR, you can stop it either from inside the WPAR or from the global environment.

Tip: When you need to know whether you are in the global environment or inside a WPAR, you can execute the `uname -W` command. This returns 0 if you are in the global environment and a value other than 0 if you are inside a WPAR. You can also check the host name or the mounted file systems.

Example 5-9 Shut down a system WPAR from inside

```
# hostname
MySysWpar
# uname -W
70
# shutdown now
SHUTDOWN PROGRAM
Wed Apr 25 09:51:06 CDT 2007
Wait for '....Halt completed....' before stopping.
Error reporting has stopped.
...
All processes currently running will now be killed...
Unmounting the file systems...
....Halt completed....
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# lswpar
Name          State Type  Hostname  Directory
-----
```

The fast path for the SMIT window for this operation is **stopwpar_sys**. From the global environment, the command is **stopwpar <name of WPAR>**. Use the **-F** flag to force the shutdown of the WPAR: **stopwpar -F <name of WPAR>**.

5.1.3 Change a WPAR

After the creation of a WPAR, you might need to change some of its attributes. You might use a SMIT window to do it but depending on the type of WPAR that needs to be changed, you will have different paths: **smi t cfgwpar_sys** for system WPARs and **smi t cfgwpar_app** for application WPARs. At the command line, you only use one command: **chwpar**.

WPAR name

As reported by **lswpar**, WPARs have names. This is the identifier to use when referring to the WPAR from the global environment.

You can change this name only when the WPAR is in the defined state. In Example 5-10, the WPAR is active and the **chwpar** fails. After stopping the WPAR, the **chwpar** changes the WPAR name and we can check it with the **lswpar** command.

Example 5-10 Changing a WPAR's name

```
# chwpar -n MyNewName MySysWpar
Cannot change the name (-n) of a running workload partition.
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    A      S     MySysWpar /wpars/MySysWpar
# stopwpar -F MySysWpar
Stopping workload partition MySysWpar.
Stopping workload partition subsystem cor_MySysWpar.
0513-044 The cor_MySysWpar Subsystem was requested to stop.
Shutting down all workload partition processes.
Advanced Accounting is not running.
Unmounting all workload partition file systems.
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    D      S     MySysWpar /wpars/MySysWpar
# chwpar -n MyNewName MySysWpar
# lswpar
```

```

Name      State Type Hostname Directory
-----
MyNewName D    S    MySysWpar /wpars/MySysWpar
# startwpar MyNewName
Starting workload partition MyNewName.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Starting workload partition subsystem cor_MyNewName.
0513-059 The cor_MyNewName Subsystem has been started. Subsystem PID is 393264.
Verifying workload partition startup.

```

Host name

By default, if you do not provide a host name at the time of the creation of the WPAR, the host name is the same as the WPAR's name or the basename of the command, depending on whether it is a system WPAR or an application WPAR. Using the command-line interface (CLI) as shown in Example 5-11 or SMIT, you can change its host name.

Example 5-11 Change a WPAR's host name with chwpar command

```

# lswpar
Name      State Type Hostname Directory
-----
MySysWpar A    S    MySysWpar /wpars/MySysWpar
# chwpar -h pop3mail MySysWpar
# lswpar
Name      State Type Hostname Directory
-----
MySysWpar A    S    pop3mail /wpars/MySysWpar

```

You can also change the host name using the normal AIX commands inside the WPAR. Example 5-12 shows that you must be in the WPAR environment to use the `hostname` command.

Note: Using the `hostname` command inside the WPAR does not persist across a reboot. You must use `chwpar -h` to permanently change the host name.

Example 5-12 Change a WPAR's host name with hostname command

```

# clogin MySysWpar
Last login: Wed Apr 25 09:50:50 CDT 2007 on /dev/Globa1 from
braker.itsc.austin.ibm.com
# hostname

```



```

pop3mail
# hostname smtpServer
smtpServer
# exit
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    A      S     smtpServer /wpars/MySysWpar

```

You can change a WPAR's host name in the active state, and for system WPARs, you can also change it in the defined state.

Base directory

You can change the mount points of your system WPARs using the `-d` flag for base directory.

As with the WPAR's name, you can only change the base directory when in the defined state. Example 5-13 shows this and then proceeds to a successful task. The `chwp` command changes to the new mount points and removes the old ones.

Example 5-13 Changing the base directory of a system WPAR

```

# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    A      S     MySysWpar /wpars/MySysWpar
# chwp -d /wpars/MyNewMountPoint MySysWpar
Cannot change the base directory (-d) of a running workload partition.
# stopwpar -F MySysWpar
Stopping workload partition MySysWpar.
Stopping workload partition subsystem cor_MySysWpar.
0513-044 The cor_MySysWpar Subsystem was requested to stop.
Shutting down all workload partition processes.
Advanced Accounting is not running.
Unmounting all workload partition file systems.
# chwp -d /wpars/MyNewMountPoint MySysWpar
# startwpar MySysWpar
Starting workload partition MySysWpar.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Starting workload partition subsystem cor_MySysWpar.
0513-059 The cor_MySysWpar Subsystem has been started. Subsystem PID is 327854.
Verifying workload partition startup.

```

```

# lswpar
Name      State  Type  Hostname  Directory
-----
MySysWpar A      S      MySysWpar  /wpars/MyNewMountPoint
# mount
node      mounted      mounted over  vfs      date      options
-----
/dev/hd4      /              jfs2  May 09 10:55 rw,log=/dev/hd8
/dev/hd2      /usr           jfs2  May 09 10:55 rw,log=/dev/hd8
/dev/hd9var   /var           jfs2  May 09 10:58 rw,log=/dev/hd8
/dev/hd3      /tmp           jfs2  May 09 10:58 rw,log=/dev/hd8
/dev/hd1      /home          jfs2  May 09 10:59 rw,log=/dev/hd8
/proc        /proc          procfs May 09 10:59 rw
/dev/hd10opt  /opt           jfs2  May 09 10:59 rw,log=/dev/hd8
/dev/fs1v00   /wpars/MyNewMountPoint jfs2  May 09 11:32 rw,log=INLINE
/dev/fs1v01   /wpars/MyNewMountPoint/home jfs2  May 09 11:32
rw,log=INLINE
/opt          /wpars/MyNewMountPoint/opt namefs May 09 11:32 ro
/proc        /wpars/MyNewMountPoint/proc namefs May 09 11:32 rw
/dev/fs1v02   /wpars/MyNewMountPoint/tmp jfs2  May 09 11:32
rw,log=INLINE
/usr          /wpars/MyNewMountPoint/usr namefs May 09 11:32 ro
/dev/fs1v03   /wpars/MyNewMountPoint/var jfs2  May 09 11:32
rw,log=INLINE

```

Resource Control

One of the most important features of the workload partitions is the ability to specify resource allocation to each workload. You can modify the CPU and memory usage, the number of processes, the number of threads, and the resource set where they run.

Resource control is performed using the **-R** option of the **mkwpar**, **chwpar**, **wparexec**, and **lswpar** commands.

Resource control is explained in detail in Chapter 7, “Resource control” on page 237.

Networks

A system WPAR works in the same manner as a full LPAR. You might want to have a service IP address and be able to log in to it remotely. In order to do this, you must consider the networks that you want to use, because you can only add IPs to the existing adapters of the global environment as aliases. Also, the networks must be routable, that is, the WPAR’s IP address must be in the same subnet of the global environment’s IP address. In case you are adding an IP

address from a different subnet, remember to explicitly define the **interface** where you want to place that IP.

In the global environment, use the **ifconfig** command to check the IP address and interface. Example 5-14 shows how to check for existing IPs in the global environment.

Example 5-14 Check the global environment IP address

```
# ifconfig en0
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
  inet 9.3.5.145 netmask 0xfffffe00 broadcast 9.3.5.255
  tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
```

In Example 5-15, the IP address that we add is in the same subnet. It is already registered in the DNS server, but you can register it later or use **/etc/hosts** on the WPAR and the global environment. Because we want to use the DNS server, we can edit **/etc/resolv.conf** of the WPAR and add our nameserver and domain. If your global environment already has the **/etc/resolv.conf** configured, you might copy it to the **/wpar/<your wpar>/etc** from the global environment. Example 5-15 also shows the difference between checking for IPs in the global environment and in the WPAR environment. In “Name resolution and auto start” on page 75, we show a preferable way to do this at creation time.

Example 5-15 Change the network settings

```
# cp /etc/resolv.conf /wpars/MySysWpar/etc
# chwpar -N interface=en0 address=9.3.5.147 netmask=255.255.254.0
MySysWpar
# ifconfig en0
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
  inet 9.3.5.145 netmask 0xfffffe00 broadcast 9.3.5.255
  inet 9.3.5.147 netmask 0xfffffe00 broadcast 9.3.5.255
  tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
# clogin MySysWpar
# ifconfig -a
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
  inet 9.3.5.147 netmask 0xfffffe00 broadcast 9.3.5.255
  tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
```

```

lo0:
flags=e08084b<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPRT,6
4BIT>
    inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
    inet6 ::1/0
    tcp_sendspace 131072 tcp_recvspace 131072 rfc1323 1

```

The IP address is registered as `rutland`, but the host name is `smtpServer`. Although they can coexist if the name resolution is correct, we can change it with:

```
# chwpar -h rutland MySysWpar
```

Now, we can telnet to the WPAR from outside the global environment.

Application WPARs work the same way as system WPARs regarding networks, except that because of the use of the global environment's file systems, application WPARs also use the global environment's `/etc/resolv.conf` and other name resolution resources.

The system administrator can add several IPs to the WPARs.

5.1.4 Creating WPARs: Advanced options

At creation time, there are options available for system and application WPARs. System workload partitions are created with the `mkwpar` and application workload partitions are created with the `wparexec` command. Each of these commands has its own flags.

System WPAR

Several of the changes illustrated in the previous example by the `chwpar` command can be defined at creation time. The flags you can use with `mkwpar` are similar to the `chwpar` flags. They have the same syntax and the same meaning:

-n < <i>name of WPAR</i> >	Name of the WPAR
-h < <i>hostname</i> >	Host name
-d < <i>full path</i> >	Base directory
-R < <i>resources</i> >	Resource controls settings (active, rset, CPU, memory, procVirtMem, shares_CPU, shares_memory, totalProcesses, and totalThreads)
-N < <i>settings</i> >	Network settings (interface, address, netmask, and broadcast)

Name resolution and auto start

One interesting and useful flag is the `-r`, which copies the global network name resolution configuration files into the workload partition. With the `-A` flag, you can set this WPAR to start automatically on system boot. With the `-s` flag, the WPAR starts after creation.

Therefore, if you want to create or recreate the system WPAR that we have been using as an example with resource controls and additionally copy all of the name resolution configuration files from the global environment, the following command shows you how:

```
# mkwpar -n MySysWpar -h rutland -N interface=en0 address=9.3.5.147
netmask=255.255.254.0 -r -R shares_CPU=10 CPU=6%-30%,45%
shares_memory=20 memory=5%-10%,20% totalProcesses=128 totalThreads=512
-A -s
```

At the very end of the creation of the WPAR, you will see the new `-r` flag output, followed by the auto start of the WPAR, as shown in Example 5-16.

Example 5-16 End of the output of `mkwpar -r`

```
...
Copying network name resolution configuration...
    /etc/resolv.conf
    /etc/hosts
    /etc/netsvc.conf
mkwpar: Workload partition MySysWpar created successfully.
Starting workload partition MySysWpar.
Mounting all workload partition file systems.
Loading workload partition.
Exporting workload partition devices.
Starting workload partition subsystem cor_MySysWpar.
0513-059 The cor_MySysWpar Subsystem has been started. Subsystem PID is
303250.
Verifying workload partition startup.
```

A simpler way to configure the network is to use the resolvable host name as the WPAR name, because the `mkwpar` and the `wparexec` will recognize it and put the proper IP address at the first compatible available adapter. This process is shown in Example 5-17.

Example 5-17 Create a WPAR using the host name as the name of the WPAR

```
# host rutland
rutland is 9.3.5.147
# mkwpar -n rutland -r -R shares_CPU=10 CPU=6%-30%,45% shares_memory=20
memory=5%-10%,20% totalProcesses=128 totalThreads=512 -A -s
...
# clogin rutland
# ifconfig -a
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GR
OUPRT,64BIT,CHECKSUM_OFFLOAD(ACTIVE),CHAIN>
    inet 9.3.5.147 netmask 0xfffffe00 broadcast 9.3.5.255
    tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1
lo0:
flags=e08084b<UP,BROADCAST,LOOPBACK,RUNNING,SIMPLEX,MULTICAST,GROUPRT,6
4BIT>
    inet 127.0.0.1 netmask 0xff000000 broadcast 127.255.255.255
    inet6 ::1/0
    tcp_sendspace 131072 tcp_recvspace 131072 rfc1323 1
```

Root password at creation time

For added security, you might want to create a root password at the WPAR's creation time. This will prevent unwanted logins from the network. The flag is `-P`. It asks for a password and confirmation after the creation of the WPAR as shown in Example 5-18.

Example 5-18 Assigning a password to the root user at creation time

```
# mkwpar -n MySysWpar -P
...
Changing password for "root@MySysWpar"
root@MySysWpar's New password:
Re-enter root@MySysWpar's new password:

mkwpar: Workload partition MySysWpar created successfully.
To start the workload partition, execute the following as root:
startwpar [-v] 'MySysWpar'
```

Other file systems

If you need to use a volume group other than the rootvg for your WPAR's file systems, use the flag `-g <name of VG>`.

The default `/usr` and `/opt` of system WPARs are just read-only namefs mounts of the global environment's `/usr` and `/opt`. If you need writable `/usr` or `/opt`, you can use the `-1` flag. This, of course, will take longer to copy all the `/usr` and `/opt` files.

Note: *Namefs* is a pseudo-file system used to mount a subdirectory of a real file system on another mount point with the same or different mount options. In WPAR, namefs means the ability to mount `/usr` as a read-only file system under `/wpars/<WPAR name>/usr`.

The creation of a system WPAR involves creating, by default, several file systems under `/wpars` that might not be the best solution if you have hundreds or even thousands of system WPARs.

One way to address this issue is to use directories instead of file systems. You can create just one file system for your system files and afterwards create the necessary file systems for your application data.

Another way is to use a remote Network File System (NFS) to store your WPAR. In fact, if you intend to have a mobility-enabled WPAR, you must use NFS for all of your WPAR file systems.

The command you use to define your file systems instead of using the default ones is `mkwpar -M` and these are its attributes:

vfs	jfs, jfs2, nfs, namefs, or directory
dev	For local file systems: existing logical volume in global environment to hold WPAR file system for namefs: name of the global environment's directory for remote NFS: directory shared on the remote host
directory	Name of the directory inside the WPAR
size	Size of the file system in <code>crfs</code> acceptable format. Applicable to <code>vfs=jfs</code> and <code>vfs=jfs2</code> only
crfsopts	Way to pass options to the <code>crfs</code> command that are different than the ones that can otherwise be specified with the <code>mkwpar</code> command
mode	Octal permission mode of the base directory

vg	Name of existing volume group in the global environment where the file systems are created. Applicable to vfs=jfs and vfs=jfs2 only
logname	The file system's log device name. Applicable to vfs=jfs and vfs=jfs2 only
host	Name of the NFS server (if vfs=nfs)
mountopts	Options in the same format as the -o of the mount command and saved in /etc/filesystems.

In Example 5-19, we show how to create a system WPAR called OneFSWpar with only one jfs2 file system on local disks on another VG (datavg) and in an existing, already mirrored LV (wpar1lv). In this case, the /usr and /opt are just directories of the same file systems. This means that they are private to the WPAR and will not share space in disk or memory with the global environment or other WPARs.

Example 5-19 One file system system WPAR

```
# lslv wpar1lv
LOGICAL VOLUME:      wpar1lv          VOLUME GROUP:      datavg
LV IDENTIFIER:      000d3ed80000d60000001128c50f0de.1  PERMISSION:        read/write
VG STATE:           active/complete    LV STATE:           closed/syncd
TYPE:                jfs2                WRITE VERIFY:       off
MAX LPs:             512                PP SIZE:            8 megabyte(s)
COPIES:              2                SCHED POLICY:       parallel
LPs:                 128                PPs:                256
STALE PPs:           0                BB POLICY:          relocatable
INTER-POLICY:        minimum            RELOCATABLE:        yes
INTRA-POLICY:        middle            UPPER BOUND:        32
MOUNT POINT:         N/A                LABEL:              None
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LP COPY ON A SEPARATE PV ?: yes
Serialize IO ?:      NO
# mkwpar -M directory=/ vfs=jfs2 dev=/dev/wpar1lv -M vfs=directory directory=/var -M
vfs=directory directory=/tmp -M vfs=directory directory=/home -M vfs=directory
directory=/usr -M vfs=directory directory=/opt -n OneFSWpar
...
# lswpar
Name      State  Type  Hostname      Directory
-----
OneFSWpar  D      S      OneFSWpar     /wpars/OneFSWpar
# startwpar OneFSWpar
...
# lswpar -M OneFSWpar
Name      MountPoint      Device      Vfs      Nodename  Options
```



```

-----
OneFSWpar /wpars/OneFSWpar /dev/wpar1lv jfs2
OneFSWpar /wpars/OneFSWpar/proc /proc namefs rw
# mount
node mounted mounted over vfs date options
-----
/dev/hd4 / jfs2 Apr 26 17:49 rw,log=/dev/hd8
/dev/hd2 /usr jfs2 Apr 26 17:49 rw,log=/dev/hd8
/dev/hd9var /var jfs2 Apr 26 17:50 rw,log=/dev/hd8
/dev/hd3 /tmp jfs2 Apr 26 17:50 rw,log=/dev/hd8
/dev/hd1 /home jfs2 Apr 26 17:50 rw,log=/dev/hd8
/proc /proc procfs Apr 26 17:50 rw
/dev/hd10opt /opt jfs2 Apr 26 17:50 rw,log=/dev/hd8
/dev/wpar1lv /wpars/OneFSWpar jfs2 Apr 27 14:26 rw,log=INLINE
/proc /wpars/OneFSWpar/proc namefs Apr 27 14:26 rw
-----

```

Live mobility conditions

Because this WPAR is using local file systems, it will not be able to have live mobility to another global environment. In order to have live mobility with WPARs, they must meet three conditions: the file systems must be NFS mounts, the WPAR must be checkpointable, and WPAR Manager must be working. To create a system WPAR with NFS mounts, use the command in Example 5-20.

Example 5-20 Create a system WPAR on NFS mounts

```

# mkwpar -M directory=/ vfs=nfs dev=/big/wpars/OneFSWpar host=gracyfarms -M
vfs=directory directory=/var -M vfs=directory directory=/tmp -M vfs=directory
directory=/home -M vfs=directory directory=/usr -M vfs=directory directory=/opt -n
OneFSWpar
...
# lswpar
Name State Type Hostname Directory
-----
OneFSWpar D S OneFSWpar /wpars/OneFSWpar
# startwpar OneFSWpar
...
# mount
node mounted mounted over vfs date options
-----
/dev/hd4 / jfs2 Apr 30 15:15 rw,log=/dev/hd8
/dev/hd2 /usr jfs2 Apr 30 15:15 rw,log=/dev/hd8
/dev/hd9var /var jfs2 Apr 30 15:15 rw,log=/dev/hd8
/dev/hd3 /tmp jfs2 Apr 30 15:15 rw,log=/dev/hd8
/dev/hd1 /home jfs2 Apr 30 15:15 rw,log=/dev/hd8
/proc /proc procfs Apr 30 15:15 rw
-----

```

```

/dev/hd10opt      /opt          jfs2   Apr 30 15:15 rw,log=/dev/hd8
gracyfarms /big/wpars    /wpars/OneFSWpar nfs3   Apr 30 15:23 bg,intr,rw
                /proc        /wpars/OneFSWpar/proc namefs Apr 30 15:23 rw

```

Install the WPAR Manager agent in this WPAR's global environment and install the WPAR Manager and its components in an LPAR of your choice.

Finally, use the flag **-c** with the command **mkwpar** to create a checkpointable WPAR. If the WPAR already exists, stop it and then change it with **chwpar -c**.

Specification files

An example of a putting it all together is the WPAR shown in Table 5-1.

Table 5-1 Define a complex WPAR's CLI parameters

Flag	Description	Example value
-n	name	MySysWpar
-h	host name	rutland
-N	network	interface=en0 address=9.3.5.147 netmask=255.255.254.0
-r	name resolution	N/A
-R	resource controls	shares_CPU=10 CPU=6%-30%,45% shares_memory=20 memory=5%-10%,20% totalProcesses=128 totalThreads=512
-A	auto start with global	N/A
-s	auto start after creation	N/A
-M	filesystems	directory=/ vfs=nfs dev=/big/wpars host=gracyfarms

The command for this advanced WPAR is in Example 5-21 on page 81.

Example 5-21 Create a complex WPAR with one command

```
# mkwpar -n MySysWpar -h rutland -N interface=en0 address=9.3.5.147
netmask=255.255.254.0 -r -R shares_CPU=10 CPU=6%-30%,45%
shares_memory=20 memory=5%-10%,20% totalProcesses=128 totalThreads=512
-A -s -M directory=/ vfs=nfs dev=/big/wpars host=gracyfarms -M
vfs=directory directory=/var -M vfs=directory directory=/tmp -M
vfs=directory directory=/home -M vfs=directory directory=/usr -M
vfs=directory directory=/opt
```

A simpler and better way to create an advanced WPAR is to use specification files. There is a sample file in `/usr/samples/wpars/` called `sample.spec`. Specification files are text files with stanzas, just like `/etc/filesystems` for example, that we use to create a WPAR without writing a long command. To use a spec file, all you need is the spec file and the flag `-f`:

```
# mkwpar -f /tmp/MySysWpar.spec
```

You can create a spec file from the command line using the `mkwpar` command with the flags `-o <name of the spec file> -w`, edit it, and use it later as shown in Example 5-22.

Example 5-22 Creating a spec file from the command line

```
# mkwpar -n MySysWpar -h rutland -N interface=en0 address=9.3.5.147
netmask=255.255.254.0 -r -R shares_CPU=10 CPU=6%-30%,45%
shares_memory=20 memory=5%-10%,20% totalProcesses=128 totalThreads=512
-A -s -M directory=/ vfs=nfs dev=/big/wpars host=gracyfarms -M
vfs=directory directory=/var -M vfs=directory directory=/tmp -M
vfs=directory directory=/home -M vfs=directory directory=/usr -M
vfs=directory directory=/opt -o /tmp/MySysWpar.spec -w
```

Example 5-21 is the specification file for this WPAR.

Example 5-23 Specification file for an advanced WPAR

```
general:
    name = "MySysWpar"
    hostname = "rutland"
    directory = "/wpars/MySysWpar"
    privateusr = "yes"
    devices = "/etc/wpars/devexports"
    auto = "yes"
    preserve = "no"
    copy_nameres = "yes"
```

```
network:
```

```
        broadcast = "9.3.5.255"
        interface = "en0"
        address = "9.3.5.147"
        netmask = "255.255.254.0"

mount:
    dev = "/big/wpars"
    mountopts = "bg,intr"
    directory = "/"
    vfs = "nfs"
    host = "gracyfarms"

mount:
    vfs = "directory"
    directory = "/var"

mount:
    vfs = "directory"
    directory = "/tmp"

mount:
    vfs = "directory"
    directory = "/home"

mount:
    vfs = "directory"
    directory = "/usr"

mount:
    vfs = "directory"
    directory = "/opt"

mount:
    dev = "/proc"
    directory = "/proc"
    vfs = "namefs"
    mountopts = "rw"

resources:
    shares_CPU = "10"
    memory = "5%-10%,20%"
    totalProcesses = "128"
    totalThreads = "512"
    shares_memory = "20"
    CPU = "6%-30%,45%"
```

```
security:
    secfile = "/etc/wpars/secattrs"
```

There is another way to create WPARs: Use other working WPARs. With the flag **-e**, you can create a new WPAR with exactly the same parameters as the other running WPAR or create a new spec file if you add the **-w** flag.

Additional parameters specified in the command line supersede the parameters coming from the spec file.

The SMIT fastpath to create advanced system WPARs is **advancedwpar_sys**.

Application WPAR

As with system WPARs, application WPARs can have several attributes set at creation time. The command is **wparexec**. Several of the flags are the same and have the same meaning as the ones in **chwpar** or in **mkwpar**. Because application WPARs use the same file systems as the global environment and do not exist after the application stops, the **-r** or **-A** options do not apply. Additionally, they start with the application. They do not need the **-s** flag.

```
# wparexec -n MyApp -h rutland -R shares_CPU=10 CPU=6%-30%,45%
shares_memory=20 memory=5%-10%,20% totalProcesses=128 totalThreads=512
-N interface=en0 address=9.3.5.147 netmask=255.255.254.0 /tmp/MyApp
```

Using the same name for the WPAR's name and the WPAR's host name saves some time. As with system WPAR, using the host name as the WPAR's name sets the host name and configures the proper IP address for the application WPAR:

```
# wparexec -n rutland -R shares_CPU=10 CPU=6%-30%,45% shares_memory=20
memory=5%-10%,20% totalProcesses=128 totalThreads=512 /tmp/MyApp
```

Application WPARs use the global environment's file systems. If needed, you can create or use some auxiliary file systems. If you want JFS or JFS2 file systems, you must create them at the global environment level. Additionally, if the application depends on some file systems, using the **-M** flag allows **wparexec** to verify if they are mounted and if they are not, it mounts them.

5.1.5 Recover from a broken state

If a WPAR ever gets into a broken state, as explained in Chapter 4, “State management” on page 55, check:

- ▶ The logs at /var/adm/ras
- ▶ The logs at /var/adm/wpars
- ▶ For WPARs running processes. Use `ps -@` to show processes by WPAR.
- ▶ For WPAR-mounted file systems
- ▶ The locks at /etc/wpars/locks

Before you can start the WPAR again, it must go to the defined state:

```
# stopwpar -F MySysWpar
```

5.1.6 Remove a WPAR

In order to remove a system WPAR from the system, all you have to do is:

- ▶ Verify that the WPAR is in the defined state
- ▶ Make sure you do not need it or that you have a working backup of it
- ▶ Issue the command `rmwpar <name of the WPAR>`

In Example 5-24, you can see how to remove a WPAR.

Example 5-24 Remove a WPAR

```
# lswpar
Name      State  Type  Hostname  Directory
-----
MySysWpar A      S     rutland  /wpars/MySysWpar
# rmwpar MySysWpar
rmwpar: Workload partition 'MySysWpar' is running.
rmwpar: Specify -s or -F to stop the workload partition before
removing.
rmwpar: Return Status = FAILURE
# stopwpar MySysWpar
stopwpar: COMMAND START, ARGS: MySysWpar
stopwpar: Stopping workload partition 'MySysWpar'
stopwpar: Stopping workload partition subsystem
0513-044 The cor_MySysWpar Subsystem was requested to stop.
stopwpar: Waiting up to 600 seconds for workload partition to halt
Advanced Accounting is not running.
stopwpar: Shutting down all workload partition processes
```

```

stopwpar: Unmounting all workload partition file systems
stopwpar: Return Status = SUCCESS
# lswpar
Name          State Type  Hostname  Directory
-----
MySysWpar    D      S     rutland  /wpars/MySysWpar
# rmwpar MySysWpar
rmwpar: Removing workload partition MySysWpar
rmwpar: Removing filesystem /wpars/MySysWpar/var.
rmlv: Logical volume fslv03 is removed.
rmwpar: Removing filesystem /wpars/MySysWpar/usr.
rmwpar: Removing filesystem /wpars/MySysWpar/tmp.
rmlv: Logical volume fslv02 is removed.
rmwpar: Removing filesystem /wpars/MySysWpar/proc.
rmwpar: Removing filesystem /wpars/MySysWpar/opt.
rmwpar: Removing filesystem /wpars/MySysWpar/home.
rmlv: Logical volume fslv01 is removed.
rmwpar: Removing filesystem /wpars/MySysWpar.
rmlv: Logical volume fslv00 is removed.
rmwpar: Return Status = SUCCESS

```

Issuing the **rmwpar** without any option flags removes the file systems for that WPAR. If you might want to preserve the contents of the file systems, use the flag **-p** to prevent the deletion of those file systems as shown in Example 5-25.

Example 5-25 Preserve the file systems when removing a WPAR

```

# rmwpar -p MySysWpar
rmwpar: Removing workload partition MySysWpar
rmwpar: Preserving filesystems.
rmwpar: Return Status = SUCCESS

```

If you preserve the file systems, you can recreate the WPAR using the **-p** flag of the **mkwpar** command. Remember to explicitly specify all of the file systems with the **-M** flag unless you specify the removed WPAR's name after the **-p**:

```

# mkwpar -n MyNewWpar -p MySysWpar

```

5.2 Software installation and maintenance

System WPARs have, by default, two read-only file systems: /usr and /opt. This saves space on disk and time for the WPAR creation but prevents the normal software installation inside the WPAR. Also, the WPAR's root user cannot delete software from the /usr or /opt directories. In order to install software on a system WPAR, we have to think differently. There are usually two ways to install software: in the global environment using normal installation procedures and inside the WPAR using read and write file systems.

Application WPARs do not pose a problem on software installation, because they use whichever software is installed in the global environment.

5.2.1 Software availability

The global environment is a regular AIX. Accessing the CD-ROM or DVD-ROM, an NFS mount or NIM is the same as usual. However, for WPARs, there are differences.

In order to have software available inside the WPAR, you need to have it in a file system form: the software repository, a mounted CD or DVD, or an NFS mount.

If the software you want to install is in a CD format, copy it to a regular file system. You can use `smit bffcreate` to do that. Just mounting the CD might be enough for the operation that you want. In this case, pass the mount point as a `namefs` parameter to the WPAR at creation time: `-M directory=/cdrom dev=/cdrom`.

If your WPAR is already running, the best way to temporarily mount file systems is under `/wpar/<name of the WPAR>/<my mount point>` in the global environment, as shown in Example 5-26.

Example 5-26 Mounting a temporary file system

```
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# mount
```

node	mounted	mounted over	vfs	date	options
/dev/hd4	/		jfs2	May 01 18:27	rw,log=/dev/hd8
/dev/hd2	/usr		jfs2	May 01 18:27	rw,log=/dev/hd8
/dev/hd9var	/var		jfs2	May 01 18:27	rw,log=/dev/hd8
/dev/hd3	/tmp		jfs2	May 01 18:27	rw,log=/dev/hd8


```

/dev/hd1      /home      jfs2   May 01 18:27 rw,log=/dev/hd8
/proc        /proc      procfs May 01 18:27 rw
/dev/hd10opt /opt       jfs2   May 01 18:27 rw,log=/dev/hd8
/dev/fslv00  /wpars/MySysWpar jfs2   May 02 10:16 rw,log=INLINE
/dev/fslv01  /wpars/MySysWpar/home jfs2   May 02 10:16 rw,log=INLINE
/dev/fslv02  /wpars/MySysWpar/opt jfs2   May 02 10:16 rw,log=INLINE
/proc        /wpars/MySysWpar/proc namefs May 02 10:16 rw
/dev/fslv03  /wpars/MySysWpar/tmp jfs2   May 02 10:16 rw,log=INLINE
/dev/fslv04  /wpars/MySysWpar/usr jfs2   May 02 10:16 rw,log=INLINE
/dev/fslv05  /wpars/MySysWpar/var jfs2   May 02 10:16 rw,log=INLINE

```

```
# crfs -v jfs2 -g datavg -m /wpars/MySysWpar/big -a size=1G -a logname=INLINE
```

```
File system created successfully.
1040148 kilobytes total disk space.
New File System size is 2097152
```

```
# mount /wpars/MySysWpar/big
```

```
# cp -R /cdrom /wpars/MySysWpar/big
```

```
# clogin MySysWpar
```

```
Last login: Wed May 2 13:49:28 CDT 2007 on /dev/Global from
braker.itsc.austin.ibm.com
```

```
# hostname
```

```
rutland
```

```
# uname -w
```

```
70
```

```
# mount
```

node	mounted	mounted over	vfs	date	options
/dev/fslv00	/		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv01	/home		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv02	/opt		jfs2	May 02 10:16	rw,log=INLINE
/proc	/proc		namefs	May 02 10:16	rw
/dev/fslv03	/tmp		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv04	/usr		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv05	/var		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv10	/big		jfs2	May 02 13:58	rw,log=INLINE

For NFS, use the -M flag at creation time, or if the WPAR is already running, mount it inside the WPAR, as shown in Example 5-27.

Example 5-27 Mounting nfs inside the WPAR

```
# mkdir -p /nfs/big
# mount gracyfarms:/big /nfs/big
# mount
```

node	mounted	mounted over	vfs	date	options
/dev/fslv00	/		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv01	/home		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv02	/opt		jfs2	May 02 10:16	rw,log=INLINE
/proc	/proc		namefs	May 02 10:16	rw
/dev/fslv03	/tmp		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv04	/usr		jfs2	May 02 10:16	rw,log=INLINE
/dev/fslv05	/var		jfs2	May 02 10:16	rw,log=INLINE
gracyfarms /big	/nfs/big		nfs3	May 02 14:11	nodev

You can also use SMIT.

5.2.2 Using installp

Because application WPARs use the global environment's file systems and installed software, this section only refers to system WPARs.

Installing in the global environment

This type of installation is the best if you want to save space and need to run the same version of the software in several WPARs.

First, you install the software in the global environment using the **installp** command or SMIT. After it is complete, you synchronize the installation with the WPAR. This installs the root part of the newly installed software onto the WPAR.

Note: Synchronize the WPAR after the software installation in the global environment to prevent inconsistencies between the root and user parts of the WPARs.

In Example 5-28 on page 89, we start the installation of the openssh server.

Example 5-28 Installing openssh in the global environment WPAR

```
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# lswpar
Name      State Type  Hostname  Directory
-----
MySysWpar A      S      rutland  /wpars/MySysWpar
# lslpp -L openssh.base.server
Fileset                                Level State Type  Description (Uninstaller)
-----
lslpp: 0504-132 Fileset openssh.base.server not installed.
# installp -ac -d /tmp openssh.base.server
+-----+
+-----+
Pre-installation Verification...
+-----+
+-----+
Verifying selections...done
Verifying requisites...done
Results...

SUCSESSES
-----
Filesets listed in this section passed pre-installation verification
and will be installed.

Selected Filesets
-----
openssh.base.server 4.3.0.5301          # Open Secure Shell Server

<< End of Success Section >>

+-----+
+-----+
BUILDDATE Verification ...
+-----+
+-----+
Verifying build dates...done
FILESET STATISTICS
-----
  1 Selected to be installed, of which:
    1 Passed pre-installation verification
-----
  1 Total to be installed

+-----+
```

Installing Software...

+-----+

```
installp: APPLYING software for:
         openssl.base.server 4.3.0.5301
```

```
. . . . . << Copyright notice for openssl.base >> . . . . .
Licensed Materials - Property of IBM
```

```
5765E6111
```

```
...
```

```
0513-071 The sshd Subsystem has been added.
```

```
0513-059 The sshd Subsystem has been started. Subsystem PID is 233682.
```

```
Finished processing all filesets. (Total time: 4 secs).
```

+-----+

Summaries:

+-----+

Installation Summary

Name	Level	Part	Event	Result
openssl.base.server	4.3.0.5301	USR	APPLY	SUCCESS
openssl.base.server	4.3.0.5301	ROOT	APPLY	SUCCESS

Now that we have the openssl fileset installed in the global environment, we must synchronize it with the WPAR. At this point, the WPAR's /usr already has the software installed, but the root part is not in sync with the user part of the installation. In Example 5-29 on page 91, we use the **syncwpar MySysWpar** from the global environment, but we can also use the **syncroot** from inside the WPAR.

The two main differences between these commands are:

- syncwpar -A** Synchronizes all WPARs in the global environment
- syncroot** Even if the WPAR's root user does not have access to the global environment, the root user can synchronize the root user's WPAR environment.

Tip: Syncwpar calls syncroot inside the targeted WPAR.

Example 5-29 Synchronizing root part with WPAR

```
# syncwpar MySysWpar
*****
syncwpar: Synchronizing workload partition 'MySysWpar' (1 of 1)
*****
syncwpar: COMMAND START, ARGS: MySysWpar
syncwpar: Executing "/usr/sbin/syncroot" in workload partition 'MySysWpar'
syncroot: Processing root part installation status.
syncroot: Synchronizing installp software.
+-----+
                    Pre-installation Verification...
+-----+
Verifying selections...done
Verifying requisites...done
Results...

SUCCESSSES
-----
Filesets listed in this section passed pre-installation verification
and will be installed.

Selected Filesets
-----
openssh.base.server 4.3.0.5301           # Open Secure Shell Server

<< End of Success Section >>

+-----+
                    BUILDDATE Verification ...
+-----+
Verifying build dates...done
FILESET STATISTICS
-----
  1 Selected to be installed, of which:
    1 Passed pre-installation verification
-----
  1 Total to be installed

+-----+
                    Installing Software...
+-----+

installp: APPLYING software for:
        openssh.base.server 4.3.0.5301
```

0513-071 The sshd Subsystem has been added.
Finished processing all filesets. (Total time: 0 secs).

+-----+
| Summaries: |
+-----+

Installation Summary

Name	Level	Part	Event	Result
openssh.base.server	4.3.0.5301	ROOT	APPLY	SUCCESS

syncroot: Processing root part installation status.
syncroot: Installp root packages are currently synchronized.
syncroot: Root part is currently synchronized.
syncroot: Returns Status = SUCCESS
syncwpar: Workload partition 'MySysWpar' synchronized successfully

syncwpar: Return Status = SUCCESS

After the installation, check the openssh server from inside the WPAR as shown in Example 5-30.

Example 5-30 Checking openssh server inside WPAR

```
# lssrc -s sshd
```

Subsystem	Group	PID	Status
sshd	ssh	332014	active

When you have software in the global environment that is not needed in the WPARs, you can make the software private to the global environment. This will prevent the synchronization of the filesets to the WPARs and therefore, missing synchronization errors. The command is **swvpdmgr**. Use the **-p** flag to turn the filesets to a private state and **-s** flag to return to the shared (default) state.

Installing inside the system WPAR

In order to install software inside the system WPAR, you must have writable file systems. These file systems can be private `/usr` and `/opt` or required for an application installation. The process for installing in the system WPAR is similar to the global environment but you do not need to synchronize. Simply install using `installp`, `SMIT`, or your application's scripts.

Note: Certain application's installation scripts search for particular devices. If the your application's scripts fail inside the WPAR, consider installing in the global environment.

In Example 5-31, we see a system WPAR with private /usr and /opt. It was created with the `-1` flag of the `mkwpar` command.

Example 5-31 Installing in a system WPAR

```
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# mount
node          mounted      mounted over  vfs      date      options
-----
/dev/hd4      /            /            jfs2     May 01 13:31 rw,log=/dev/hd8
/dev/hd2      /usr        /usr        jfs2     May 01 13:31 rw,log=/dev/hd8
/dev/hd9var   /var        /var        jfs2     May 01 13:31 rw,log=/dev/hd8
/dev/hd3      /tmp        /tmp        jfs2     May 01 13:31 rw,log=/dev/hd8
/dev/hd1      /home       /home       jfs2     May 01 13:31 rw,log=/dev/hd8
/proc        /proc       /proc       procfs   May 01 13:31 rw
/dev/hd10opt  /opt        /opt        jfs2     May 01 13:31 rw,log=/dev/hd8
/dev/fslv00   /wpars/MySysWpar jfs2     May 01 14:15 rw,log=INLINE
/dev/fslv01   /wpars/MySysWpar/home jfs2     May 01 14:15 rw,log=INLINE
/dev/fslv02   /wpars/MySysWpar/opt jfs2     May 01 14:15 rw,log=INLINE
/proc        /wpars/MySysWpar/proc namefs   May 01 14:15 rw
/dev/fslv03   /wpars/MySysWpar/tmp jfs2     May 01 14:15 rw,log=INLINE
/dev/fslv04   /wpars/MySysWpar/usr jfs2     May 01 14:15 rw,log=INLINE
/dev/fslv05   /wpars/MySysWpar/var jfs2     May 01 14:15 rw,log=INLINE
# clogin MySysWpar
root's Password:
Last unsuccessful login: Tue May  1 13:37:05 CDT 2007 on ssh from 9.3.4.174
Last login: Tue May  1 13:37:25 CDT 2007 on /dev/pts/0 from 9.3.4.174
# hostname
rutland
# uname -W
70
# installp -ac -d /tmp openssh.base.server
```

If your application does not need to write in /usr or /opt, then you can have a system WPAR with the default namefs /usr and /opt and add the necessary file system to your application. Applications that can be relocated do not need to write in /usr or /opt and, for that reason, are good candidates for installation into a system WPAR, which shares /usr and /opt with the global environment.

5.2.3 Using NIM

The Network Installation Manager (NIM) poses new challenges to WPARs. Because it depends on the network, including the adapter, and WPARs only have aliases, the usual NIM operations work differently. To perform NIM operations, you must be in the global environment, because that is where the network adapter is. NIM servers cannot be run inside a WPAR (AIX V6.1 does not support running NFS servers inside a WPAR).

In the global environment, the NIM operations work as usual. The SMIT fast path is `smit nim`.

After the installation of any software in the global environment that uses `/usr` or `/opt`, you must synchronize the WPARs that have a shared `/usr` or `/opt`.

5.2.4 Updating software

As with the software installation, update the global environment first and then the WPARs that have private `/usr` and `/opt`.

Because the WPARs share the same running kernel with the global environment, they cannot be on different levels. For operating system updates, when you want to update one WPAR, you have to consider updating them all. If the WPAR has a shared `/usr` and `/opt`, then it is updated with the `syncwpar` command after the update of the global environment. First, run `smit update_all` as usual. After it ends, run the `syncwpar <name of the WPAR>` for each of the WPARs or run `syncwpar -A` to synchronize them all at the same time.

Alternatively, if the WPARs have a private `/usr` and a private `/opt`, you must update them one by one after the global environment. First, use `smit update_all` in the global environment, then log in to the WPAR that you want and run `smit update_all`.

If you are updating an application, such as DB2 or WebSphere®, you can do it without updating the nearby WPARs as long as the WPAR has a private `/usr` and `/opt`, or they can coexist in the same file system.

Note: For non-operating system files and applications, only update the WPARs you need.

SUMA

Service Update Management Assistant (SUMA) is a way to automatically download AIX software updates. In an environment without WPARs, SUMA lets you schedule the download of fixes for later update. When you have an

environment with system WPARs, you must make the same considerations as for manual WPAR updates: same kernel in all WPARs, same operating system level, and different file systems.

Because the download of updates does not interfere with the running WPARs, you can schedule the update download using `smit suma_task_new`. But, you must only perform the update in one place to avoid different levels of downloaded updates. If you have a system WPAR with mobility capabilities, this is a good candidate for a SUMA server, because it improves its availability. If, however, there is no system WPAR with mobility, then the best place to configure SUMA is in the global environment.

5.2.5 Alternate disk installation

One method to clone regular AIXs is to use `smit alt_install`. In the global environment, the `alt_disk_install` command or the previous SMIT fast path work in the same manner as in a regular AIX. It will restore an mksysb, create a clone, or work in conjunction with NIM to help you in a migration scenario.

System WPARs do not have physical disks. Alternate disk installation depends on disks. This means that you cannot use `alt_disk_install` to restore, clone, or migrate WPARs. For more information about restoring and cloning WPARs, check 5.4, “Backup, restore, and cloning” on page 102.

5.3 File system considerations

At creation time, a WPAR can have several types of file systems: namefs, jfs, jfs2, or nfs. By default, the system creates `/`, `/home`, `/tmp`, and `/var` as jfs2 and `/usr`, `/opt`, and `/proc` as namefs. If you want more file systems for your applications, you need to create them. Inside the WPAR, you can mount nfs but you cannot create, change, or delete file systems. You need to do it from the global environment.

5.3.1 Creating a file system for a running WPAR

The best way to add a file system to an existing WPAR is from the CLI. To make it permanent to a WPAR, specify the mount group with the name of that WPAR. In Example 5-32, we show the command to create a file system that will be mounted on /big of the MySysWpar WPAR.

Example 5-32 Creating a file system for a running WPAR

```
# crfs -v jfs2 -g datavg -m /wpars/MySysWpar/big -u MySysWpar -a
logname=INLINE -a size=1G
File system created successfully.
1040148 kilobytes total disk space.
New File System size is 2097152
```

Note: The file system mount point in the global environment has to be under the base directory for the WPAR in order for that mount to be visible within the WPAR.

The **-u** flag indicates the mount group. In case you want to use SMIT, there is an additional step, because **smit crjfs2std** does not allow you to specify the mount group. After the creation of the file system, change it using **smit chjfs2** so that it has the name of the WPAR in the **Mount GROUP** option.

The file system stanza in /etc/filesystems shows the type equal to the name of the WPAR, as shown in Example 5-33. This is the mount group.

Example 5-33 /etc/filesystems stanza of a WPAR's file system

```
/wpars/MySysWpar/big:
    dev           = /dev/fs1v10
    vfs           = jfs2
    log           = INLINE
    mount        = false
    type         = MySysWpar
    account      = false
```

After the creation, mount the new file system. From now on, stopping and starting the WPAR handles this file system's mounts automatically.

At creation time of a WPAR, use the **mkwpar -M** command as explained in "Other file systems" on page 77.

Note: Automount is not supported for WPAR's file systems. **Startwpar** mounts all file systems of the WPAR before booting the WPAR. At that stage, a WPAR automount daemon is not running. Although for user-defined file systems, automount might work, it is not supported.

5.3.2 Changing a file system

In order to change a file system, you must be in the global environment. The changes you make are effective immediately to the WPAR.

Note: Plan your changes before executing them.

In Example 5-34, we use the CLI to increase the size of a WPAR's file system, but you can use SMIT as well.

Example 5-34 Increasing a WPAR's file system size

```
# hostname
braker.itsc.austin.ibm.com
# uname -w
0
# chfs -a size=+512M /wpars/MySysWpar/big
Filesystem size changed to 3145728
# clogin MySysWpar
# df -k
```

Filesystem	1024-blocks	Free	%Used	Iused	%Iused	Mounted on
/dev/fs1v00	65536	47220	28%	1633	14%	/
/dev/fs1v01	32768	31904	3%	5	1%	/home
/opt	131072	41052	69%	1352	13%	/opt
/proc	-	-	-	-	-	/proc
/dev/fs1v02	98304	96608	2%	13	1%	/tmp
/usr	1474560	113896	93%	32473	54%	/usr
/dev/fs1v03	131072	121036	8%	353	2%	/var
/dev/fs1v04	1572864	1564104	1%	4	1%	/big

In Figure 5-3, we use SMIT to change the mount point of the WPAR's /big file system to /local.

```

Change / Show Characteristics of an Enhanced Journaled File System

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

[Entry Fields]
File system name          /wpars/MySysWpar/big
NEW mount point          [/wpars/MySysWpar/local]
SIZE of file system
    Unit Size              512bytes          +
    Number of units        [3145728]          #
Mount GROUP               [MySysWpar]
Mount AUTOMATICALLY at system restart?  no          +
PERMISSIONS               read/write      +
Mount OPTIONS             []                +
Start Disk Accounting?    no          +
Block Size (bytes)        4096
Inline Log?               yes
Inline Log size (MBytes)  [8]          #
Extended Attribute Format  [v1]
ENABLE Quota Management?  no          +
Allow Small Inode Extents? no          +
Enable EFS?               no          +

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

```

Figure 5-3 smit chjfs2: Changing mount point

If the file system is mounted, unmount it. Now, mount it on the new mount point.

5.3.3 Shared /usr with writable file system

Default system WPARs have shared /usr and /opt. As we have seen, they are read-only, but if you only need to write to a local directory under /usr, then you can use the shared /usr and be able to write. In this case, create a new file system in the global environment and mount it under the WPAR's structure. Then, go to the global environment's /usr and make a soft link to that file system. In Example 5-35 on page 99, we show you how to do this.

Note: The soft link refers to the WPAR's vision of the directory.

Because the soft link is a file with a pointer to a file or directory and inside the WPAR, that directory is different from the one in the global environment or other WPARs. The soft link points to different things whether it is read inside or outside the WPAR. For example, `/wpars/MySysWpar/local` in the global environment is `/local` inside the `MySysWpar` WPAR.

One good way to create a writable `/usr/local` under a shared `/usr` is to create a mount point for `/usr/local` in the global environment, then create a `/usr/local` file system within each WPAR.

Example 5-35 Creating a writable subdirectory in a read-only /usr

```
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# crfs -v jfs2 -g datavg -m /wpars/MySysWpar/local -u MySysWpar -a logname=INLINE -a
size=1G
File system created successfully.
1040148 kilobytes total disk space.
New File System size is 2097152
# mount /wpars/MySysWpar/local
# mount
```

node	mounted	mounted over	vfs	date	options
/dev/hd4	/		jfs2	May 02 17:21	rw,log=/dev/hd8
/dev/hd2	/usr		jfs2	May 02 17:21	rw,log=/dev/hd8
/dev/hd9var	/var		jfs2	May 02 17:21	rw,log=/dev/hd8
/dev/hd3	/tmp		jfs2	May 02 17:21	rw,log=/dev/hd8
/dev/hd1	/home		jfs2	May 02 17:21	rw,log=/dev/hd8
/proc	/proc		procfs	May 02 17:21	rw
/dev/hd10opt	/opt		jfs2	May 02 17:21	rw,log=/dev/hd8
/dev/fs1v00	/wpars/MySysWpar		jfs2	May 03 11:41	rw,log=INLINE
/dev/fs1v01	/wpars/MySysWpar/home		jfs2	May 03 11:41	rw,log=INLINE
/opt	/wpars/MySysWpar/opt		namefs	May 03 11:41	ro
/proc	/wpars/MySysWpar/proc		namefs	May 03 11:41	rw
/dev/fs1v02	/wpars/MySysWpar/tmp		jfs2	May 03 11:41	rw,log=INLINE
/usr	/wpars/MySysWpar/usr		namefs	May 03 11:41	ro
/dev/fs1v03	/wpars/MySysWpar/var		jfs2	May 03 11:41	rw,log=INLINE
/dev/fs1v04	/wpars/MySysWpar/local		jfs2	May 03 14:17	rw,log=INLINE

```
# cd /usr
# ln -s /local /usr/local
```

```
# ls -l |grep local
lrwxrwxrwx  1 root  system          6 May 03 14:18 local -> /local
# clogin MySysWpar
# hostname
rutland
# cd /usr
# ls -l |grep local
lrwxrwxrwx  1 root  system          6 May 03 14:18 local -> /local
# mount
  node          mounted          mounted over  vfs      date          options
-----
  /dev/fs1v00   /                  /             jfs2     May 03 11:41  rw,log=INLINE
  /dev/fs1v01   /home             /             jfs2     May 03 11:41  rw,log=INLINE
  /opt          /opt              /             namefs   May 03 11:41  ro
  /proc        /proc            /             namefs   May 03 11:41  rw
  /dev/fs1v02   /tmp              /             jfs2     May 03 11:41  rw,log=INLINE
  /usr         /usr              /             namefs   May 03 11:41  ro
  /dev/fs1v03   /var              /             jfs2     May 03 11:41  rw,log=INLINE
  /dev/fs1v04   /local            /             jfs2     May 03 14:17  rw,log=INLINE
```

5.3.4 Enabling EFS in WPAR

Encrypted file systems (EFS) allow you to have protected files that no one else but you can decode. This technology uses cryptographic libraries, public key infrastructure (PKI) technology, and keystores that you need know about. For further details about how to administer an efs-enabled system, refer to *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430. In a WPAR environment, first load the kernel extensions for encryption both in the global environment and the WPAR. Then, create the necessary users and their keystores in the WPAR.

First, we make the system efs-enabled, starting by installing the necessary filesets: **opnssl** and **cl ic.rte** - CryptoLite for C. Then, we must synchronize the installed software: **syncwpar -A**. Now that we have installed the software, we load the kernel extensions and initialize the key manager in the global environment.

Example 5-36 Enabling a system to have efs

```
# efsenable -a
Enter password to protect your initial keystore:
Enter the same password again:
```

We must also do it for each WPAR that is going to use efs.

The users created after this will be able to have private encrypted files in an efs-enabled file system.

To enable a file system to be encrypted, use:

```
# chfs -a efs=yes /wpars/MySysWpar/local
```

Then, you can set the inheritance for that file system or you can leave it as it is and let the users decide which files to encrypt.

```
# efsmgr -s -E /local
# efsmgr -L /local
EFS inheritance is set with algorithm: AES_128_CBC
```

Afterwards, each user inside the WPAR must authenticate/load the encryption keys in order to encrypt/decrypt the files in the efs. The best way is to associate your shell with your keys so that each command spawn in that shell can encrypt and decrypt your files.

Example 5-37 Associate encryption keys with a shell

```
# hostname
rutland
# su - pac
$ efskeymgr -V
There is no key loaded in the current process.
$ efskeymgr -o ksh
pac's EFS password:
$ efskeymgr -V
List of keys loaded in the current process:
Key #0:
  Kind ..... User key
  Id (uid / gid) ..... 204
  Type ..... Private key
  Algorithm ..... RSA_1024
  Validity ..... Key is valid
  Fingerprint .....
7e30259f:c559cb15:dc25d693:ab95b0a3:300efd91
$ cd /local
$ ls -lrU
total 16
drwxr-xr-x  2 root  system      256 May 03 14:17 lost+found
-rw-r--r--  1 pac  staff        50 May 03 16:29 ibm.test
$ cat ibm.test
this file is encrypted.
Only user pac can see it.
$ whoami
```

```
pac
$ exit
$ exit
# whoami
root
# cd /local
# cat ibm.test
cat: 0652-050 Cannot open ibm.test.
```

Each WPAR has its own users. This means that encrypted file systems cannot move from one WPAR to another WPAR unless you export and import the different keys of the owners of the files.

5.3.5 Network File System (NFS) considerations

To share a directory or an entire file system, you must be in the global environment. NFS serving is not supported inside the WPAR. Also, remember not to share directories that belong to WPARs, because the WPARs can be stopped or moved, leaving an empty directory or simply removing the mount point.

The NFS client is supported in the global environment and inside the WPARs. NFS V4 is supported.

5.4 Backup, restore, and cloning

When the system administrator creates a backup through the `mksysb` command, the system administrator usually sends it to a physical device. In a WPAR environment, there are no physical devices for backup, which means that there is a different way to back up a WPAR.

5.4.1 Backup

Similar to the `savevg` command, we can make a backup of the WPAR with the `savewpar` command. This saves the files and the configuration of the WPAR. When you have a system WPAR with shared `/usr` and `/opt`, the backup is very small, because it does not save those file systems. You must be in the global environment to execute this backup. See Example 5-38 on page 103.

Example 5-38 Backing up a system WPAR

```
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# savewpar -f /backups/MySysWpar.bk MySysWpar
Creating information file for workload partition MySysWpar.
Creating list of files to back up.
Backing up 2208 files
2208 of 2208 files (100%)
0512-038 savewpar: Backup Completed Successfully.
```

You can also back up to tape, CD, or DVD. For CD, use `mkcd` command. For DVD, use the `mkdvd` command.

One option is the `-Z` flag. It allows you to skip encrypted files.

5.4.2 Restore

The restore of a wpar needs no flags if you accept the defaults. This also means having the backup on a tape and on the default tape drive. In most cases, you have a file as a backup and want to restore that file. In order to do this, use the `-f` flag as shown in Example 5-39. The `restwpar` command has three major steps:

1. Create the necessary file systems according to the `image.data` file that is created with the `savewpar` command and mount them.
2. Restore the files in the backup to their proper places. This might include the `/usr` and `/opt` depending on the type of WPAR.
3. Synchronize the WPAR with the global environment.

Example 5-39 Restoring a WPAR backup file

```
# restwpar -f /backups/MySysWpar.bk
New volume on /backups/MySysWpar.bk:
Cluster size is 51200 bytes (100 blocks).
The volume number is 1.
The backup date is: Fri May 4 11:47:14 CDT 2007
Files are backed up by name.
The user is root.
x      2974 ./savewpar_dir/wpar.spec
x      6404 ./savewpar_dir/image.data
x     143741 ./savewpar_dir/backup.data
The total size is 153119 bytes.
The number of restored files is 3.
```

```

mkwpar: Creating filesystems...
/
Creating logical volume 'fslv00' specified in image.data
Creating file system '/' specified in image.data
/home
Creating logical volume 'fslv01' specified in image.data
Creating file system '/home' specified in image.data
/local
Creating logical volume 'fslv04' specified in image.data
Creating file system '/local' specified in image.data
/opt
/proc
/tmp
Creating logical volume 'fslv02' specified in image.data
Creating file system '/tmp' specified in image.data
/usr
/var
Creating logical volume 'fslv03' specified in image.data
Creating file system '/var' specified in image.data
populate: Mounting all workload partition file systems
New volume on /backups/MySysWpar.bk:
Cluster size is 51200 bytes (100 blocks).
The volume number is 1.
The backup date is: Fri May 4 11:47:14 CDT 2007
Files are backed up by name.
The user is root.
x      136 ./savewpar_dir/data.files741496
x      136 ./savewpar_dir/vgdata.files
x     4514 ./savewpar_dir/filesystems
x     2974 ./savewpar_dir/wpar.spec
x       15 ./savewpar_dir/image.info
x     6404 ./savewpar_dir/image.data
x    143741 ./savewpar_dir/backup.data
...
The total size is 21725320 bytes.
The number of restored files is 2166.
syncroot: Processing root part installation status.
syncroot: Synchronizing installp software.
+-----+
Pre-installation Verification...
+-----+
Verifying selections...done
Verifying requisites...done
Results...
...

```

mkwpar: Workload partition MySysWpar created successfully.
To start the workload partition, execute the following as root: startwpar [-v]
'MySysWpar'

The **restwpar** command has several flags, which allow you to modify the behavior of the restore based on whether you want to change the base directory or just want to start it after the restore. Several of the flags are similar to the **chwpar** and **mkwpar** commands:

- a** Automatically resolves erroneous or conflicting settings if required
- A** Starts the WPAR automatically at system restart
- b <devexportsfile>** Alternative file for devexports
- d <directory>** Base directory for this WPAR
- h <hostname>** New host name
- M <flags mkwpar>** Extra flags to pass to the **mkwpar** command
- n <name>** New name of the WPAR
- r** Copies the network name resolution from the global environment
- s** Start after restoring
- w <wpar specification file>** Specification file with -M type of contents to pass to **mkwpar** as an alternative to those defined in the backup

There also flags that are similar to the flags from the AIX **restore** command:

- B <number of blocks>** Number of 512-byte blocks to read in a single input operation
- f <device or file>** Device name or file that contains the backup. The default is /dev/rmt0.

Additionally, **restwpar** has specific flags:

- C** Forces the creation of the WPAR even if the compatibility checks fail. This means that if you are restoring in a system that is incompatible with the system where you did the backup, then the restore will not fail due to compatibility issues. However, forcing the restore does not make the system compatible. The WPAR might not function properly after forced restores.

- F** Forces the creation of the named WPAR. If the named WPAR already exists, then it will stop the WPAR, remove it, and then deploy the backed up WPAR.
- i <image.data file name>** Alternative image.data to pass to the restore instead of the one provided by the backup file
- k** Creates the logical volumes (LVs) with the minimum size needed for the restore
- S** Type of synchronization after restore:
 - Sa** Only required installation without removing software. This is the default.
 - SA** Same as **-a** but ignores errors. Can bring the WPAR to an unusable state
 - Sf** Total synchronization, including rejection and deinstallation of software
 - SF** Same as **-f** but ignores errors. Can bring the WPAR to an unusable state
 - Sn** No synchronization. Can bring the WPAR to an unusable state
- U** Ignores MAP files

5.4.3 Cloning

To clone a WPAR, use the **savewpar** and **restwpar** commands. In Example 5-40, we take one working WPAR named `MySysWpar` and clone it in the same global environment. The **savewpar** command is the same as in Example 5-38 on page 103. First, we have to plan the WPAR's name. Secondly, check the network settings, especially the IP address for the name that we chose, `cloneWpar`, and for the IP address, `9.3.5.146`. Last, because the WPAR is in the same global environment, we have to choose a different mount point: `/wpars/cloneWpar`. The **restwpar** command tries to restore the file systems to logical volumes with the name specified in the `image.data` of the backup file. In the example, these LVs already exist. The restore detects and corrects this using different unique LV names.

Example 5-40 Cloning a system WPAR

```
# restwpar -n cloneWpar -h metric -d /wpars/cloneWpar -r -M "-N address=9.3.5.146" -U
-f /backups/MySysWpar.bk
New volume on /backups/MySysWpar.bk:
Cluster size is 51200 bytes (100 blocks).
```

The volume number is 1.
The backup date is: Mon May 7 11:44:21 CDT 2007
Files are backed up by name.
The user is root.
x 2974 ./savewpar_dir/wpar.spec
x 6404 ./savewpar_dir/image.data
x 143810 ./savewpar_dir/backup.data
The total size is 153188 bytes.
The number of restored files is 3.
mkwpar: Creating filesystems...

ATTENTION: Logical volume 'fslv00' is not unique. Renaming to 'wlv0'.

Creating logical volume 'wlv0' specified in image.data

Creating file system '/' specified in image.data

/home

ATTENTION: Logical volume 'fslv01' is not unique. Renaming to 'wlv1'.

Creating logical volume 'wlv1' specified in image.data

Creating file system '/home' specified in image.data

/local

ATTENTION: Logical volume 'fslv04' is not unique. Renaming to 'wlv2'.

Creating logical volume 'wlv2' specified in image.data

Creating file system '/local' specified in image.data

/opt

/proc

/tmp

ATTENTION: Logical volume 'fslv02' is not unique. Renaming to 'wlv3'.

Creating logical volume 'wlv3' specified in image.data

Creating file system '/tmp' specified in image.data

/usr

/var

ATTENTION: Logical volume 'fslv03' is not unique. Renaming to 'wlv4'.

Creating logical volume 'wlv4' specified in image.data

Creating file system '/var' specified in image.data

populate: Mounting all workload partition file systems

New volume on /backups/MySysWpar.bk:

Cluster size is 51200 bytes (100 blocks).

The volume number is 1.

The backup date is: Mon May 7 11:44:21 CDT 2007

Files are backed up by name.

The user is root.

x 136 ./savewpar_dir/data.files471158

x 136 ./savewpar_dir/vgdata.files

x 4514 ./savewpar_dir/filesystems

x 2974 ./savewpar_dir/wpar.spec

x 15 ./savewpar_dir/image.info

```

x          6404 ./savewpar_dir/image.data
x        143810 ./savewpar_dir/backup.data
x           0 ./var
x           0 ./var/lost+found
x           0 ./var/adm
x           0 ./var/adm/cron
x           4 ./var/adm/cron/at.deny
x           3 ./var/adm/cron/cron.deny
x          186 ./var/adm/cron/log
x          1317 ./var/adm/cron/queuedefs
x           0 ./var/adm/cron/FIFO
...
Copying network name resolution configuration...
    /etc/resolv.conf
    /etc/hosts
    /etc/netsvc.conf
mkwpar: Workload partition cloneWpar created successfully.
To start the workload partition, execute the following as root: startwpar [-v]
'cloneWpar'
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    A      S     rutland  /wpars/MySysWpar
cloneWpar    D      S     metric   /wpars/cloneWpar
# startwpar cloneWpa
...
startwpar: Return Status = SUCCESS
# clogin cloneWpar
# hostname
metric
# ifconfig en0
en0:
flags=1e080863,480<UP,BROADCAST,NOTRAILERS,RUNNING,SIMPLEX,MULTICAST,GROUPRT,64BIT,CH
ECKSUM_OFFLOAD(ACTIVE),CHAIN>
    inet 9.3.5.146 netmask 0xfffffe00 broadcast 9.3.5.255
    tcp_sendspace 262144 tcp_recvspace 262144 rfc1323 1

```

You can clone a WPAR to a different host. In Example 5-41 on page 109, we transfer the backup file to a different system and restore it there. In the origin system, braker, there are two VGs, rootvg and datavg, and one of the WPAR's file systems resides on datavg. The new system is parmer, but it only has one VG: rootvg. We need to edit the image.data or include the new LVs in the CLI when invoking the **restwpar** command. We show you how to extract image.data from the backup and after editing it, use it on the restore.

```
# hostname
parmer.itsc.austin.ibm.com
# lswpar
# restwpar -f /MySysWpar.bk
...
0516-306 getlvodm: Unable to find volume group datavg in the Device
Configuration Database.
0516-822 mklv: Unable to create logical volume.
bosinst_mkobject: failed command: /usr/sbin/mklv -o n -L /local -u 32 -r y -b y -d p
-v n -s y -w a -a m -e m -c 1 -x 512 -t jfs2 -y wlv2 datavg 128 hdisk3
wparidata: Error processing image.data
restwpar: 0960-514 /usr/sbin/mkwpar exited with return code 1.
# restore -xf /MySysWpar.bk ./savewpar_dir/image.data
Please mount volume 1 on /MySysWpar.bk.
Press the Enter key to continue.

x ./savewpar_dir/image.data
# vi ./savewpar_dir/image.data
...
# restwpar -i ./savewpar_dir/image.data -f /MySysWpar.bk
New volume on /MySysWpar.bk:
Cluster size is 51200 bytes (100 blocks).
The volume number is 1.
The backup date is: Mon May 7 14:19:10 CDT 2007
Files are backed up by name.
The user is root.
x          2974 ./savewpar_dir/wpar.spec
x          6404 ./savewpar_dir/image.data
x         143810 ./savewpar_dir/backup.data
The total size is 153188 bytes.
The number of restored files is 3.
mkwpar: Creating filesystems...
/
ATTENTION: Logical volume 'fslv00' is not unique. Renaming to 'wlv0'.
Creating logical volume 'wlv0' specified in image.data
Creating file system '/' specified in image.data
/home
ATTENTION: Logical volume 'fslv01' is not unique. Renaming to 'wlv1'.
Creating logical volume 'wlv1' specified in image.data
Creating file system '/home' specified in image.data
/local
ATTENTION: Logical volume 'fslv04' is not unique. Renaming to 'wlv2'.
Creating logical volume 'wlv2' specified in image.data
```

Creating file system '/local' specified in image.data

/opt
/proc
/tmp

...

mkwpar: Workload partition MySysWpar created successfully.

To start the workload partition, execute the following as root: startwpar [-v]

'MySysWpar'

lswpar

Name	State	Type	Hostname	Directory
------	-------	------	----------	-----------

MySysWpar	D	S	rutland	/wpars/MySysWpar
-----------	---	---	---------	------------------

startwpar MySysWpar

Starting workload partition MySysWpar.

Mounting all workload partition file systems.

Loading workload partition.

Exporting workload partition devices.

Starting workload partition subsystem cor_MySysWpar.

0513-059 The cor_MySysWpar Subsystem has been started. Subsystem PID is 430190.

Verifying workload partition startup.

mount

node	mounted	mounted over	vfs	date	options
/dev/hd4	/		jfs2	May 02 17:25	rw,log=/dev/hd8
/dev/hd2	/usr		jfs2	May 02 17:25	rw,log=/dev/hd8
/dev/hd9var	/var		jfs2	May 02 17:25	rw,log=/dev/hd8
/dev/hd3	/tmp		jfs2	May 02 17:25	rw,log=/dev/hd8
/dev/hd1	/home		jfs2	May 02 17:25	rw,log=/dev/hd8
/proc	/proc		procfs	May 02 17:25	rw
/dev/hd10opt	/opt		jfs2	May 02 17:25	rw,log=/dev/hd8
/dev/wlv0	/wpars/MySysWpar		jfs2	May 07 14:39	rw,log=INLINE
/dev/wlv1	/wpars/MySysWpar/home		jfs2	May 07 14:39	rw,log=INLINE
/dev/wlv2	/wpars/MySysWpar/local		jfs2	May 07 14:39	rw,log=INLINE
/opt	/wpars/MySysWpar/opt		namefs	May 07 14:39	ro
/proc	/wpars/MySysWpar/proc		namefs	May 07 14:39	rw
/dev/wlv3	/wpars/MySysWpar/tmp		jfs2	May 07 14:39	rw,log=INLINE
/usr	/wpars/MySysWpar/usr		namefs	May 07 14:39	ro
/dev/wlv4	/wpars/MySysWpar/var		jfs2	May 07 14:39	rw,log=INLINE

Another way to clone a WPAR is through the WPAR Manager commands. For more information about this topic, consult 6.12, "Introduction to mobility command-line interface" on page 230.

The **savewpar** command creates the image.data file under `/.savewpar_dir` in the WPAR environment as part of the **savewpar** command as **savevg** or **mksysb** do in the global environment.

In the same way that **mkszfile** creates the image.data file of a global environment, **mkwpardata** does it for the WPAR environment. Both commands must be run in the global environment.

```
# mkwpardata MySysWpar
```

The **lssavewpar** command lists the contents of a WPAR's backup.

5.5 Users and groups

In a WPAR environment, the user management depends on whether you have a system or an application WPAR. Because application WPARs use the same file systems as the global environment, they have the same authentication files and methods. Managing users and groups in the global environment is the same as with regular AIX.

5.5.1 Default access: Users and root

In system WPARs, you can manage users and groups inside the WPAR. You can use SMIT or the CLI. The users inside a system WPAR are not related to the users of the global environment. There is a root user for the global environment and another root user for each of the WPARs. In the same way, if the user `pac` has id 204 inside `wparA`, and there is another user `pac` with the same id in `wparB` in the same global environment, they cannot interfere with each other's processes and files.

The root user inside a WPAR only has access to the virtual operating system (OS). In a default system, WPAR `/usr` and `/opt` are read-only; thus, the root user inside the WPAR is unable to delete files from these directories. If, by mistake, the root user inside `wparA` removes all files that the user is allowed to remove, the user is only removing what is under `/wpars/wparA` of the global environment. At this point, the user must restore a previous backup of that WPAR or recreate the WPAR, but the user does not affect the global environment and the other WPARs.

Management of users and groups inside a system WPAR is basically the same as in the global environment.

In Example 5-42, we show how to create a user inside the MySysWpar WPAR and also that the user we create inside the MySysWpar WPAR does not exist in the global environment.

Example 5-42 Create a user inside a WPAR only

```
# hostname
rutland
# uname -W
87
# ls -l /wpars
total 0
lrwxrwxrwx  1 root    system          1 May 08 09:47 MySysWpar -> /
# mkuser pac
# lsuser pac
pac id=205 pgrp=staff groups=staff home=/home/pac shell=/usr/bin/ksh login=true
su=true rlogin=true daemon=true admin=false sugroups=ALL admgroups= tpath=nosak
ttys=ALL expires=0 auth1=SYSTEM auth2=NONE umask=22 registry=files SYSTEM=compat
logintimes= loginretries=0 pwdwarntime=0 account_locked=false minage=0 maxage=0
maxexpired=-1 minalpha=0 minother=0 mindiff=0 maxrepeats=8 minlen=0 histexpire=0
histsize=0 pwdchecks= dictionlist= default_roles= fsize=2097151 cpu=-1 data=262144
stack=65536 core=2097151 rss=65536 nofiles=2000 roles=
# exit
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# lsuser pac
3004-687 User "pac" does not exist.
```

5.5.2 Recover root password

If the owner of a system WPAR loses the root password, the user can ask the global environment's system administrator to reset it. Log in to the global environment as **root** and **chroot** to the proper WPAR:

```
# chroot /wpars/MySysWpar /bin/ksh
```

Use the **passwd** command to change the password:

```
# passwd
Changing password for "root"
root's New password:
Enter the new password again:
# exit
```

For the global environment's recovering procedures, use the standard AIX recover procedures, such as booting from CD-ROM or NIM.

5.5.3 Role-based access control (RBAC)

Having one user to manage all of the regular administrative tasks of a operating system can, in certain environments, pose a challenge, especially if the business rules require the restriction of system administration tasks to a subset of operations. Delegating tasks to different users facilitates the execution of those tasks, prevents some mistakes, and limits the damage of others.

Role Based Access Control is a framework that allows the delegation of administration tasks through roles. Each user is assigned one or more roles, thus allowing that user to perform several administration tasks without being the super user root. Examples of roles are:

- ManageBackup** Which allows the user to back up files
- ManageShutdown** Which allows the user to shut down the system
- ManageRoles** Which allows the user to manage the roles

Looking into the roles, we see that they are a set of authorizations that allow privileged actions to the users and objects to which they are assigned. Examples of authorizations are:

- aix.network.config** Allows the user to configure the network
- aix.network.config.no** Allows the user to configure the network options only
- backup** Allowsthe user to back up

For each process and command, there is a set of privileges needed to bypass the system restrictions. Examples of privileges are:

- PV_FS_CHOWN** Needed to change owner
- PV_KER_WPAR** Needed to work with WPAR

When issuing the command **mkwpar**, it reports to the system that it needs several privileges to run:

- innateprivs** PV_DAC_O, PV_DAC_R, PV_DAC_W, PV_DAC_X, PV_FS_CHOWN, PV_KER_RAC, PV_NET_CNTL, PV_NET_PORT, PV_PROC_PRIV
- inheritprivs** PV_AU_ADD, PV_AU_PROC, PV_AZ_CHECK, PV_AZ_ROOT, PV_DAC_GID, PV_DAC_O, PV_DAC_R, PV_DAC_UID, PV_DAC_W, PV_DAC_X, PV_DEV_LOAD, PV_DEV_QUERY, PV_FS_CHOWN,

PV_FS_MKNOD, PV_FS_MOUNT, PV_KER_ACCT,
PV_KER_LVM, PV_KER_RAC, PV_KER_VARS,
PV_KER_WLM, PV_NET_CNTL, PV_NET_PORT,
PV_PROC_PRIV, PV_PROC_TIMER, PV_SU_UID,
PV_TCB

If the user issuing the **mkwpar** command has the proper access authorization, such as **aix.wpar.system.create**, then it is possible to create the WPAR without being a super user.

WPARs have private users and groups, which have running processes and files. For RBAC, this means that WPARs can restrict even further the delegation of tasks. Each WPAR has its own set of allowed privileges. We refer to it as the *WPAR privilege set (WPS)*. The commands and processes inside a WPAR get, at most, the inherited privileges of that WPS. Its users can only execute commands that fit in those sets and that they are authorized to use. This means that even though the root user in the global environment is able to run the **mount** command, the root user inside the WPAR does not have the **PV_FS_MOUNT** in the WPS.

In Example 5-43, we remove the privileges from the WPS, then we list them, and finally, we add the privileges again.

Example 5-43 Removing, listing, and adding privileges to WPS

```
# chwpar -S privs--PV_FS_MOUNT MySysWpar
# lswpar -S MySysWpar
=====
MySysWpar - Defined
=====
Privileges:  PV_AU_,PV_AU_ADD,PV_AU_ADMIN,PV_AU_READ,
              PV_AU_WRITE,PV_AU_PROC,PV_AZ_ADMIN,PV_AZ_READ,
              PV_AZ_ROOT,PV_AZ_CHECK,PV_DAC_,PV_DAC_R,
              PV_DAC_W,PV_DAC_X,PV_DAC_O,PV_DAC_UID,
              PV_DAC_GID,PV_DAC_RID,PV_FS_MKNOD,PV_FS_CHOWN,
              PV_FS_QUOTA,PV_FS_LINKDIR,PV_FS_CNTL,
              PV_FS_CHROOT,PV_PROC_PRIO,PV_PROC_CORE,
              PV_PROC_RAC,PV_PROC_ENV,PV_PROC_CKPT,
              PV_PROC_CRED,PV_PROC_SIG,PV_PROC_PRIV,
              PV_PROC_TIMER,PV_PROC_RTCLK,PV_PROC_VARS,
              PV_TCB,PV_TP,PV_TP_SET,PV_KER_ACCT,PV_KER_DR,
              PV_KER_RAC,PV_KER_EWLM,PV_KER_REBOOT,
              PV_KER_LVM,PV_KER_NFS,PV_KER_CONF,
              PV_KER_EXTCONF,PV_KER_IPC,PV_KER_IPC_R,
              PV_KER_IPC_W,PV_KER_IPC_O,PV_DEV_CONFIG,
              PV_DEV_QUERY,PV_SU_UID,PV_NET_PORT,PV_MIC,
              PV_MIC_CL,PV LAB_,PV LAB_CL,PV LAB_CLTL,
```

```
PV_LAB_LEF,PV_LAB_SLDG,PV_LAB_SLDG_STR,  
PV_LAB_SL_FILE,PV_LAB_SL_PROC,PV_LAB_SL_SELF,  
PV_LAB_SLUG,PV_LAB_SLUG_STR,PV_LAB_TL,PV_MAC_,  
PV_MAC_CL,PV_MAC_R,PV_MAC_R_CL,PV_MAC_R_STR,  
PV_MAC_R_PROC,PV_MAC_W,PV_MAC_W_CL,  
PV_MAC_W_DN,PV_MAC_W_UP,PV_MAC_W_PROC,  
PV_MAC_OVRRD  
# chwpar -S privs+=PV_FS_MOUNT MySysWpar
```

For more information about Role Based Access Control, consult *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430.

5.6 Print spooling considerations

Printing from inside a WPAR is supported with restrictions.

Because there are no physical devices inside a system WPAR, printers cannot be local. You cannot connect parallel or serial printers directly to the WPAR. Because the global environment can manage local printers, and remote printers work inside the WPARs, you can use them. If you only have a parallel or serial printer, then:

1. In the global environment:
 - a. Connect the printer to the appropriate port (parallel or serial).
 - b. Install the appropriate software and device drivers.
 - c. Create a print queue.
 - d. Add the WPAR's host name to the print server: Add Print Access for a Remote Client.
 - e. Start the print server.
2. Inside the WPAR:
 - a. Create a remote print queue where:
 - Name of QUEUE to add**
The WPAR's queue name
 - HOSTNAME of remote server**
The host name of the global environment
 - Name of QUEUE on remote server**
The global environment's queue name

After creating the print queue inside the WPAR, you can see its status with the command shown in Example 5-44.

Example 5-44 WPAR's print queues status

```
# lpstat
Queue  Dev  Status  Job Files          User          PP %  Blks  Cp Rnk
-----
PScript @brak  READY
PScript PSe   READY
```

For system WPARs with shared read-only /usr and /opt, install the print software and drivers at the global environment level and then synchronize it with the WPARs. For system WPARs with private read and write /usr and /opt, you must install the printing software and printers' drivers inside the WPAR. Managing queues inside a WPAR is otherwise similar to regular AIX.

Application WPARs use the global environment's print queues. Manage them as regular AIX at the global environment.

5.7 System environments

Some of the usual customizations made in AIX, such as changing the date or the language, pose new challenges when considering WPARs:

- ▶ The console of a WPAR is the global environment where it runs. If the proper daemons are running to support these types of connections, you can log in to a WPAR using **clogin** or a remote mechanism, such as **rsh**, **telnet**, **rlogin**, or **ssh**.
- ▶ You cannot change the date or the time inside a WPAR, because there is only one system time. You can change the time zone for each WPAR. You can have one WPAR in America/Chicago (Central time) and another in Europe/Lisbon.
- ▶ WPARs support different languages. Install the language sets in the global environment if the system WPAR is using shared /usr and /opt or install the language sets in the WPAR if it has private /usr and /opt.
- ▶ Changing the characteristics of the operating system inside the WPAR, such as the number of processes allowed per user or the maximum number of pages in the block I/O buffer cache, is not allowed. These characteristics are inherited from the global environment.

- ▶ Dump devices do not exist. It is not possible to generate a dump inside a WPAR. In the global environment, you can change dump devices and generate dumps.
- ▶ The AIX Security Expert option in SMIT helps the system administrator to increase the security levels of the operating system by running scripts that modify several files. These files reside in several file systems, some of which might be read-only.

5.8 Processes and subsystems

The system administrator can view the system processes using the normal `ps` command or by using `smit process`. Inside a WPAR, the system administrator can only see the processes that belong to that WPAR. In Example 5-45, we show that we are inside a WPAR and the processes that are visible. You can see that `/etc/init` has `PID=1` and `PPID=0` inside the WPAR.

Example 5-45 Viewing processes inside a WPAR

```
# hostname
MySysWpar
# uname -W
11
# ls -l /wpars
total 0
lrwxrwxrwx  1 root  system          1 May 09 11:00 MySysWpar -> /
# ps -ef
  UID  PID  PPID  C   STIME  TTY  TIME CMD
  root 225306 331830 0 11:32:39 - 0:00 /usr/sbin/rsct/bin/IBM.HostRMd
  root 229586 1 0 14:13:00 ? 0:00 clogin MySysWpar
  root 290882 1 0 11:32:38 - 0:00 /usr/sbin/cron
  root 307204 331830 0 11:33:11 - 0:00 /usr/sbin/rsct/bin/IBM.LPRMd
  root 1 0 0 11:32:18 - 0:00 /etc/init
  root 331830 1 0 11:32:22 - 0:00 /usr/sbin/srcmstr
  root 344118 331830 0 11:32:38 - 0:00 /usr/sbin/syslogd
  root 352496 331830 0 11:32:40 - 0:00 /usr/sbin/rsct/bin/IBM.ServiceRMd
  root 364626 331830 0 11:32:38 - 0:00 /usr/sbin/qdaemon
  root 381150 331830 0 11:32:38 - 0:00 /usr/sbin/writesrv
  root 405648 229586 0 14:13:00 ? 0:00 -ksh
  root 413714 331830 0 11:32:38 - 0:00 sendmail: accepting connections
  root 417792 331830 0 11:32:38 - 0:00 /usr/sbin/portmap
  root 421952 331830 0 11:32:41 - 0:00 /usr/sbin/rsct/bin/IBM.ERmd
  root 430216 331830 0 11:32:40 - 0:00 /usr/sbin/rsct/bin/IBM.AuditRMd
  root 471162 405648 8 14:13:04 ? 0:00 ps -ef
```

```

root 475284 331830 0 11:32:38 - 0:00 /usr/sbin/inetd
root 491588 1 0 11:32:38 - 0:29 /usr/bin/xmwlw -L
root 495820 331830 0 11:32:39 - 0:00 /usr/sbin/rsct/bin/rmcd -a
IBM.LPCommands -r
root 499890 1 0 14:12:28 - 0:00 /usr/sbin/shdaemon
root 512038 331830 0 11:32:41 - 0:00 /usr/sbin/rsct/bin/IBM.DRMd
root 516232 331830 0 11:32:39 - 0:00 /usr/sbin/rsct/bin/IBM.CSMAgentRMd

```

In the global environment, the system administrator has a normal view of all the processes, including the ones that belong to the WPARs. If the system administrator wants to distinguish the processes for each WPAR, the system administrator can use the flag `-@`, which shows a new column with the WPAR's name. In Example 5-46, "Viewing processes in the global environment" on page 118, we show that the `/etc/init` of the WPAR has a different PID in the global environment and that the global environment has its own `/etc/init`.

Example 5-46 Viewing processes in the global environment

```

# ls -l /wpars
total 8
drwxr-xr-x 17 root system 4096 May 09 13:52 MyNewMountPoint
# hostname
braker.itsc.austin.ibm.com
# uname -W
0
# ps -ef@
      WPAR      UID      PID      PPID      C      STIME      TTY      TIME CMD
      Global      root         1         0      0 10:58:53      -      0:00 /etc/init
...
MySysWpar      root 327854 155752      0 11:32:18      -      0:00 /etc/init
...

```

Specifying the `ps` command with the `-@ <name of WPAR>` in the global environment shows the processes for that WPAR only.

The system administrator can start, list, and stop subsystems using the system resource controller (SRC) in the same way that you do in AIX. The SRC works both inside the system WPAR and in the global environment. Each environment has its own SRC and its own subsystems. In Example 5-47 on page 119, after stopping the group spooler of the SRC in both the WPAR and the global environment, we start it inside the WPAR. As we can see, the spooler in the global environment is inoperative.

Example 5-47 Stopping and starting different subsystems with SRC

```
# hostname
braker.itsc.austin.ibm.com
# lssrc -g spooler
Subsystem      Group          PID           Status
qdaemon        spooler        471194        inoperative
writesrv       spooler        507976        inoperative
lpd            spooler        503964        inoperative
# clogin MySysWpar
Last login: Wed May  9 14:44:39 CDT 2007 on /dev/Global from
braker.itsc.austin.ibm.com
# lssrc -g spooler
Subsystem      Group          PID           Status
qdaemon        spooler        471194        inoperative
writesrv       spooler        507976        inoperative
lpd            spooler        503964        inoperative
# startsrc -g spooler
0513-059 The qdaemon Subsystem has been started. Subsystem PID is 471194.
0513-059 The writesrv Subsystem has been started. Subsystem PID is 507976.
0513-059 The lpd Subsystem has been started. Subsystem PID is 503964.
# lssrc -g spooler
Subsystem      Group          PID           Status
qdaemon        spooler        471194        active
writesrv       spooler        507976        active
lpd            spooler        503964        active
# hostname
MySysWpar
# hostname
braker.itsc.austin.ibm.com
# lssrc -g spooler
Subsystem      Group          PID           Status
qdaemon        spooler        471194        inoperative
writesrv       spooler        507976        inoperative
lpd            spooler        503964        inoperative
```

You can also trace a subsystem inside a WPAR.

5.9 Performance and resource schedule

As a rule, the system administrator does all the performance analysis, monitoring, and tuning from the global environment, because most statistics and tuning tools need access to some kind of device, such as the /dev/mem global memory device.

There are several statistics that you can get inside a WPAR. In Appendix A, “Modified AIX commands” on page 263, we provide a list of the commands that were modified and enhanced to work with WPARs.

In Example 5-48, you can see Topas Monitor in the global environment, and in Example 5-49 on page 121, you can see the differences of running `topas` inside a WPAR.

Example 5-48 Topas Monitor in the global environment

```

Topas Monitor for host:   braker
Wed May  9 15:19:03 2007 Interval: 2
EVENTS/QUEUES          FILE/TTY
Cswitch                71  Readch      8086
Syscall                242 Writech      225
Reads                   5  Rawin       0
Writes                   0  Ttyout      225
Forks                    0  Igets       0
Execs                    0  Namei       12
Runqueue                0.0 Dirblk       0
Waitqueue              0.0
Network KBPS I-Pack O-Pack KB-In KB-Out
en0      1.5  22.5   0.5   1.3   0.2
en1      1.2  22.0   0.0   1.2   0.0
lo0      0.0   0.0   0.0   0.0   0.0
Disk Busy% KBPS TPS KB-Read KB-Writ PgpsIn PgpsOut
hdisk1  0.0   0.0   0.0   0.0   0.0
hdisk0  0.0   0.0   0.0   0.0   0.0
hdisk2  0.0   0.0   0.0   0.0   0.0
hdisk3  0.0   0.0   0.0   0.0   0.0
Sios                    0
PAGING
Real,MB                2048
% Comp                  35.2
% Noncomp               10.1
% Client                10.1
PAGING SPACE
Size,MB                 512
% Used                   1.3
% Free                   98.6
FileSystem              KBPS TPS KB-Read KB-Writ NFS (calls/sec)
/usr                    0.0 0.0 0.0 0.0 ServV2 0 WPAR Activ 1
/var                    0.0 0.0 0.0 0.0 Cliv2 0 WPAR Total 2
/tmp                    0.0 0.0 0.0 0.0 ServV3 0 Press: "h"-help
/home                   0.0 0.0 0.0 0.0 Cliv3 0 "q"-quit
/proc                   0.0 0.0 0.0 0.0
/opt                    0.0 0.0 0.0 0.0
/                        0.0 0.0 0.0 0.0
/wpars/..ntPoint       0.0 0.0 0.0 0.0

```

```

/wpars/..int/var 0.0    0.0    0.0    0.0
/wpars/..nt/home 0.0    0.0    0.0    0.0
/wpars/..int/opt 0.0    0.0    0.0    0.0
/wpars/..nt/proc 0.0    0.0    0.0    0.0
/wpars/..int/tmp 0.0    0.0    0.0    0.0
/wpars/..int/usr 0.0    0.0    0.0    0.0

```

```

WLM-Class (Active)   CPU%   Mem%   Blk-I/O%
System                13     20     0
Unmanaged             0      20     0

```

```

Name          PID CPU% PgSp Class
xmwl         315558 12.1 29.6 System
topas        237712  1.9 25.9 System
wlmsched     69666  0.0  0.5 System
gil          65568  0.0  0.9 System
rpc.lock    127186  0.0  1.2 System
sendmail    143538  0.0  1.1 System
aixmibd     98446  0.0  1.2 System
n4bg        188526  0.0  0.4 System
rdpgc       151642  0.0  0.4 System
netm        61470  0.0  0.4 System
xmwl         491588  0.0 28.2 MySysWpar
syncd       180320  0.0  0.5 System

```

In the global environment, the system administrator has the statistics of both the disks and the file systems. The system administrator can also check for the WLM's classes and available networks.

Inside a WPAR, there are no disks, so **topas** does not show them. Also, **topas** only shows what is available to this WPAR, for example, one network interface, as shown in Example 5-49.

Example 5-49 The topas Monitor inside a WPAR

```

Topas Monitor for host: MySysWpar
Wed May 9 15:20:31 2007 Interval: 2
EVENTS/QUEUES  FILE/TTY
Cswitch        4  Readch  33270
Syscall        1047 Writech  1742
Reads          36  Rawin   0
Writes         8   Ttyout  321
Forks          0   Igets   0
Execs          1   Namei   572
Runqueue       0.0 Dirblk   0
Waitqueue      0.0
Network KBPS  I-Pack  O-Pack  KB-In  KB-Out
lo0       0.0    0.0    0.0    0.0    0.0
MEMORY

```

```

en0      0.0    0.0    0.0    0.0    0.0    0.0    PAGING          Real,MB    2048
          Faults      152    % Comp      44.6
FileSystem      KBPS      TPS    KB-Read    KB-Writ    Steals      0    % Noncomp    0.7
/home          0.0      0.0      0.0      0.0      PgpsIn      0    % Client     0.5
/opt           0.0      0.0      0.0      0.0      PgpsOut      0
/              0.0      0.0      0.0      0.0      PageIn       0    PAGING SPACE
/proc          0.0      0.0      0.0      0.0      PageOut      0    Size,MB     512
/var           0.0      0.0      0.0      0.0      Sios         0    % Used      1.3
/tmp           0.0      0.0      0.0      0.0                        % Free     98.6
/usr           0.0      0.0      0.0      0.0
          NFS (calls/sec)
          Serv2       0    Press: "h"-help
          Cliv2       0    "q"-quit
          Serv3       0
          Cliv3       0

Name          PID    CPU%    PgSp
topas         405718  0.4    25.6
init          1      0.3    0.6
shdaemon     499770  0.3    0.2
clogin       381178  0.0    0.1
xmwlm        491588  0.0    28.2
syslogd      344118  0.0    0.3
IBM.Serv     352496  0.0    1.4
cron         290882  0.0    0.5
IBM.Host     225306  0.0    1.4
sendmail     413714  0.0    1.1
ksh          159924  0.0    0.5
portmap      417792  0.0    0.9
IBM.CSMA     516232  0.0    2.4
IBM.ERrm    421952  0.0    1.8
IBM.Audi     430216  0.0    1.6
qdaemon     471194  0.0    0.2
inetd        475284  0.0    0.5
IBM.LPRM    307204  0.0    1.5
rmcd         495820  0.0    2.4
srcmstr     331830  0.0    0.8

```

Tuning tools allow you to set the parameters in the global environment. Inside a WPAR, the system administrator can use commands, such as **ioo**, **vmo**, **no**, **nfso**, **raso**, and **schedo** to display parameter values as shown in Example 5-50.

Example 5-50 Tuning tools allow you to read parameter values

```

# hostname
MySysWpar
# vmo -a|grep minperm%
          minperm% = 3
# vmo -o minperm%=10
Value of the tunable minperm% cannot be changed in a WPAR

```

```
# ioo -a|grep aio_maxservers
      aio_maxservers = 30
      posix_aio_maxservers = 30
# ioo -o aio_maxservers=15
Value of the tunable aio_maxservers cannot be changed in a WPAR
```

Another useful tool is the **wlmstat -@**. In the global environment, you can check for the occupation of each WPAR in terms of CPU, memory, and total disk I/O (512 byte blocks) as shown in Example 5-51.

Example 5-51 Showing CPU, memory, and disk I/O for the WPARs with wlmstat -@

```
# wlmstat -@ 2 2
      CLASS    CPU    MEM    DKIO
MySysWpar    0.00    0.04    0.00
MyCompetingWpar 0.24    0.04    0.00
      TOTAL    0.24    0.08    0.00

      CLASS    CPU    MEM    DKIO
MySysWpar    0.00    0.04    0.00
MyCompetingWpar 0.44    0.04    0.00
      TOTAL    0.44    0.08    0.00
```

Archived

IBM WPAR Manager for AIX

IBM Workload Partition (WPAR) Manager is used to monitor the WPARs and manage WPAR mobility functions. It will also gather information about CPU, memory, and other useful statistics. This chapter introduces and explains how to use the Workload Partition Manager for AIX. This chapter contains the following sections:

- ▶ 6.1, “Overview of WPAR Manager for AIX” on page 126
- ▶ 6.2, “WPAR monitoring” on page 127
- ▶ 6.3, “WPAR relocation” on page 128
- ▶ 6.4, “WPAR load balancing” on page 131
- ▶ 6.5, “WPAR Manager component overview” on page 135
- ▶ 6.6, “WPAR Manager installation overview” on page 140
- ▶ 6.7, “WPAR Manager installation and configuration” on page 141
- ▶ 6.8, “WPAR Agent installation and configuration” on page 155
- ▶ 6.9, “Creating a System WPAR” on page 162
- ▶ 6.11, “Migrating an application WPAR” on page 192
- ▶ 6.12, “Introduction to mobility command-line interface” on page 230
- ▶ 6.13, “Explanation of mobility CLI commands” on page 231
- ▶ 6.14, “Comparison of CLI commands” on page 232
- ▶ 6.15, “Step-by-step relocation of a workload partition” on page 233

6.1 Overview of WPAR Manager for AIX

WPAR Management for AIX Server is a Java application running in the management server. The WPAR Manager User Interface (UI) provides a browser-driven interface to the WPAR Management Server. The UI displays information that has been collected through the agents and also provides management capabilities, such as the creation, deletion, relocation of WPARs, and so forth. The agent is based on Common Agent Services (CAS) technology. Many of these tasks can also be accomplished from the command-line interface. For more information about this, read 6.12, “Introduction to mobility command-line interface” on page 230.

Workload Partition Manager provides centralized management of system and application WPARs, which is important and valuable to IBM clients. It provides resource optimization. Physical servers can be consolidated and deconsolidated dynamically. For application granularity, this allows for greater utilization of the already powerful Virtualization (APV) capability of AIX and System p. For applications today that require less than 1/10 of a processor to run, the WPAR approach allows their consolidation into a Global LPAR that can distribute the workload with highly granular utilization of the CPU and other system resources. This provides better use of the system and future cost savings to the enterprise.

WPAR Mobility provides another key to success and lowers the cost in the optimization of uptime: Applications can be relocated during maintenance windows and proactive failover occurs in the case of an indication of degradation (predictive failure analysis). This is non-interruptive maintenance, which provides zero downtime for server fixes and upgrades through virtual server/application relocation. You clearly need to test this capability before using it in a production environment. WPAR Mobility is not a replacement for high availability software, such as HACMP or similar products. This is an enhancement to the flexibility and power of the IBM UNIX story as it becomes a stronger and more highly available solution as it evolves with more dynamic allocation and reallocation along with the configuration of virtual servers, storage, and network resources.

Lowering the total cost of ownership is important in every data center. With WPAR technology, the total cost of ownership can help reduce system administration costs through the reduced number of operating system (OS) images, fewer fixes and updates to apply, the ease of creating WPARs, and the ability to delegate administration within system WPARs. To help be successful, a skilled UNIX administrator must be helped with the design, implementation, and support, but this has always been true in the UNIX world. WPARs also provide optimization of performance: applications or virtual servers can be scaled up or down based on the actual throughput demand and performance requirements,

and application text, kernel data, and text are shared, which improves the efficiency of partitioning.

In the next three sections, we describe several of the important features of WPAR Manager: 6.2, “WPAR monitoring” on page 127, 6.3, “WPAR relocation” on page 128, and 6.4, “WPAR load balancing” on page 131.

6.2 WPAR monitoring

During WPAR Agent registration, you have to provide the host name of the CAS Agent Manager. The WPAR Manager then instructs the WPAR Agent to send the information in the format of an XML document to the WPAR Manager at regular intervals. The default is one minute. This is a system-wide value for all servers managed by the WPAR Manager.

The information received by the WPAR Manager is maintained in database tables. With this information, the WPAR Management console allows us to monitor various aspects of the managed WPAR, such as:

- ▶ WPAR status:
 - WPAR name
 - Operational state
 - Type
 - Time of the last modification

- ▶ WPAR performance:

A significant number of performance metrics are sent by the WPAR Agent to the WPAR Manager at a regular interval. This allows the administrator, who uses the WPAR Management console, to monitor the performance of each WPAR and managed system. This also enables the WPAR Manager to know which WPAR might need to be relocated and which Logical Partition or managed system is the most suitable candidate to host the WPAR.

The performance metrics contain:

- WPAR Agent’s GUID
 - WPAR Manager uses GUID to identify from which client the information comes
- The global environment performance metrics
- A list of WPARs
- Performance metrics for each active WPAR

- ▶ Application health inside WPAR:
 - The administrator can also provide scripts to check the health of the application running in WPAR.

6.3 WPAR relocation

An important WPAR Manager feature is its ability to manage relocation of a WPAR by checkpointing the application in one workload partition and restarting it on another Logical Partition or managed system. All of this is done without any application downtime or user interruption.

This relocation can be done for different purposes, such as:

- ▶ To optimize application availability

By moving the WPAR away from the managed system that will be down for preventive maintenance (for example, to upgrade the firmware) or from the managed system that is likely to experience a possible outage in the future (for example, when temporary hardware errors occur more frequently than the specified threshold value), we can ensure that the application will always be available to the users.
- ▶ To optimize application performance

By moving the WPAR to another managed system that can respond to the current workload demand better, we can ensure that the application performance will always be satisfactory to the user regardless of the current workload demand.
- ▶ To optimize resources utilization

By moving the WPAR to another managed system that has idle resources, we can achieve a better overall utilization of all managed systems in the enterprise.

Note that Partition Load Manager (PLM) can only move the idle or underutilized resources from one LPAR to another LPAR within a single managed system. WPAR Manager can move a WPAR to any of the underutilized managed systems (that are compatible with the source system) that are under its control. As a general rule, WPAR Manager can move WPARs between compatible systems that have global interfaces on the same network.

WPAR relocation can occur by using different methods:

► Manual relocation

The administrator, upon reviewing the performance state of the WPAR, selects the source WPAR that needs to be relocated.

Then, the administrator selects the target managed system to which to relocate that WPAR.

Notice that in this approach, the administrator makes the decision of choosing which WPAR to move, when to move it, and where to move it.

The WPAR Management console assists the administrator by providing the following information:

- All WPARs' performance metrics
- All managed systems' performance metrics
- Automatically handle the system compatibility checking between the source and target by presenting only compatible managed systems to be chosen as the target

► Semi-automatic relocation

In this approach, WPAR Manager makes all decisions about choosing the WPAR to move, when to move it, and where to move it to based on the user-defined policy.

However, before it will launch the relocation, it notifies the administrator, asking for an “approval” to proceed. (Hence, it is called semi-automatic.)

This approach was offered so that the administrator can get accustomed to how the policy works before the administrator entrusts the WPAR Manager to perform relocation automatically.

Important: We include a description of the “semi-automatic relocation” option in order to fully describe all of the methods. But, the initial release of WPAR Manager included with IBM AIX Version 6.1 will support only the manual relocation and fully-automatic relocation methods.

► Fully-automatic relocation

This approach is similar to semi-automatic, except that WPAR Manager does not notify the administrator, but proceeds to perform relocation immediately.

The benefit of this method over the semi-automatic method is that the WPAR will be relocated to a managed system that can handle the workload better immediately, and there is no waiting for a confirmation from the administrator, which might not occur in time to handle the surge.

Notice that this semi-automatic/automatic relocation can be used not only to “scale-up” the WPAR to a bigger managed system to handle the workload, but also to “scale-out” the WPARs, which are on the same managed system, to a different managed system, so that they can handle more workload.

6.3.1 Various WPAR states during relocation

WPAR states change during the relocation process. For more details about various WPAR states and the state transition diagram, refer to Chapter 4, “State management” on page 55.

6.3.2 WPAR relocation flow

The following steps are taken during the WPAR relocation:

1. Checkpoint.

Source WPAR is checkpointed.

Note: What happens during the checkpoint of a process?

- ▶ A “virtual” signal is sent to all processes in the WPAR. Because this signal is not known to the process, it cannot be trapped, so all processes will use a special signal handler in the kernel to handle it.
- ▶ This special signal handler swaps out the text segment of the process with the checkpoint code and then runs the checkpoint code, which records all information related to the process to the checkpoint statefile.
- ▶ After the checkpoint code completes recording the information, the signal handler swaps the old text segment back and proceeds to handle the next process.

2. Verify (Pause)

Verify that the checkpoint is completed and the source WPAR is in paused state.

3. Restart

Restart WPAR on the target server using the statefile.

4. Verify (Active)

Verify that the new WPAR is active and running on the target server.

5. Verify (Health)

Verify that the applications in the new WPAR are running and healthy.

Note that this health checking requires the user to provide the scripts to the WPAR Manager.

Note: After this step, the users will be able to continue to use the application. They will have no idea that the application has been moved from one server to another server, except for the feeling that it took a bit longer this time for the application to respond.

6. Kill

Remove the paused WPAR on the source server.

6.4 WPAR load balancing

By automating the relocation of a WPAR to a managed system that is better suited to its current workload, we are effectively load balancing all systems under WPAR Manager control to achieve, not only a better performance for each application, but also a better utilization for the entire enterprise.

In this section, we introduce the concepts of WPAR group and the server group. Then, we discuss the WPAR Manager workload balancer component and WPAR Relocation Policy, the brain of the semi-automatic/automatic relocation.

6.4.1 WPAR group

A *WPAR group* is a collection of WPARs containing similar application instances. A WPAR group shares the same policy and is managed as a unit by the WPAR Manager load balancer component.

For example, a hotel reservation application might consist of three WPAR groups:

- ▶ Web server group, containing all WPARs hosting the Web server
- ▶ Application server group, containing all WPARs hosting the application server
- ▶ Database server group, containing all WPARs hosting the database

All managed WPARs must be associated to one and only one WPAR group.

Note: The performance state of a WPAR group is determined by averaging the performance state of all WPARs in the group.

6.4.2 Server group

A *Server group* is a collection of logical partitions or managed systems to which the WPAR belonging to a WPAR group can be deployed. WPAR relocation is only allowed within a WPAR's server group. It is possible to have multiple server groups associated with a WPAR group.

6.4.3 WPAR Manager workload balancer component

There are three major functions of the WPAR Manager workload balancer:

- ▶ Relocation analysis

This component analyzes the current utilization of all WPARs to find whether there is any requirement to relocate any WPAR. If there are multiple events, it also prioritizes the order of relocation events. It selects the most appropriate managed system as the target for relocation, based on the user-defined policy for each WPAR group.

In brief, this component finds which WPAR to relocate, when to relocate it, and where to relocate it.

- ▶ Relocation workflow management
- ▶ Relocation recovery

6.4.4 WPAR relocation policy

WPAR relocation policy (WRP) determines when and where WPAR is relocated in response to the variance in the demand for the application inside WPAR.

WRP is defined using either the standard performance metrics gathered by the WPAR Agent or custom metrics defined in scripts provided by the user when defining the WPAR group.

Note: In the initial release of WPAR Manager, only the standard performance metrics are supported.

WRP relocation policy can be classified into:

- ▶ WPAR-level policy defines how WPAR Manager calculates the performance state of each WPAR from a collection of performance metrics, each of which has the following attributes:
 - Weight

The weight must be a non-negative value, default is 1.

It specifies the importance of this metric relative to other metrics when calculating the weighted average for a WPAR.

- Limit

In order to have an accurate weighted value from various performance metrics, we need to normalize the observed values so that they all fall between the same range of 0 to 100.

For example, CPU utilization does not need to be normalized, but the disk I/O rate has to be normalized. This is done by dividing the observed value by this limit.

In brief, the limit is the expected maximum possible value that will be used to normalize the observed value so that its range is between 0 to 100.

Notice that no error will be given if the normalized value is more than 100.

- Minimum and maximum threshold

Even though the performance state for a WPAR is normally the weighted average of all performance metrics, WPAR Manager will override this weighted average by the observed value of a metric that violates the minimum or maximum threshold.

This is done to ensure that the not-normal observed value, one that is more than maximum or less than minimum, always gets noticed (and does not disappear with the average calculation).

- Critical

If the metric is critical, and it violates the minimum or maximum threshold value, the load balancer will not continue to calculate the weighted average, but will generate a relocation event immediately.

For example, if the number of disk I/O errors occurrences is more than the maximum threshold, you might want the WPAR Manager to relocate the WPAR immediately to minimize the risk of any potential failure that the hosted system will have on WPAR.

Note: The performance state of a WPAR is calculated by using the weighted average of all performance metrics of that WPAR.

The performance state of a WPAR group is calculated by averaging the performance state of all WPARs in the WPAR group.

- ▶ Group-level policy defines how WPAR Manager interprets the performance state of all WPARs in a WPAR group. It has the following attributes:
 - Minimum

If the performance state of a WPAR is less than the minimum, the WPAR is considered “cold,” which indicates that the resource it has is more than enough for the current demand.

If the performance state of a WPAR group is less than the minimum, WPAR Manager considers this WPAR group “too idle” and a WPAR relocation event will be created.
 - Maximum

If the performance state of a WPAR is greater than the maximum, the WPAR is considered “hot,” which indicates that the resources it has are insufficient for the current demand.

If the performance state of a WPAR group is greater than the maximum, WPAR Manager considers this WPAR group “too busy” and a WPAR relocation event will be created.
 - Period

This is the amount of time in minutes that WPAR Manager used for averaging the performance state of WPARs to obtain the performance state of a WPAR group.

By adjusting the period, you can customize how fast WPAR Manager responds to the surge in the workload.

The default is 5 minutes and it is configurable.

Important: For metrics relating to the utilization, there are two values that WPAR Manager can use:

- ▶ Measured utilization: This is the value that is gotten from the WPAR Agent.
- ▶ Effective utilization: This is the calculated value from the following formula:

Effective utilization = $\text{WPAR utilization} / (100 - (\text{server utilization} - \text{WPAR utilization}))$

For example, assume that WPAR utilization is 20% of the total server. The following example shows how the effective utilization of WPAR depends on the global server load:

If global server utilization is 20%, WPAR effective utilization is 20%.

If global server utilization is 60%, WPAR effective utilization is 33%.

If global server utilization is 100%, WPAR effective utilization is 100%.

Notice that this effective utilization not only considers the WPAR utilization, but also considers the global server utilization, that is, what other things are also happening in the same server as this WPAR.

Thus, if possible, WPAR Manager will use the effective utilization value instead of the observed value, because this gives WPAR Manager a more accurate view of what is happening in the system.

The next section describes the WPAR Manager components and how to install and configure WPAR Manager and WPAR Agent.

6.5 WPAR Manager component overview

Because WPAR Manager makes use of *Common Agent Services* (CAS) in order to have a clear understanding of WPAR Manager components (Figure 6-1 on page 136), we provide a brief overview of the Common Agent Services framework.

6.5.1 Common agent services (CAS) framework

Deploying management software usually requires a server that hosts the management software and an agent that has to be installed on each server that is to be managed, the *managed system*. See Figure 6-1.

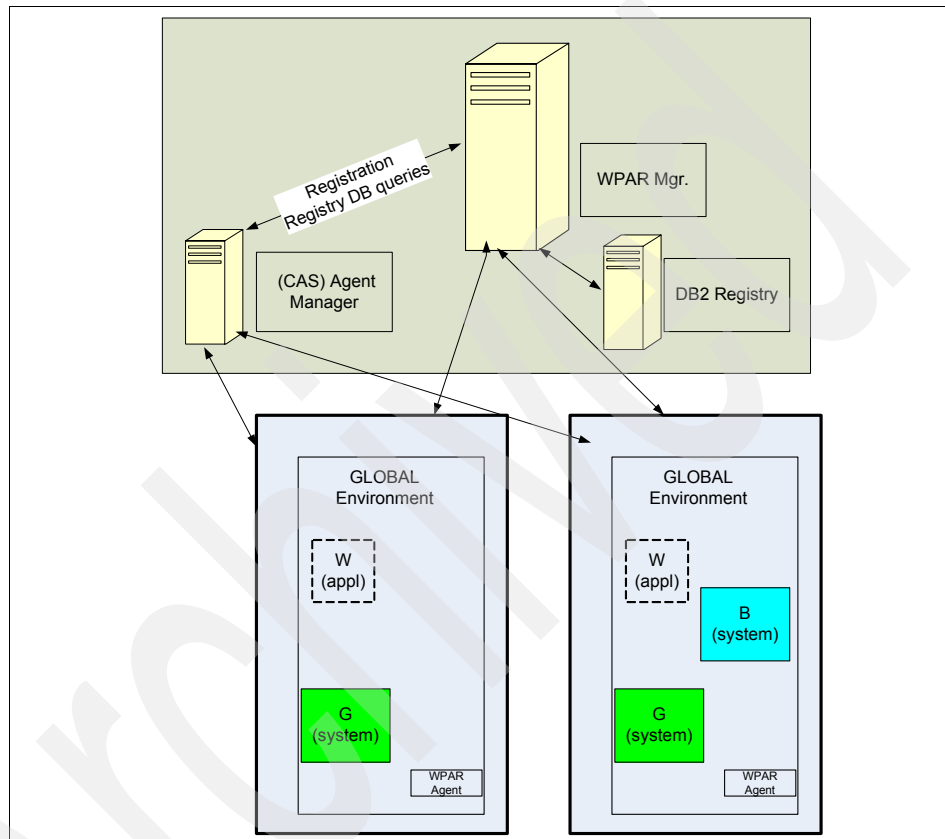


Figure 6-1 Components of WPAR Manager

The effort required to install, configure, maintain, and monitor different agent implementations can become a very time-consuming management task. Besides, the amount of redundant functions provided by each agent implementation (for example, listening to ports, the runtime daemon, and memory consumption) leads to an ineffective usage of the system resources.

The goal of Common Agent Services is to minimize the complexity of the software management deployment by reducing the efforts needed for deployment and utilizing system resources more effectively.

This is accomplished by providing a common framework that can be reused by various management applications that are deployed across the enterprise.

This Common Agent Services framework consists of the following components:

- ▶ The Common Agent
- ▶ The Agent Manager
- ▶ The Resource Manager

6.5.2 Common agent

Each managed system in your deployment runs a *common agent*, which provides a single implementation of common services that can be used by all management products.

The common agent consists of:

- ▶ Common agent services
- ▶ Product-specific subagent (for example, for WPAR Manager, this is the piece of code for WPAR Agent)

For instance, if two management products (A and B) manage the same system, the common agent is installed only once on that system and it will run two product-specific subagents: one for product A and one for product B.

The common agent provides the following functions:

- ▶ A single set of security credentials and a common security infrastructure for all management applications
- ▶ Automated management of security credentials

The common agent certificate is automatically renewed once it is near the expiration date.

- ▶ Common agent health monitoring and configuration monitoring services

The common agent has a “heartbeat” function that sends periodic status and configuration reports to the agent manager. It also allows any subagent to participate and to provide application-specific status.

The frequency of this update can be set or it can be turned off. Management applications can register to the agent manager to receive these updates.

The registry contains only the most recent configuration update. By default, only the most recent status information is saved for each common agent. The retention period is configurable.

Note: WPAR Manager utilizes the information in the registry on the Agent Manager to discover new managed systems.

Note: The common agent functionality for WPAR Manager is in the fileset *wparmgt.cas.agent*.

The product-specific subagent, WPAR Agent, functionality for WPAR Manager is in the fileset *wparmgt.agent.rte*.

WPAR Agent listens to port 9510 and communicates to port 9511, 9512, and 9513 on the CAS Agent Manager.

Notice that these are default ports that can be overridden by the user during the configuration of WPAR Manager.

6.5.3 Agent manager

The *agent manager* is the server component of the common agent services. It provides authentication and authorization services, enables secure connections between managed systems in your deployment, and maintains the registry about the managed systems and the software running on those systems. It also handles queries from the resource managers against the database.

The agent manager has the following components:

- ▶ The agent manager service

The agent manager service serves as a certificate and registration authority to provide authentication and authorization using X.590 digital certificates and the Secure Socket Layer (SSL) protocol. It also handles requests for registry information from common agents and resource managers.

Resource managers and common agents must register with the agent manager service before they can use its services to communicate with each other. This registration is password-protected and there are separate passwords for the common agents and resource managers.

Note: For WPAR Manager, you only need to specify the registration password for the common agents. The password for resource manager is automatically generated during the configuration of WPAR Manager.

- ▶ The registry

The registry is the database that contains the current configuration of all known common agents and resource managers.

Information contained in the registry includes:

- The identity, digital certificates, and communication information for each resource manager
- The identity, digital certificates, and communication information for each common agent
- Basic configuration information for each common agent, for example, hardware type and operating system version
- The status of each common agent
- The last error or, optionally, a configurable number of errors, reported by each common agent

The registry information is updated by events, such as the registration of common agents and resource managers, and by periodic updates from the common agents.

Note: For WPAR Manager, the agent manager uses an Apache Derby database to implement the registry. This Derby database is included within the lightweight runtime fileset that is delivered with WPAR Manager.

► The agent recovery service

The agent recovery service provides error logging for common agents that cannot communicate with other agent manager services.

Note that WPAR Manager does not make use of this feature.

Note: The agent manager functionality for WPAR Manager is in the fileset `wparmgt.cas.agentmgr`.

CAS Agent Manager listens to communication from WPAR Agent and WPAR Manager on ports 9511, 9512, and 9513.

Notice that these are default ports that can be overridden by the user during the configuration of WPAR Manager.

6.5.4 Resource manager

A *resource manager* is management software that uses the services of the agent manager to communicate securely with and to obtain information about the managed systems running the common agent software.

The resource manager uses the services of the common agent to deploy and run its software on the managed systems.

Note that in our case, the WPAR Manager component of WPAR Manager is the resource manager of the CAS framework.

Note: The resource manager functionality for WPAR Manager is in the fileset *wparmgt.mgr.rte*. This fileset also includes the functionality of the WPAR Management console.

WPAR Manager listens to ports 14080 and 14443 and communicates to port 9510 on the WPAR Agent. WPAR Manager listens to ports 9511, 9512, and 9513 on the CAS Agent Manager.

Notice that these are default ports that can be overridden by the user during the configuration of WPAR Manager.

6.6 WPAR Manager installation overview

There are two components of WPAR Manager installation:

- ▶ WPAR Manager

WPAR Manager is the server component. It is installed on a server that will be your management server.

This component also includes the GUI portion of WPAR Management console.

- ▶ WPAR Agent

WPAR Agent is the client component. It is installed on each server that will be managed by the WPAR Manager.

Important: WPAR Manager uses the CAS framework, which has three components:

- ▶ Common agent and product-specific subagent: This is the CAS Agent and WPAR Agent component.
- ▶ Agent manager: This is CAS Agent Manager.
- ▶ Resource manager: This is the WPAR Manager component.

Each component can be deployed on a different server. However, to simplify the installation, by default, WPAR Manager and CAS Agent Manager are installed on the same system, which will be the management server. CAS Agent and WPAR Agent are installed on any server that will be managed.

6.7 WPAR Manager installation and configuration

In addition to CAS Agent Manager, you are required to install DB2 and configure it for use by WPAR Manager as a data repository. This DB2 software is included with the distribution of WPAR Manager.

In this scenario, we start with a completely new installation with no existing CAS Agent Manager or any DB2 server in our environment.

We will install WPAR Manager, CAS Agent Manager, and DB2 on the machine called `stonehollow`. We will install WPAR Agent on four machines: `braker`, `parmer`, `metric`, and `gracyfarms`.

The steps required are:

- ▶ Check prerequisites and prepare for installation for installation.
- ▶ Install WPAR Manager and CAS Agent Manager.
- ▶ Install and configure DB2.
- ▶ Configure WPAR Manager and CAS Agent Manager.
- ▶ Verify WPAR Manager installation.

6.7.1 Check prerequisites and prepare for installation

Use this checklist for the software prerequisites. Perform the following actions to prepare your system before we install WPAR Manager:

- ❑ IBM AIX Version 6.1:
 - ▶ Use `oslevel -r` to verify.
- ❑ Java Version 5:
 - ▶ Use `ls1pp -lq 'Java5*'` to verify.
- ❑ DB2 Version 9 preparation:
 - ▶ Free space in /tmp - 125 MB
This free space is used for temporary files during DB2 installation.
 - ▶ Free space in /opt - 500 MB
This free space is used for the /opt/IBM/WPAR/manager/db2 directory, which stores DB2 program code.
 - ▶ Free space in /home - 800 MB
This free space is used for the /home/db2wgmt directory, which is the database instance owner home directory. It also contains database *WPARMGTDDB*, which is used by WPAR Manager.

Important: Make sure that there is plenty of free space left in these file systems.

6.7.2 Install WPAR Manager and CAS Agent Manager

The following steps show how to install WPAR Manager and CAS Agent Manager:

1. Log in as root to the partition that you plan to use as the WPAR Manager server.
2. Mount the WPAR Manager CD-ROM:

```
mount -o ro /dev/cd0 /cdrom
```
3. Install the WPAR Manager fileset, *wparmgt.mgr*, using SMIT.

Example 6-1 on page 143 shows the content of the SMIT log file.

Example 6-1 Example output from the WPAR Manager Installation

```
geninstall -I "a -cgNQqwXY -J" -Z -d . -f File 2>&1

File:
  I:tivoli.tivguid           1.3.0.0
  I:wparmgt.mgr.rte         1.1.1.0

-----+-----
                          Pre-installation Verification...
-----+-----
Verifying selections...done
Verifying requisites...done
Results...

SUCSESSES
-----
Filesets listed in this section passed pre-installation verification
and will be installed.

Selected Filesets
-----
tivoli.tivguid 1.3.0.0           # IBM Tivoli GUID on AIX
wparmgt.mgr.rte 1.1.1.0         # Workload Partitions Manager

Requisites
-----
(being installed automatically; required by filesets listed above)
wparmgt.cas.agentmgr 1.3.2.18   # Common Agent Services Agent ...

<< End of Success Section >>

-----+-----
                          BUILDDATE Verification ...
-----+-----
Verifying build dates...done
FILESET STATISTICS
-----
  2 Selected to be installed, of which:
    2 Passed pre-installation verification
  1 Additional requisites to be automatically installed
-----
  3 Total to be installed

-----+-----
                          Installing Software...
-----+-----
```

```
installp: APPLYING software for:
        wparmtg.mgr.rte 1.1.1.0
```

```
. . . . . << Copyright notice for wparmtg.mgr >> . . . . .
Licensed Materials - Property of IBM
```

```
5765WPM00
    Copyright International Business Machines Corp. 2007.
```

```
All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. . . . . << End of copyright notice for wparmtg.mgr >>. . . . .
```

```
Filesets processed: 1 of 3 (Total time: 31 secs).
```

```
installp: APPLYING software for:
        tivoli.tivguid 1.3.0.0
```

```
. . . . . << Copyright notice for tivoli.tivguid >> . . . . .
Licensed Materials - Property of IBM
```

```
5698GUID
IBM Tivoli GUID
(C) Copyright International Business Machines Corp. 2003.
```

```
All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. . . . . << End of copyright notice for tivoli.tivguid >>. . . . .
```

```
-----
Performing Configuration tasks for GUID.
```

```
-----
TIVGUID file exists, checking for /usr/tivoli/guid/tivguid
```

```
/usr/tivoli/guid/tivguid exists..., execute tivguid -show
Tivoli GUID utility - Version 1, Release 3, Level 0.
(C) Copyright IBM Corporation 2002, 2004 All Rights Reserved.
```

```
Guid:7f.59.af.5e.87.6d.11.dc.98.85.08.63.09.03.94.2a
```

```
-----
Completed Configuration tasks for GUID.
-----
```

```
Filesets processed: 2 of 3 (Total time: 33 secs).
```

```
installp: APPLYING software for:
          wparmgt.cas.agentmgr 1.3.2.18
```

```
. . . . . << Copyright notice for wparmgt.cas.agentmgr >> . . . . .
Licensed Materials - Property of IBM
```

```
5765WPM00
  Copyright International Business Machines Corp. 2007.
```

```
All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. . . . . << End of copyright notice for wparmgt.cas.agentmgr >>. . . . .
```

```
Finished processing all filesets. (Total time: 43 secs).
```

```
+-----+
                          Summaries:
+-----+
```

Installation Summary

Name	Level	Part	Event	Result
wparmgt.mgr.rte	1.1.1.0	USR	APPLY	SUCCESS
wparmgt.mgr.rte	1.1.1.0	ROOT	APPLY	SUCCESS
tivoli.tivguid	1.3.0.0	USR	APPLY	SUCCESS
tivoli.tivguid	1.3.0.0	ROOT	APPLY	SUCCESS
wparmgt.cas.agentmgr	1.3.2.18	USR	APPLY	SUCCESS
wparmgt.cas.agentmgr	1.3.2.18	ROOT	APPLY	SUCCESS

Notice that, in addition to wparmgt.mgr, three other filesets are also installed:

a. *lwi.runtime* (Prerequisite)

Lightweight Infrastructure (LWI) Run time provides a standard-based, low-overhead approach to run the application server. It is a framework for service-oriented architecture (SOA) that is implemented using Eclipse technology.

WPAR Manager, CAS Agent Manager, and the WPAR Agent all run in the LWI run time.

b. *tivoli.tivguid* (Corequisite)

GUID is a 32-hexadecimal digit number that is used to uniquely identify a common agent or a resource manager in the environment.

Example 6-2 shows the command to display the GUID.

Example 6-2 Command to show the GUID

```
stonehollow:/ ) /usr/tivoli/guid/tivguid -show
Tivoli GUID utility - Version 1, Release 3, Level 0.
(C) Copyright IBM Corporation 2002, 2004 All Rights Reserved.
```

```
Guid:8a.47.02.06.fe.90.11.db.aa.ec.08.63.09.03.05.94
```

c. *wparmgt.cas.agentmgr* (Corequisite)

This fileset contains the CAS Agent Manager function.

6.7.3 Install and configure DB2

DB2 databases are managed by a user who is usually called an *instance owner*. You can think of the instance owner as similar to a root user in AIX in that the instance owner can do all system administration tasks. The user and group for the instance owner are created automatically by *DBInstall.sh*, but you have to specify its password.

Install WPAR Manager database support

Use the following process to install WPAR Manager database support:

1. Log in as root to the partition that you plan to use as WPAR Manager server.
2. Mount WPAR Manager CDROM:

```
mount -o ro /dev/cd0 /cdrom
```
3. Install WPAR Manager fileset, *wparmgt.db*, using SMIT.

Example 6-4 on page 148 shows the SMIT log output from the WPAR Manager file set installation.

Example 6-3 Example output from the DB2 installation

```
geninstall -I "a -cgNQqwXY -J" -Z -d . -f File 2>&1
```

```
File:
  I:wparmgt.db.db2          1.1.1.0
```

```
+-----+
                        Pre-installation Verification...
+-----+
Verifying selections...done
Verifying requisites...done
```

Results...

SUCSESSES

Filesets listed in this section passed pre-installation verification and will be installed.

Selected Filesets

wparmgt.db.db2 1.1.1.0 # Workload Partitions Manager ...

<< End of Success Section >>

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

BUILDDATE Verification ...

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

Verifying build dates...done

FILESET STATISTICS

- 1 Selected to be installed, of which:
 - 1 Passed pre-installation verification

- 1 Total to be installed

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

Installing Software...

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

installp: APPLYING software for:
wparmgt.db.db2 1.1.1.0

. << Copyright notice for wparmgt.db >>
Licensed Materials - Property of IBM

5765WPM00
Copyright International Business Machines Corp. 2007.

All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. << End of copyright notice for wparmgt.db >>.

Finished processing all filesets. (Total time: 1 secs).

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

Summaries:

+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+

Installation Summary

Name	Level	Part	Event	Result
wparmgt.db.db2	1.1.1.0	USR	APPLY	SUCCESS
wparmgt.db.db2	1.1.1.0	ROOT	APPLY	SUCCESS

Install DB2

Now use the DBInstall.sh script to install and configure DB2 for WPAR Manager:

```
/opt/IBM/WPAR/manager/db/bin/DBInstall.sh -dbinstallerdir /cdrom/db2  
-dbpassword <instanceOwnerPassword>
```

Example 6-4 shows the output from running the DBInstall.sh script.

Example 6-4 Example output from the DB2 installation

```
DBInstall.sh:Database install started.  
DBInstall.sh:Database install successful.  
DBInstall.sh:Database instance creation started.  
DBInstall.sh:Database instance creation successful.  
DBInstall.sh:Database creation started.  
DBInstall.sh:Database population started.  
DBInstall.sh:Database creation successful.
```

The DBInstall.sh script performs these tasks:

- ▶ Verifies that port 50000, which will be used for DB2, is not already in use
- ▶ Verifies that there is enough space in /tmp, /opt/IBM/WPAR/manager/db2, and /home/db2wmgt (instance owner home), which is used to store db2 code
- ▶ Runs db2setup to install DB2
- ▶ Runs db2isetup to create db2 instance *db2wmgt* and database *WPARMGTDDB*
- ▶ Creates tables, indexes, views, and triggers and populate tables that WPAR Manager will use
- ▶ Sets the database to automatically start when the system starts

You can also view the detailed information in the log file at /var/opt/IBM/WPAR/manager/logs/install/WPMDBI.log.

6.7.4 Configure WPAR Manager and CAS Agent Manager

WPAR Manager Configurator (**WPMConfig.sh** command) is used to configure WPAR Manager and CAS Agent Manager.

There are three modes in which WPAR Manager Configurator can be used:

- ▶ Graphical mode (GUI - An X11 graphical environment must exist)
- ▶ Console mode (text)
- ▶ Quiet mode (use a response file)

In our example, we use WPAR Manager Configurator in console mode:

```
/opt/IBM/WPAR/manager/bin/WPMConfig.sh
```

Example 6-5 shows the output from running the `WPMConfig.sh` command in console mode.

Example 6-5 Example output from the WPMConfig.sh command

```
Preparing CONSOLE Mode Installation...
```

```
=====
Choose Locale...
-----
```

- 1- Català
- 2- Deutsch
- >3- English
- 4- Español
- 5- Français
- 6- Italiano
- 7- Português (Brasil)

```
CHOOSE LOCALE BY NUMBER:
=====
=====
```

```
IBM Workload Partitions Manager for AIX(created with InstallAnywhere by
Macrovis
ion)
-----
----
```

```
===Uninstall DB2===
```

```
In case you need to uninstall DB2:
```

```
# Login as root to the partition that is WPAR Manager server where DB2 is
running and enter.
```

```
# /opt/IBM/WPAR/manager/db/bin/DBUninstall.sh -dbuser db2wmgmt
```

```
=====
Introduction
-----
```

Welcome to the IBM Workload Partitions Manager for AIX setup. This wizard will configure access to a DB2 database server and will configure WPAR Manager to use an Agent Manager for secure communication with WPAR Agents.

To complete this configuration you will need the following:

-A running and available instance of DB2 V9.1

-All information needed to access the DB2 V9.1 database, including: hostname, username, password, service port and database name. If you do not have this information you should consult your DB2 Administrator before continuing with the configuration.

-(Advanced setup only) All information needed to access an existing Agent Manager, including: hostname, public port, resource manager username and password.

For additional information about this setup, please consult the WPAR Installation Documentation.

PRESS <ENTER> TO CONTINUE:

=====

WPAR Manager Access Information

WPAR Manager is accessed from a browser with the URL:
http://<hostname>:<public port>/ibm/console.

The secure port is used for all browser and server communication. The port numbers should only be changed if there is an existing conflict or anticipated port conflict..

Secure Port: (DEFAULT: 14443):
Public Port: (DEFAULT: 14080):

=====

Database Information

Complete the following information to configure WPAR Manager's access to a DB2 V9.1 database residing on either a local or remote server. If you do not have the following information, cancel out of configuration and consult with your system administrator.

Hostname: (DEFAULT: stonehollow):
Username: (DEFAULT: db2wmgmt):
Password: (DEFAULT:):
Service Port: (DEFAULT: 50000):
Database Name: (DEFAULT: WPARMGT):

=====
Agent Manager Information

Agent Manager is required by WPAR Manager for secure communication between Agent

s. WPAR Manager will register with an Agent Manager to retrieve secure certificates used for authenticating with WPAR Agents. You have two options for setting up the Agent Manager.

1) Configure a new Agent Manager on this server. This is recommended for a typical new installation of WPAR Manager.

2) Register to an existing Agent Manager. This is an advanced configuration and is only needed in environments where WPAR Manager may have to coexist with an Agent Manager provided by other IBM products. If you select this option, you will need to provide the hostname, public port, resource manager username, and password to be able to connect to the existing Agent Manager.

Select whether you want to configure a new Agent Manager on this server, or register to an existing Agent Manager.

- >1- Configure Agent Manager (typical)
- 2- Register to an Existing Agent Manager (advanced)

ENTER THE NUMBER FOR YOUR CHOICE, OR PRESS <ENTER> TO ACCEPT THE DEFAULT:

=====
Agent Manager Configuration

The following password will be used by WPAR Agents to register with the Agent Manager.

Create Registration Password: (DEFAULT:):
Verify Password: (DEFAULT:):

=====
Agent Manager Configuration

The following ports are used for communication between Agent Manager, Agents on managed systems, and WPAR Manager. The port numbers should only be changed if there is an existing or anticipated port conflict on your system.

Base Port: (DEFAULT: 9511):
Secure Port: (DEFAULT: 9512):
Public Port: (DEFAULT: 9513):

=====
WPAR Manager Configuration Summary

Click Next to configure WPAR Manager with the following values.
Click Cancel to terminate.

WPAR Manager Access:
Public Port: 14080
Secure Port: 14443
Database Access:
Hostname: stonehollow
Username: db2wmgmt
Password: *****
Service Port: 50000
Name: WPARMGT
Agent Manager, configure local:
Base Port: 9511
Public Port: 9513
Secure Port: 9512
Registration password: *****

->1- Next
2- Cancel

ENTER THE NUMBER OF THE DESIRED CHOICE, OR PRESS <ENTER> TO ACCEPT THE
DEFAULT:

=====
WPAR Configuration Complete

The configuration of IBM Workload Partitions Manager for AIX has completed successfully.

PRESS <ENTER> TO EXIT THE INSTALLER:

The following summarizes the actions done by the **WPMConfig.sh** command:

- ▶ If there is no existing CAS Agent Manager in the environment, create one, configure it, and assign an agent registration password.
This password is needed when you configure the WPAR Agent at the client (the server that will be managed by WPAR Manager).
- ▶ Set up the LWI runtime property file for WPAR Manager, `wpmcfg.properties`, with the appropriate ports, instance owner name and password, database name, database host name, user name, and password for WPAR Manager to register with CAS Agent Manager.
- ▶ Start CAS Agent Manager and WPAR Manager.
- ▶ Register WPAR Manager to CAS Agent Manager.
- ▶ Set WPAR Manager to autostart at reboot (`/etc/inittab` file).
- ▶ Set CAS Agent Manager to autostart at reboot (`/etc/inittab` file).

You can view the log file at `/opt/IBM/WPAR/manager/logs/install/WPMConfig.log`.

6.7.5 Verify WPAR Manager installation

To check that the WPAR Manager installation is completed:

- ▶ Verify that WPAR Manager daemon is active.
- ▶ Verify that CAS Agent Manager daemon is active.
- ▶ It is possible for a Web browser to connect to both of them.

For the WPAR Manager daemon:

- ▶ To verify, use `/opt/IBM/WPAR/manager/bin/wparmgr status`.
- ▶ To start, use `/opt/IBM/WPAR/manager/bin/wparmgr start`.
- ▶ To stop, use `/opt/IBM/WPAR/manager/bin/wparmgr stop`.

For the CAS Agent Manager daemon:

- ▶ To verify, use `/opt/IBM/WPAR/manager/bin/agentmgr status`.
- ▶ To start, use `/opt/IBM/WPAR/manager/bin/agentmgr start`.

- ▶ To stop, use `/opt/IBM/WPAR/manager/bin/agentmgr stop`.

Note: Both WPAR Manager and CAS Agent Manager are configured to autostart from inittab, so normally you do not need to start it after rebooting the system.

We can also use the Web browser to verify the installation by testing that it can connect to both CAS Agent Manager and WPAR Manager.

Note: To use your browser for the WPAR Management console:

- ▶ Use Firefox 1.5 or later and Internet Explorer® (IE) 6 or later.
- ▶ JavaScript™ must be enabled in the browser.
- ▶ Because IE does not have native support for Scalable Vector Graphic (SVG), the Adobe® SVG plug-in is needed, which you can download from:
<http://www.adobe.com/svg/viewer/install/main.html>

To verify the connection to the CAS Agent Manager:

`http://<WPAR Manager hostname>:9513/AgentMgr/Info`

This URL verifies that the CAS Agent Manager is running. If it is not running, you will get a message similar to:

`“Cannot establish a connection to the server at stonehollow:9513.”`

To verify the connection to the WPAR Manager:

`http://<WPAR Manager hostname>:14080/ibm/console`

This URL verifies that the WPAR Manager is running. If it is not running, you will get a message similar to:

`“Cannot establish a connection to the server at stonehollow:14443.”`

6.7.6 Uninstall WPAR Manager and CAS Agent Manager

In case you need to uninstall WPAR Manager and CAS Agent Manager:

1. Log in as root to the partition that is the WPAR Manager server.
2. Use SMIT to remove the following filesets:
 - wparmgt.mgr
 - wparmgt.cas.agentmgr
 - tivoli.tivguid

– lwi.runtime

6.7.7 Uninstall DB2

In case you need to uninstall DB2:

1. Log in as root to the partition that is the WPAR Manager server.
2. Mount WPAR Manager CD-ROM:

```
mount -o ro /dev/cd0 /cdrom
/cdrom/db2/DBUninstall.sh db2wmgmt
```

6.8 WPAR Agent installation and configuration

WPAR Agent is installed on each Logical Partition or managed system that will be managed by the WPAR Manager. *WPAR Mobility feature*, the ability to checkpoint an application and restart it in a different Logical Partition or managed system, is included in the WPAR Agent.

The required steps are:

1. Check prerequisites.
2. Install WPAR Agent.
3. Configure WPAR Agent.
4. Verify WPAR Agent installation.

6.8.1 Check prerequisites

Use this checklist for software prerequisite checking for WPAR Agent:

- bos.wpars 6.1.0.0
- Java5.sdk 1.5.0.0
- perfagent.tools 6.1.0.0

These three filesets are generally installed by default.

6.8.2 Install WPAR Agent

Install WPAR Agent:

1. Log in as root to the partition that will be managed by WPAR Manager.

2. Mount the WPAR Manager CD-ROM:

```
mount -o ro /dev/cd0 /cdrom
```

3. Install the WPAR Agent fileset, *wparmgmt.agent*, using SMIT. See Example 6-6.

Example 6-6 Example output from the WPAR Agent installation

```
geninstall -I "agQqwXY -J" -Z -d . -f File 2>&1
```

File:

```
I:wparmgmt.agent.rte          1.1.0.0
```

```
+-----+  
          Pre-installation Verification...  
+-----+
```

```
Verifying selections...done
```

```
Verifying requisites...done
```

```
Results...
```

```
SUCCESSSES
```

```
-----
```

```
Filesets listed in this section passed pre-installation verification  
and will be installed.
```

```
Selected Filesets
```

```
-----
```

```
wparmgmt.agent.rte 1.1.0.0          # Workload Partitions Manager ...
```

```
Requisites
```

```
-----
```

```
(being installed automatically; required by filesets listed above)
```

```
mcr.rte 4.0.25.0                    # Metacluster Checkpoint and R...
```

```
tivoli.tivguid 1.3.0.0              # IBM Tivoli GUID on AIX
```

```
wparmgmt.cas.agent 1.3.2.14        # Common Agent Services Agent
```

```
<< End of Success Section >>
```

```
+-----+  
          BUILDDATE Verification ...  
+-----+
```

```
Verifying build dates...done
```

```
FILESET STATISTICS
```

```
-----
```

```
1 Selected to be installed, of which:
```

```
1 Passed pre-installation verification
```

```

    3 Additional requisites to be automatically installed
    ----
    4 Total to be installed

+-----+
|           Installing Software...           |
+-----+

installp: APPLYING software for:
    wparmtg.cas.agent 1.3.2.14

. . . . . << Copyright notice for wparmtg.cas.agent >> . . . . .
Licensed Materials - Property of IBM

5765WPM00
  © Copyright International Business Machines Corp. 2007.

All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. . . . . << End of copyright notice for wparmtg.cas.agent >>. . . . .

Filesets processed:  1 of 4  (Total time:  20 secs).

installp: APPLYING software for:
    wparmtg.agent.rte 1.1.0.0

. . . . . << Copyright notice for wparmtg.agent >> . . . . .
Licensed Materials - Property of IBM

5765WPM00
  © Copyright International Business Machines Corp. 2007.

All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. . . . . << End of copyright notice for wparmtg.agent >>. . . . .

DR script file /opt/IBM/WPAR/agent/scripts/dr_wparmtg installed successfully

ALR0064I Successfully linked /opt/IBM/WPAR/agent to
/var/opt/IBM/WPAR/cas/agent/runtime/base/eclipse/links/EXTERNAL.1.link.

```

SUCCESS

Filesets processed: 2 of 4 (Total time: 25 secs).

installp: APPLYING software for:
mcr.rte 4.0.25.0

Filesets processed: 3 of 4 (Total time: 29 secs).

installp: APPLYING software for:
tivoli.tivguid 1.3.0.0

. << Copyright notice for tivoli.tivguid >>
Licensed Materials - Property of IBM

5698GUID

IBM Tivoli GUID

(C) Copyright International Business Machines Corp. 2003.

All rights reserved.

US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.

. << End of copyright notice for tivoli.tivguid >>.

Performing Configuration tasks for GUID.

TIVGUID file exists, checking for /usr/tivoli/guid/tivguid

/usr/tivoli/guid/tivguid exists..., execute tivguid -show
Tivoli GUID utility - Version 1, Release 3, Level 0.
(C) Copyright IBM Corporation 2002, 2004 All Rights Reserved.

Guid:35.df.20.82.fe.67.11.db.95.b3.08.63.09.03.05.90

Completed Configuration tasks for GUID.

Finished processing all filesets. (Total time: 32 secs).

+-----+
Summaries:
+-----+

Installation Summary

Name	Level	Part	Event	Result
wparmgt.cas.agent	1.3.2.14	USR	APPLY	SUCCESS
wparmgt.cas.agent	1.3.2.14	ROOT	APPLY	SUCCESS
wparmgt.agent.rte	1.1.0.0	USR	APPLY	SUCCESS
wparmgt.agent.rte	1.1.0.0	ROOT	APPLY	SUCCESS
mcr.rte	4.0.25.0	USR	APPLY	SUCCESS
tivoli.tivguid	1.3.0.0	USR	APPLY	SUCCESS
tivoli.tivguid	1.3.0.0	ROOT	APPLY	SUCCESS

File /etc/inittab has been modified.

One or more of the files listed in /etc/check_config.files have changed.

See /var/adm/ras/config.diff for details.

Notice that, in addition to wparmgt.agent, three other filesets are also installed:

a. *wparmgt.cas.agent* (Prerequisite)

This fileset contains the CAS Agent function.

b. *tivoli.tivguid* (Corequisite for wparmgt.cas.agent)

GUID is a 32-hexadecimal digit number that is used to uniquely identify a common agent or a resource manager in the environment.

Example 6-7 shows how to display the GUID.

Example 6-7 Note that this GUID is different from the GUID in Example 6-2 on page 146

```
parmer:/ ) /usr/tivoli/guid/tivguid -show
Tivoli GUID utility - Version 1, Release 3, Level 0.
(C) Copyright IBM Corporation 2002, 2004 All Rights Reserved.
```

```
Guid:35.df.20.82.fe.67.11.db.95.b3.08.63.09.03.05.90
```

c. *mcr.rte* (Corequisite)

Metachuster Checkpoint and Restart (MCR) is the software that provides the capability to *checkpoint* (capture the entire state of running applications to be relocated) a workload partition to a statefile and restart the applications, using the information in that statefile, on another logical partition or managed system.

All of this checkpointing and restarting is done without any application downtime or any interruption to the users.

6.8.3 Configure WPAR Agent

The WPAR Agent must register itself to the CAS Agent Manager to allow the managed system to participate in the CAS environment and consequently to be managed by WPAR Manager.

Important: The date and time on the managed system must be within 24 hours of the date and time on the CAS Agent Manager for successful registration. The values are compared in coordinated universal time (UTC) so the systems can be in different time zones.

To configure WPAR Agent, log in as root to the managed system:

```
/opt/IBM/WPAR/agent/bin/configure-agent -hostname <CasAgentMgrHostname>
```

Example 6-8 shows running the **configure-agent** command.

Example 6-8 Example output from the configure-agent command

```
gracyfarms:/ ) /opt/IBM/WPAR/agent/bin/configure-agent -hostname
stonehollow
Agent Registration Password:
Re-enter Agent Registration Password:
0513-059 The wparagent Subsystem has been started. Subsystem PID is
409728.
```

You will be asked to enter the *Agent Registration password*. This is the password that you provide during the **WPMConfig** step. (If you use existing CAS Agent Manager, you will need this agent registration password from the CAS administrator.)

You can view the error messages for registration in the log file at `/opt/IBM/WPAR/agent/cas/logs/rcp.log.0`.

Note: On successful registration, the following files are created on the managed system in the `/opt/IBM/WPAR/agent/cas/runtime/agent/cert` directory:

- ▶ CertificateRevocationList
- ▶ agentKeys.jks
- ▶ agentTrust.jks
- ▶ pwd

6.8.4 Verify WPAR Agent installation

To check that WPAR Agent installation is completed:

- ▶ Verify that WPAR Agent daemon is active.
- ▶ Verify that the WPAR Management console can discover the server on which the WPAR Agent has been installed.

For WPAR Agent daemon:

- ▶ To verify, use `/opt/IBM/WPAR/agent/bin/wparagent status`.
- ▶ To start, use `/opt/IBM/WPAR/agent/bin/wparagent start`.
- ▶ To stop, use `/opt/IBM/WPAR/agent/bin/wparagent stop`.

Note: WPAR Agent is configured to autostart from inittab, so normally you do not need to start it after rebooting the system.

After registration, from the WPAR Management console, you can start to *discover* the managed system (that its agent has been installed and configured) and start to manage them from the WPAR Management console.

To discover the newly registered managed system:

- ▶ Point your browser to <http://stonehollow:14080/ibm/console>
- ▶ Enter user name and password

After the initial install, there is only one user for the WPAR Management console, which is the root on the WPAR Manager machine. In this case, it is root on stonehollow.

Tip: You can add other users to the WPAR Management console via **Settings** → **Console User Authority** → **Add**.

Because WPAR Manager makes use of AIX authentication, this user must exist as an AIX user on the WPAR Manager machine.

- ▶ Select **Resource Views** → **Managed Systems** → **Discover**

Figure 6-2 on page 162 shows the result after Discover was run. Notice that all four clients that we installed the WPAR Agent are now visible to the WPAR Manager. Therefore, it is possible for the WPAR Manager to manage the WPAR on these systems from now on.

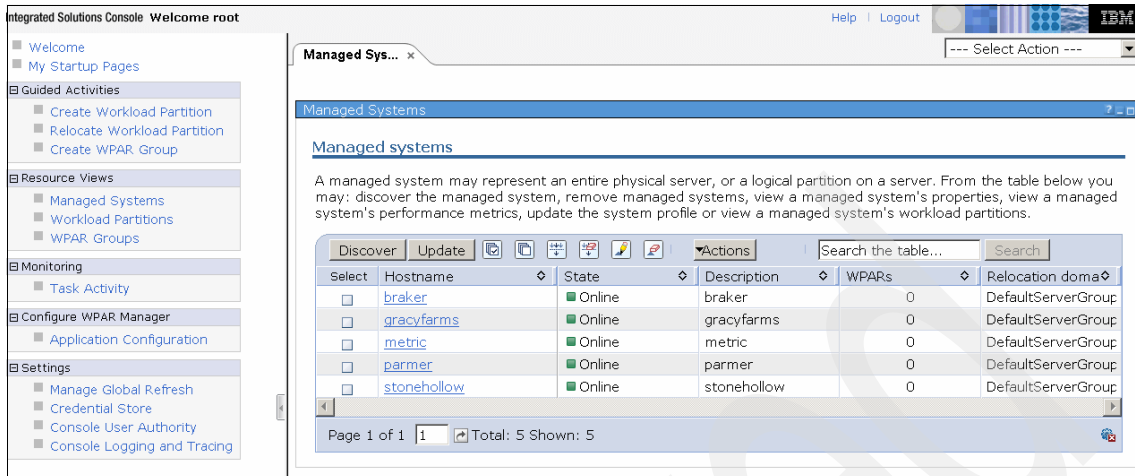


Figure 6-2 Shows the result after running Discover

6.8.5 Uninstall WPAR Agent

In case you need to uninstall the WPAR Agent:

1. Log in as root to the partition.
2. Use SMIT to remove the following filesets:
 - wparmgmt.agent
 - wparmgmt.cas.agent
 - tivoli.tivguid
 - mcr.rte

From now on, we discuss how to use the WPAR Management console to manage WPAR.

6.9 Creating a System WPAR

At this point, the topology has been set up to use WPAR Mobility. There are two System p servers with Global Environments on them. One server is called sa1tgrass and the other epp182. They represent AIX V6.x and happen to be LPARs on POWER5 systems utilizing virtual input/output (VIO) capabilities of network and disk. There is an AIX V6.x LPAR on a third system called darkmatter. This system is a very small LPAR also using VIO to serve the disk and network. It is the AIX V6.x WPAR Manager for AIX Mobility and has the WPAR Agent Manager installed on it. There are two other servers: one server is

an AIX V6.x that is set up as a NIM master, which was used to load the three servers previously mentioned, and the last server is an AIX V6.x Network File System (NFS) server where the data lives for an HTTP WebSphere Application Server. This will be the main target of mobility throughout the chapter.

Now, we will walk you through a “cookbook” style representation of what a user expects to see. The only caveat is that we used a pre-general availability (GA) version of the code with many months yet to go before GA, so several windows might have changed by the time the final product is ready for general distribution. We advise you to read the readme file and any documentation that ships with the product.

Figure 6-3 on page 164 is the first window you see when logging in to the User Interface (UI). This window is similar to many other Web interface tools on the market today.

The steps are:

1. Pick your browser of choice and enter:

```
{https://(ip address or name of server with full  
path):14443/ibm/console}
```

This opens the UI to the user login page. Enter the user ID and password and click **Log in**.

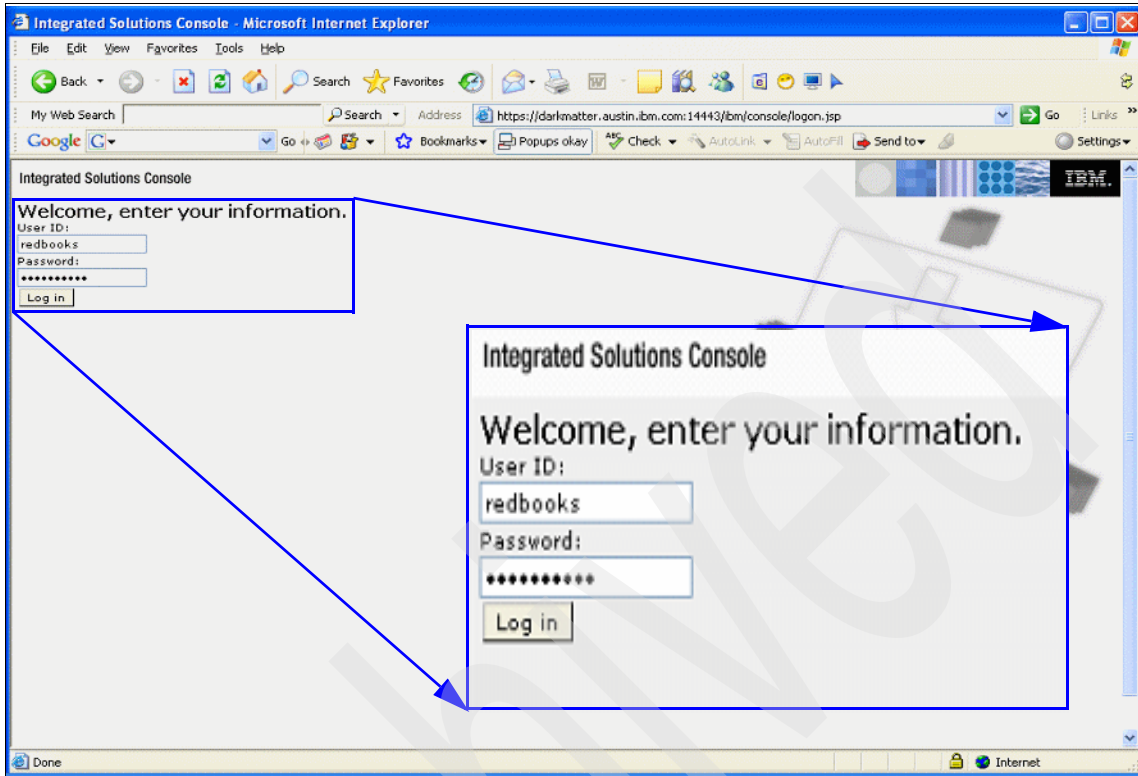


Figure 6-3 http login window for User Interface of WPAR Manager for AIX

Figure 6-4 takes you to the main welcome page in the UI. This page provides information about what a Workload Partition is and how to get started.

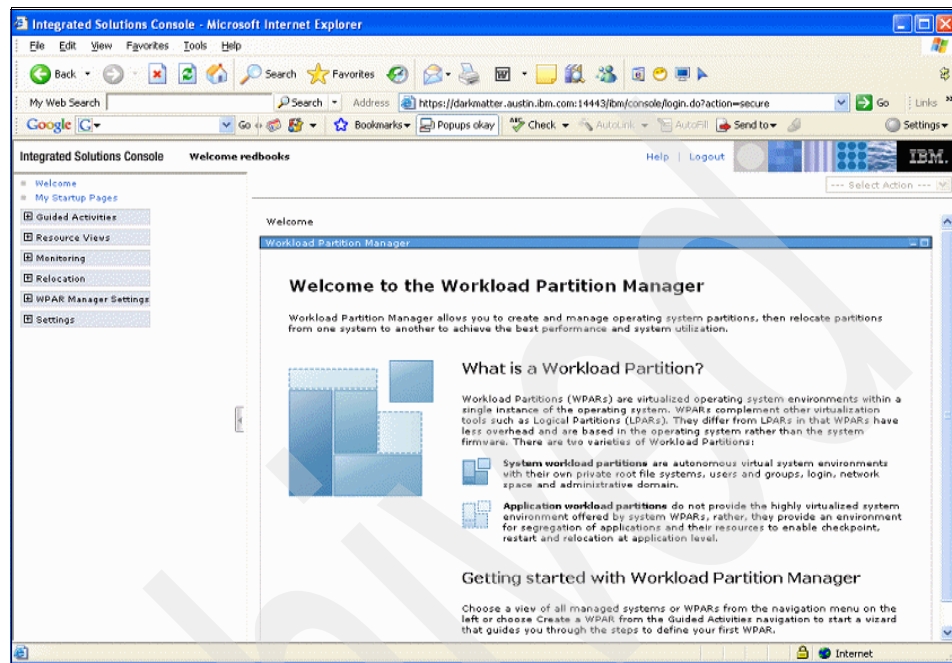


Figure 6-4 Welcome window for the Workload Partition Manager

2. The graphic interface is an intuitive wizard that drives you to the next logical step after you make a selection from a menu. First, you create and manage WPARs. Click **Guided Activities** in the upper left corner of the page. This is shown in Figure 6-5 on page 166.

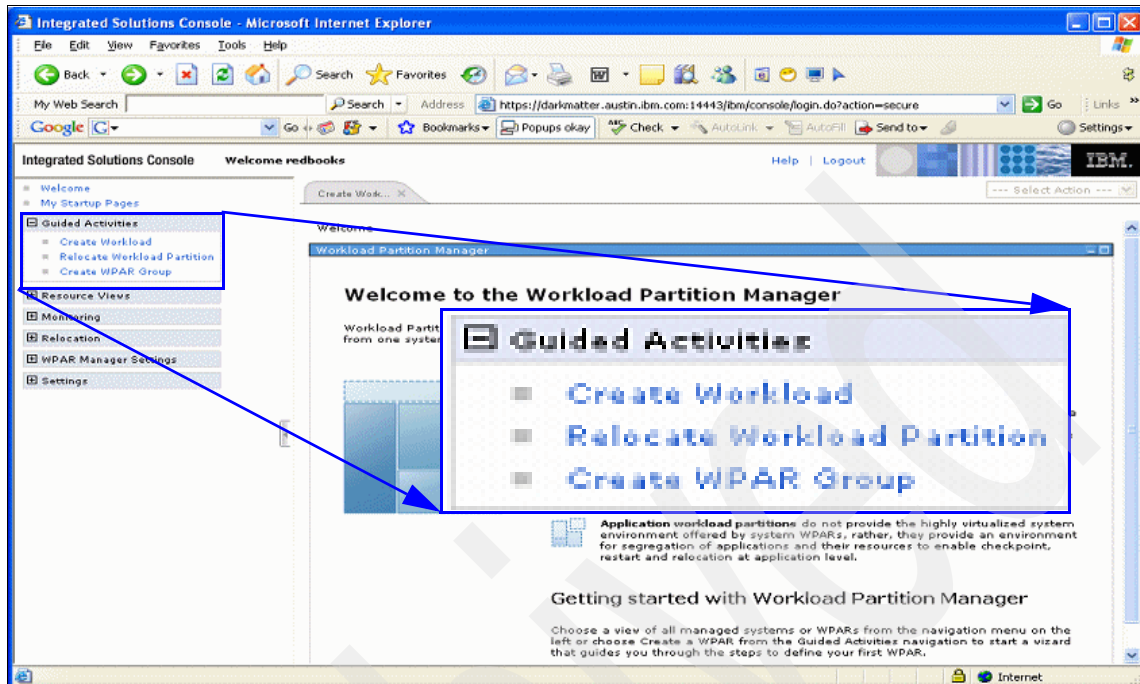


Figure 6-5 Guided Activities, Create Workload

Guided Activities opens a menu with three choices:

- Create Workload
 - Relocate Workload Partition
 - Create WPAR Group
3. Click **Create Workload**. It will open a new page, which is seen in Figure 6-6 on page 167.

- This page is the welcome page for the wizard that helps you define the workload partition. After reading the text, click **Next**.

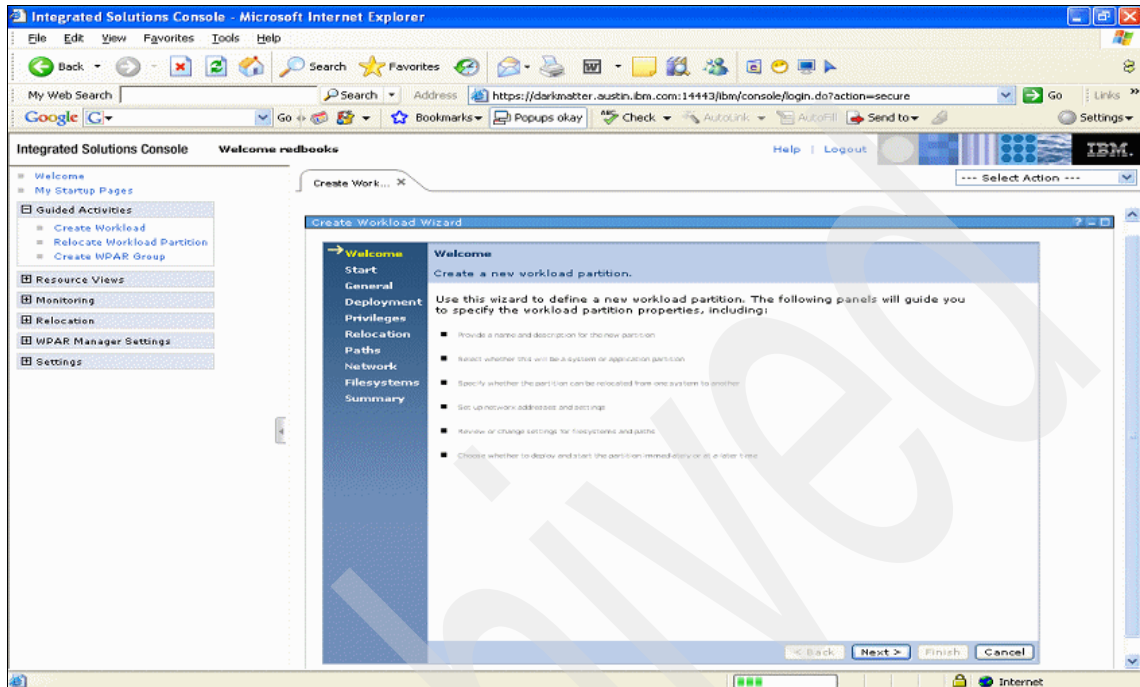


Figure 6-6 Create a new workload partition

5. In Figure 6-7, there are two options:
- Create a new workload partition.
 - Create from an existing workload partition.
- Select **Create a new workload partition**. Click **Next**.

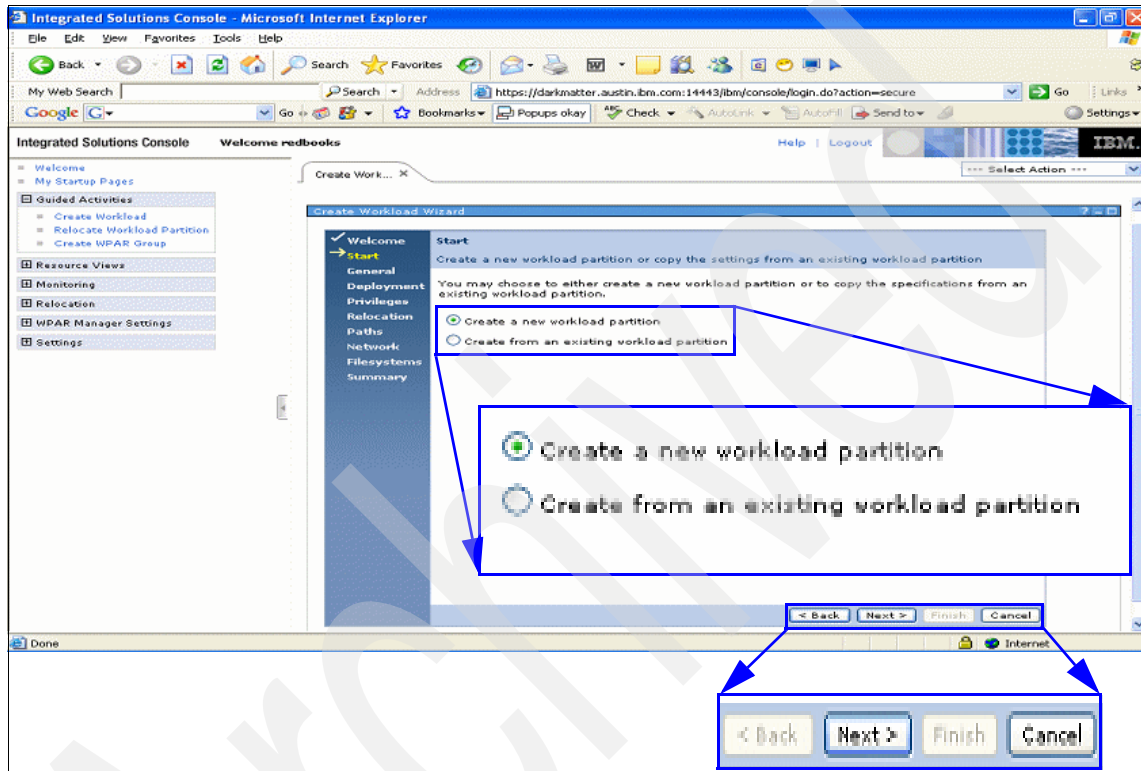


Figure 6-7 Create a new workload partition

6. In Figure 6-8 on page 170, select the partition type:
 - System Workload Partitions
 - Application Workload Partitions

For this example, select **System Workload Partitions**. There is a required field, Partition name. In the example, it is called Mars. The description is an optional field for user information for ease of identification. In the example, it is The Red Planets http Server. An optional password field is enabled for system partitions for the root user inside of the system WPAR. This is only valid if the partition is deployed. Application workload partitions do not have login capability, so this password field does not appear if you select Application Workload Partitions.

Select **Next** to move forward.

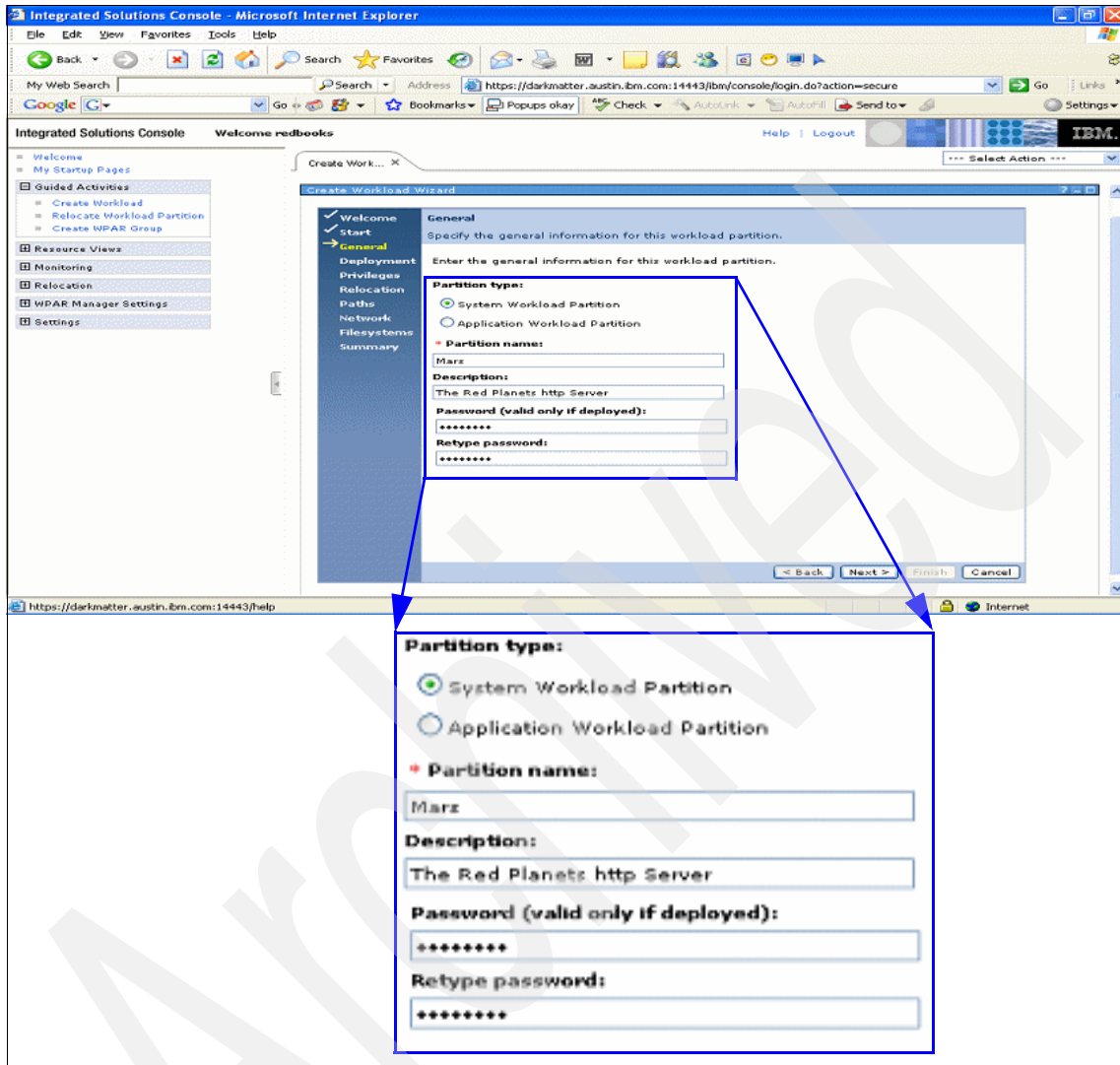


Figure 6-8 Create Workload Wizard, System Workload Partition

7. Figure 6-9 shows deployment settings. There are several options:
- Deploy this WPAR to an existing managed system.
 - Start the workload partition immediately upon deployment.
 - Start the workload partition each time the server starts.
 - Preserve local file systems on server.
 - Force deployment of workload partitions even if there are errors. Only use this option in a test debug mode. This will help prevent unstable application issues.

In the Select area, there are two Global Environment servers from which to choose: `epp182` or `saltgrass`. It does not matter which one is selected, but one must be selected because the check box `Deploy this WPAR to an existing managed system` is checked. If this check box is not selected, the profile can be stored for deployment and activated at a later time. Click **Next**.

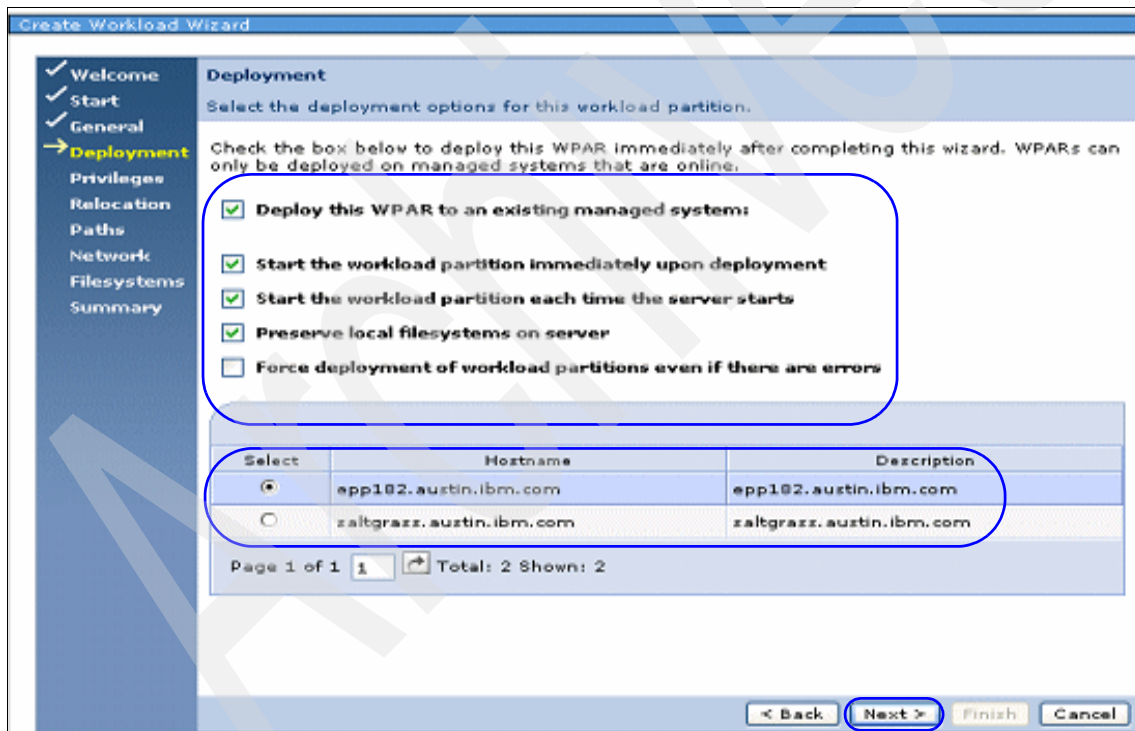


Figure 6-9 Deployment settings

8. Figure 6-10 shows the role-based access control (RBAC) Privileges window (system WPARs only).

If you create a system WPAR, it has a set of privileges within the global operating system on the target system. If you specified a managed system for deployment on the previous page, you can select privileges for the workload partition. The set of valid privileges is retrieved from the managed system, so if you did not select a system for deployment, you must select privileges later, when you actually deploy the WPAR. You can choose to assign the default operating system privileges or select a specialized set of privileges. For more information about privileges, see the operating system documentation on role-based access control (RBAC).

Figure 6-10 shows the default settings, which we used for this example. There are twelve pages of options from which to choose. Defining each option is outside the scope of this book. For these details, see *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430.

Select **Next**.

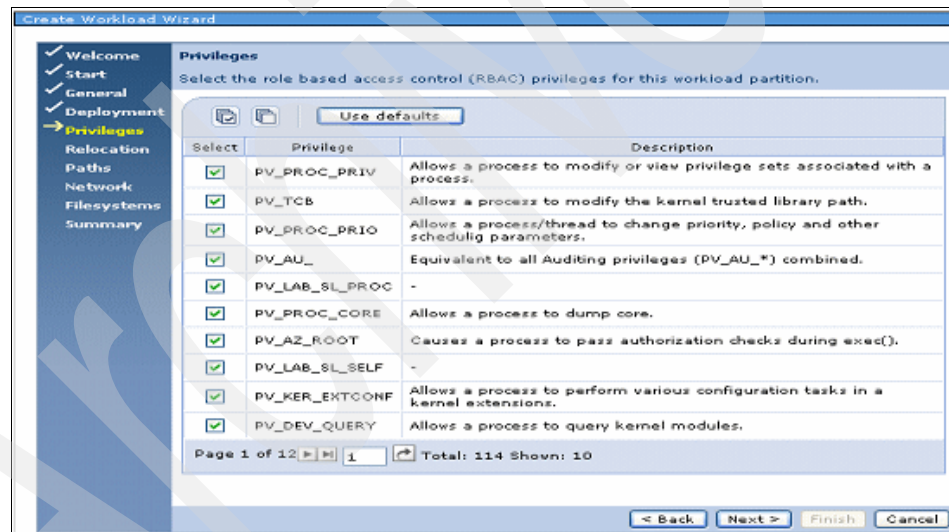


Figure 6-10 RBAC settings

9. Relocation requires the setup of a remote file system from an NFS server. This is discussed in 2.6, “WPAR mobility” on page 31. To enable this option for an application WPAR or a system WPAR, select Enable relocation in Figure 6-11. In our example, the WPAR cannot be relocated.

Select **Next**.

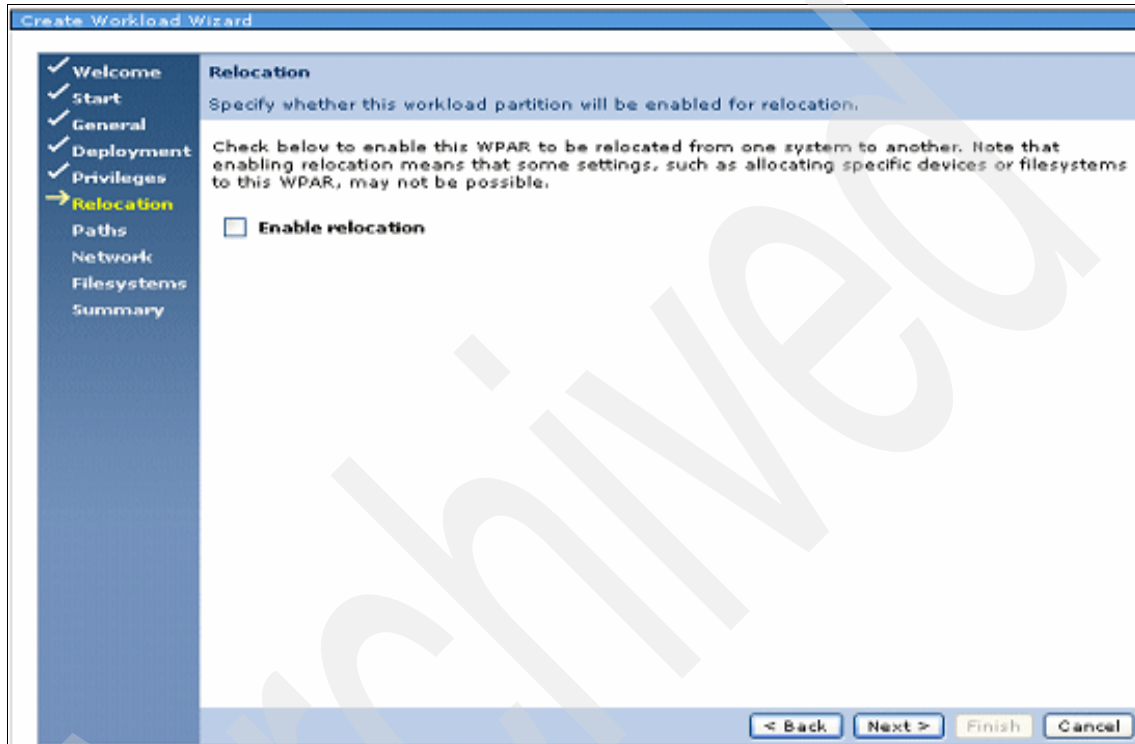


Figure 6-11 Enable relocation

10. In Figure 6-12, we set up paths and scripts. This area is used to start an application on startup. Use the path to control script field only if you are creating an application WPAR. Specify the full path or command string, including any command arguments, for the application that you want to run within the workload partition. This is the absolute path in the global file system, not the path as seen from within the WPAR. This is key for an application WPAR because this is how an application WPAR starts. For example, if there is a WebSphere application server that has an http server, put the full path to the start script in the Base directory. It is similar to this:

```
epp182: /was/IBM/WebSphere/AppServer/bin/startServer.sh Mars1
```

Use the Path to control script field for a script that must be run either before starting the workload partition or after stopping it. Enter the path (as seen from within the workload partition) to that script in this field.

Select **Next**.

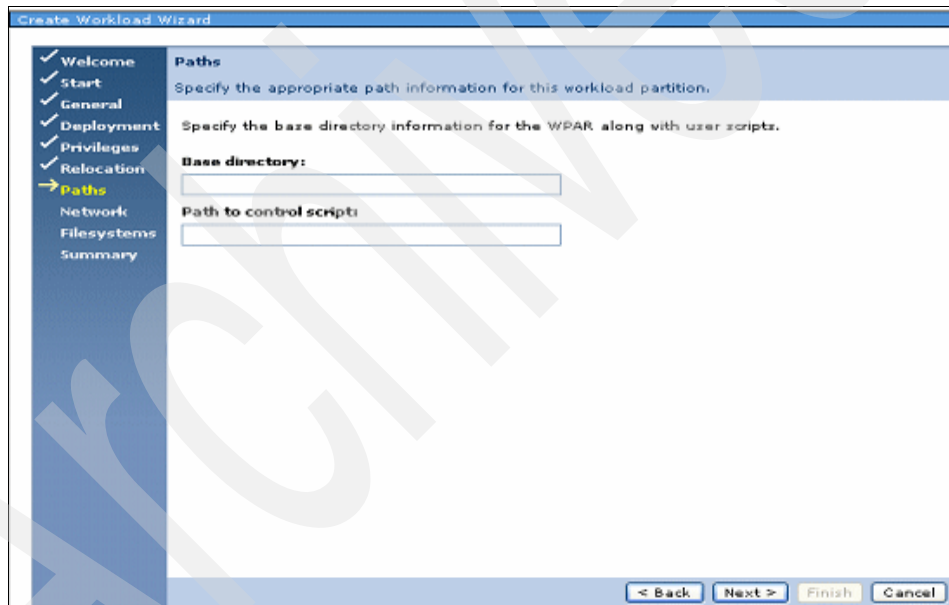


Figure 6-12 Setting the path and script

11. Workload partitions can have their own network addresses and host names, separate from the hosting system and from one another. To enable this workload partition to connect to the network, enter the network information on this page as seen in Figure 6-13 on page 176:

- The host name by default is the workload partition name that was entered on the General page. If you want a different host name, enter it here. For the example, we entered mars.
- Enter the IP address in the IPV4 format {xxx.xxx.xxx.xxx}
- Enter a subnet mask value. Our example is 255.255.255.0.
- The network interface field is not required if the network interface is not specified during creation. The WPAR software will attempt to detect an appropriate network interface to use for the WPAR. If you know the interface on the Global Environment, enter it. We selected ent3, for example.
- The Inherit name resolution settings from the global system copies name resolution files from the global system into the WPAR when the WPAR is first created. For example, this option copies the /etc/hosts, /etc/resolv.conf, and other applicable files into the workload partition. This field is displayed for system WPARs, because application WPARs always inherit name resolution settings from the global environment.

Select **Next**.

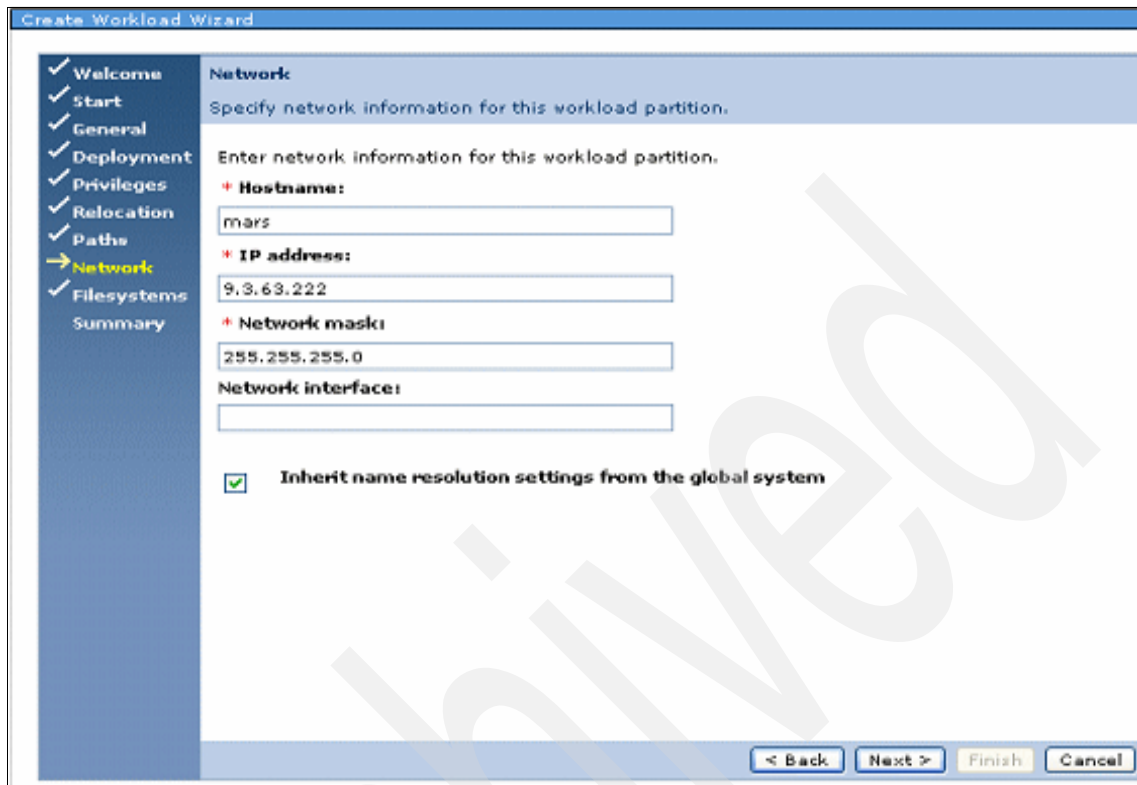


Figure 6-13 Network settings

12. The Filesystems window listed in Figure 6-14 on page 178 initially displays the default file systems mounted for the type of workload partition you are creating.

If you did not enable relocation for this WPAR, unless you need to change information or define additional file systems, you can leave this page without making any changes.

If you did enable relocation for this WPAR, you might see a series of messages on this page indicating that file systems are required. This is because mobility-enabled WPARs must have all their file systems defined as NFS-mounted or namefs read-only file systems, and WPAR Manager does not know which NFS server you plan to use. Therefore, you must manually specify NFS mounting details for each file system, as we describe under Add.

The table on this page shows the default file systems that are created for a workload partition, and selected properties for those file systems. To view all details or change a file system attribute, select an file system in the table and click Modify.

Virtual file system types that are valid types include nfs, jfs, jfs2, and namefs.

Click **Add** to define a new file system to be used by this workload partition. This displays a new page on which you enter required file system information.

Fields to complete on the Add page are:

- The WPAR Mount Point (Directory): The mount point, as perceived from within the workload partition, of the file system being added (for example, /usr)
- Virtual filesystem type: This is the type of file system being added.
For relocation-enabled WPARs, the only available file system types are NFS and Namefs. File systems /usr and /opt can be of the type Namefs, but only if the read-only (ro) mount option is set. All other writable file systems for mobile WPARs must be of the type NFS.
- Device path: This is the device path that is used for NFS file systems. Specify the exported directory on the NFS server for this file system. For namefs file systems, specify the global file system directory to be mounted within the WPAR.
- Remote node hosting this file system: Specify the host name of the NFS server hosting this file system.
- Mount Options: Enter any required mount options here. For more information about the various file system options, refer to your base workload partition documentation.

Select **Next**.

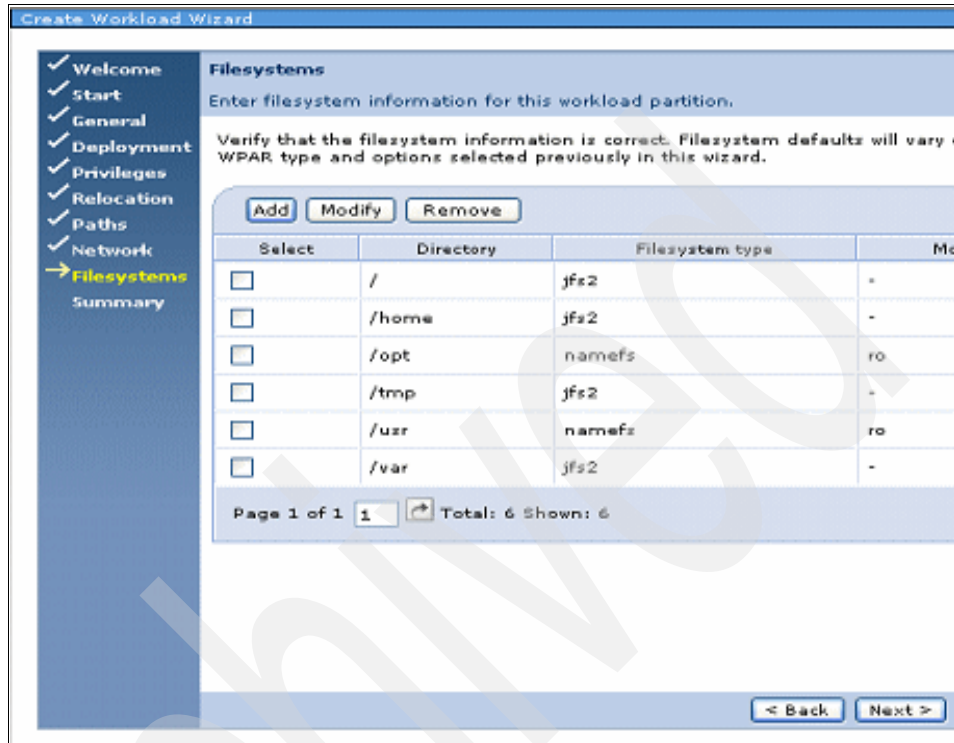


Figure 6-14 Filesystems settings

13. The final page of the wizard displays a summary of the information that you have entered or selected on the previous pages. Our example in Figure 6-15 on page 179 displays general, deployment, and paths information.

Select **Next**.

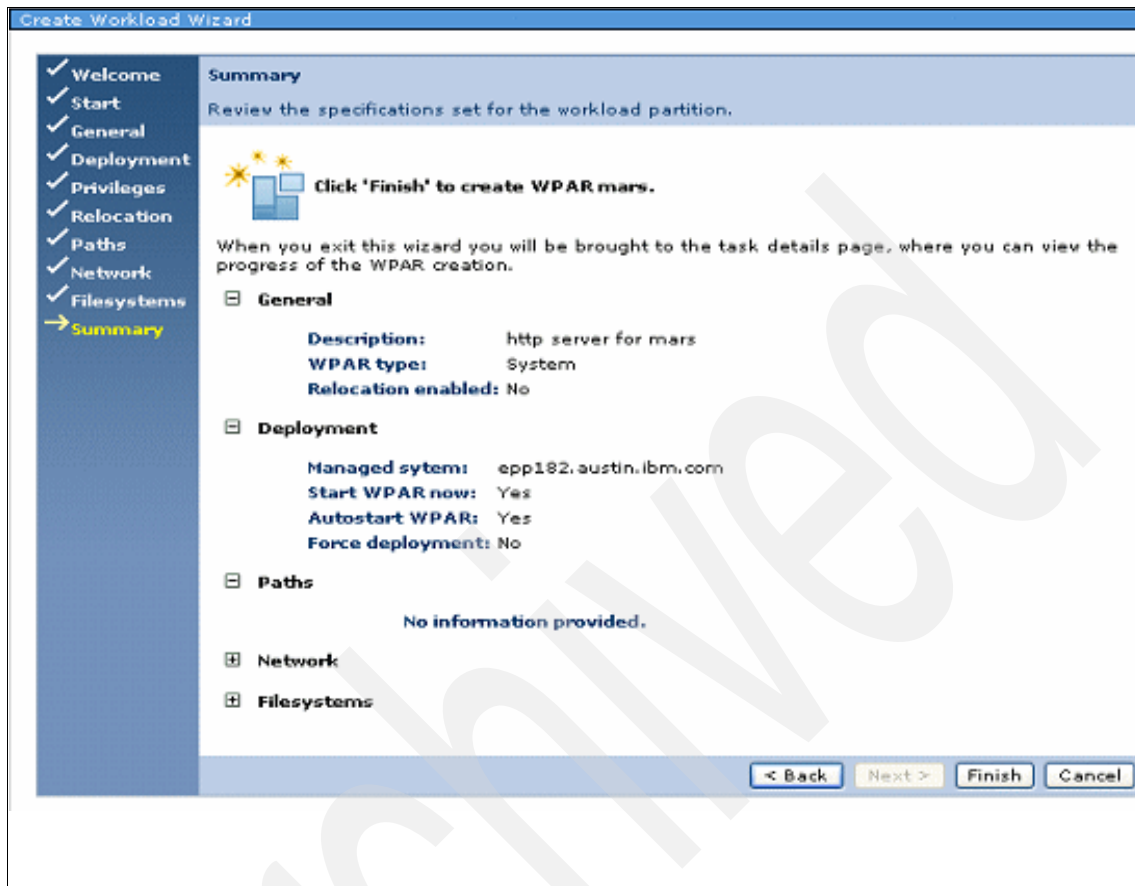


Figure 6-15 Summary page (part 1 of 2)

14. In Figure 6-16 on page 180, the network and file systems information is displayed. To make a change, click **Back** until you reach the desired page.

Click **Finish** to create the workload partition with the desired settings.

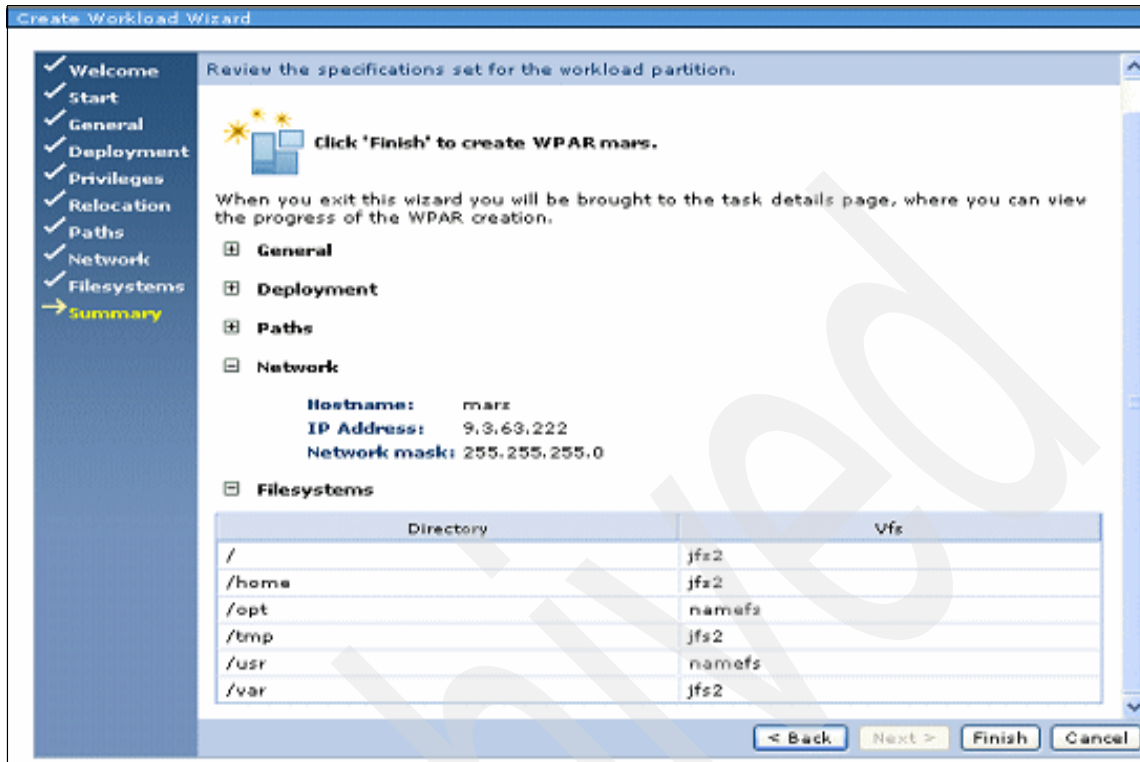


Figure 6-16 Summary (part 2 of 2)

In Figure 6-17, the wizard has completed the tasks and exited. The WPAR has been configured properly, validated through a number of checks, and then started. The window shows “Deploy Workload Partition completed successfully.” Now you can log in with `clogin` or `telnet` and applications can be started.

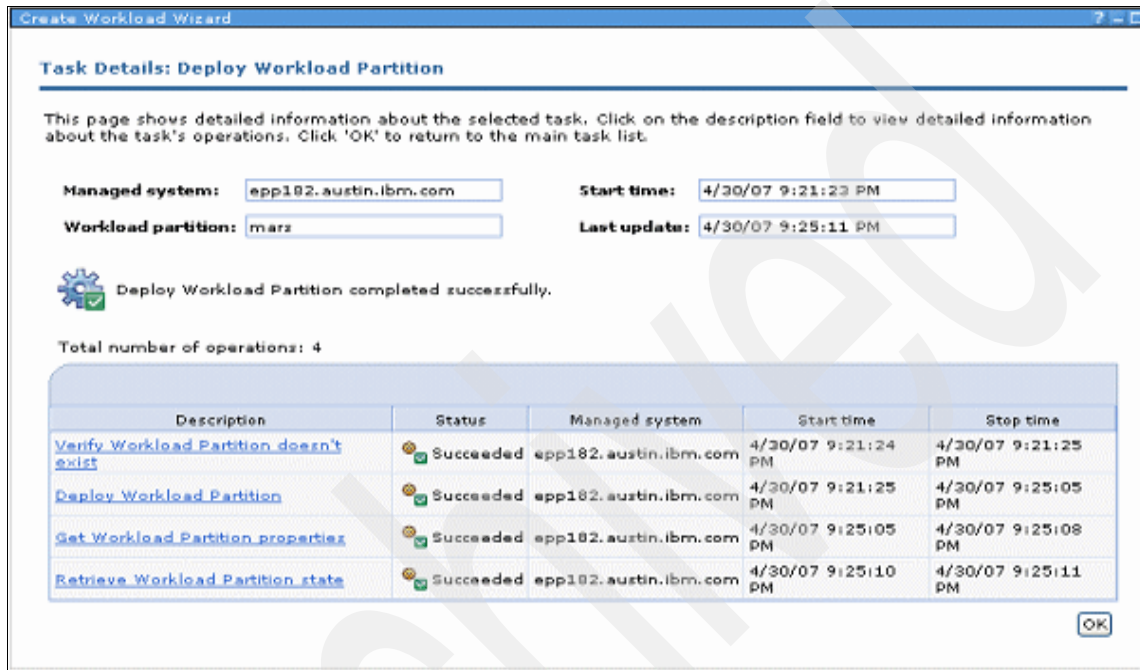


Figure 6-17 Deployed workload partition

6.10 Creating an application WPAR

The application WPAR creation process is almost the same as the system WPAR creation process. This section shows a step-by-step process of this creation using the WPAR Manager Wizard. The first three steps are the same and they can be reviewed at Figure 6-5 on page 166, Figure 6-6 on page 167, and Figure 6-7 on page 168. The rest of the steps are:

1. In Figure 6-18 on page 182, the Application Workload Partition has been selected to create a WPAR with the partition name of `venus`, which has been designated as `Venus http Server`.

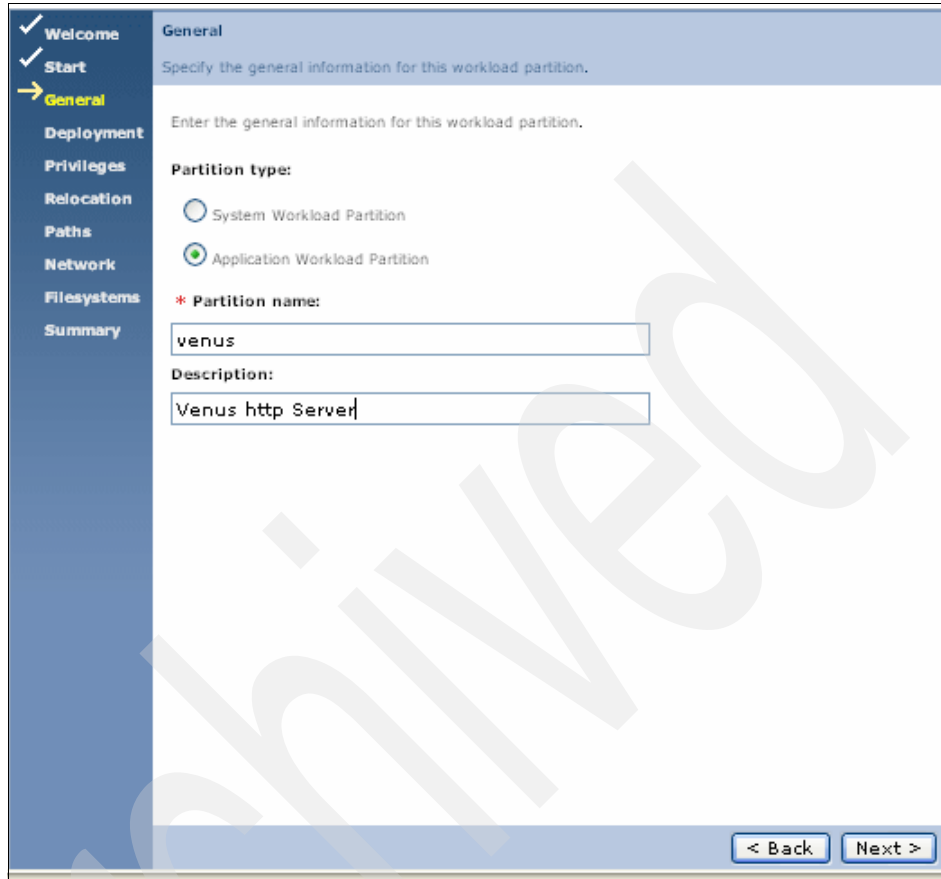


Figure 6-18 Selection of application workload partition server

2. In Figure 6-19 on page 183, if an application WPAR has been selected earlier, there are two options in this section:
 - Deploy this WPAR to an existing managed system
If you select the check box to deploy the WPAR to a managed system, a table is displayed listing all the available managed systems. Select one system from the table.
 - Force deployment of workload partitions even if there are errors
Check this box only to cause the WPAR to be deployed even if certain configuration errors are detected. See the man page for mkwpar -F for more information.

We selected the first option, Deploy this WPAR to an existing system. In our example, three Global systems are available from which to choose, and we selected epp176.

Select **Next**.

Select the deployment options for this workload partition.

Check the box below to deploy this WPAR immediately after completing this wizard. WPARs can only be deployed on managed systems that are online.

Deploy this WPAR to an existing managed system:

Force deployment of workload partitions even if there are errors

Select	Hostname	Description
<input checked="" type="radio"/>	epp176.austin.ibm.com	epp176.austin.ibm.com
<input type="radio"/>	epp182.austin.ibm.com	epp182.austin.ibm.com
<input type="radio"/>	saltgrass.austin.ibm.com	saltgrass.austin.ibm.com

Page 1 of 1 Total: 3 Shown: 3

< Back Next > Finish Cancel

Figure 6-19 Select a Global Environment on which to deploy the WPAR

3. Figure 6-19 on page 183 shows that there are no settings for RBAC in the application WPAR.
Select **Next**.



Figure 6-20 RBAC settings

4. On the Relocation page of the wizard, you can choose whether the workload partition you are creating is enabled for relocation; that is, whether it can be relocated from one managed system to another.

The option Enable relocation when checked enables this WPAR to be relocated. Note that enabling relocation means that certain settings, such as allocating specific devices or file systems to this WPAR, might not be possible. To be relocated, WPARs must use shared file systems. For more information about the restrictions that apply to relocated WPARs, refer to the base operating system WPAR documentation. In our example of Figure 6-21 on page 185, we did not select relocation. Select **Next**.

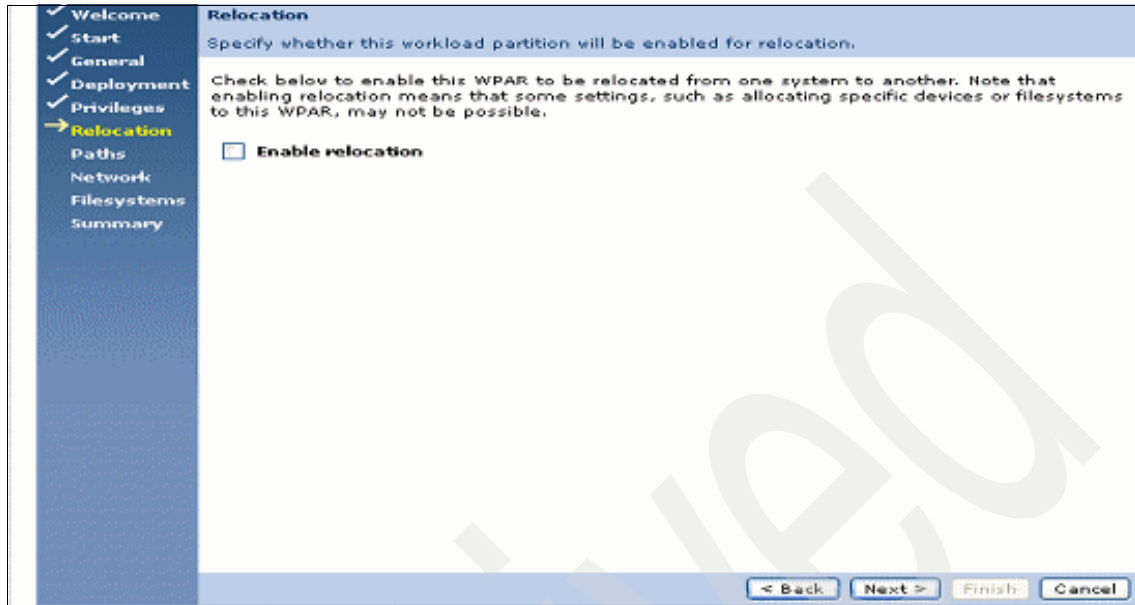


Figure 6-21 Enable relocation

5. In Figure 6-22 on page 186, the Path to executable field is used only if you are creating an application WPAR. Specify the full path or command string, including any command arguments, for the application that you want to run within the workload partition. This is the absolute path in the global file system, not the path as seen from within the WPAR. This is key for an application WPAR, because this is how it starts. For example, if there is a WebSphere application server that has an http server, this is the place to put the full path to the start script in the Base directory. It looks similar to this:

```
/was/IBM/WebSphere/AppServer/bin/startServer.sh venus1
```

In the Path to control script field, if there is a script that must be run either before starting the workload partition or after stopping it, enter the path (as seen from within the workload partition) to that script in this field. Select **Next**.

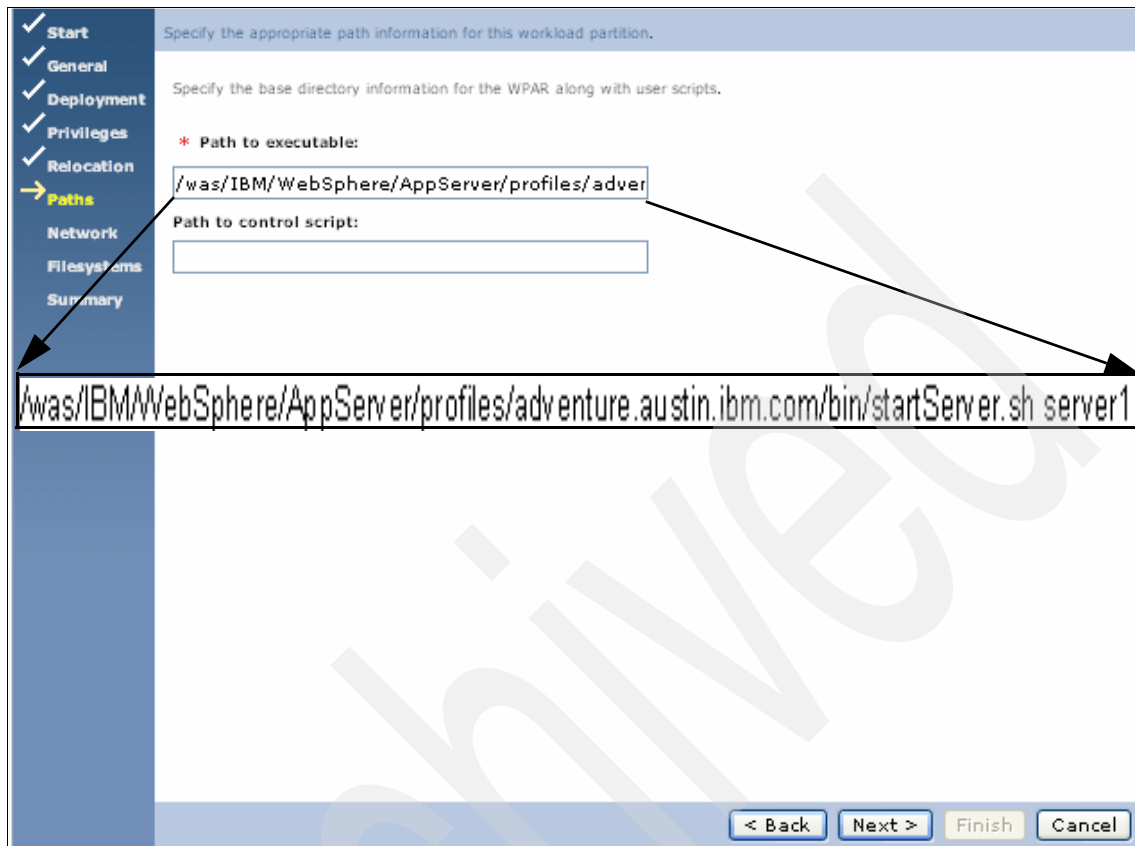


Figure 6-22 Application executable path

6. Workload partitions can have their own network addresses and host names, separate from the hosting system, and from one another. To enable this workload partition to connect to the network, enter network information on this page as shown in Figure 6-23 on page 187. Select **Next**.

Specify network information for this workload partition.

Enter network information for this workload partition.

Hostname:
venus

IP address:
9.3.6.12

Network mask:
255.255.255.0

Network interface:
en1

< Back Next > Finish Cancel

Figure 6-23 Network settings

7. If you did not enable relocation for this WPAR, unless you need to change information or define additional file systems, you can leave this page without making any changes. In Figure 6-24 on page 188, the default was selected and no changes or adds are made. Select **Next**.

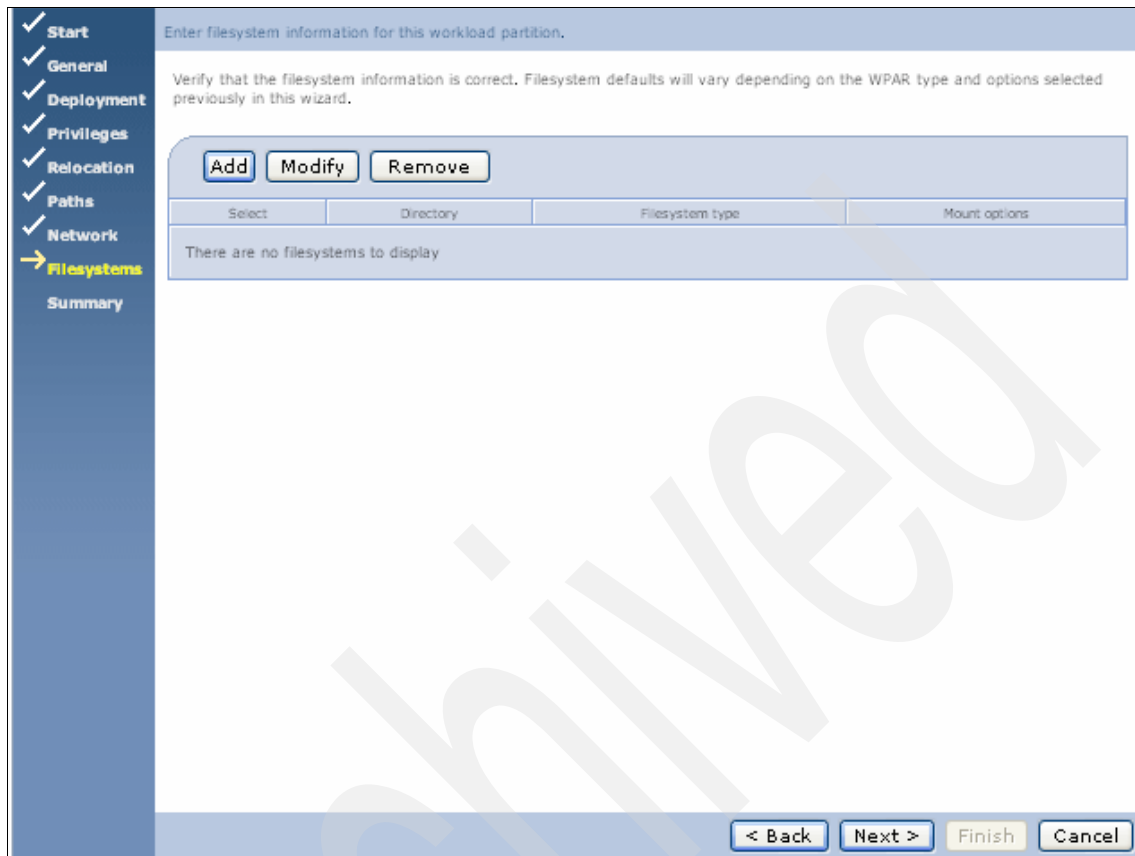


Figure 6-24 File system configuration for this application WPAR

8. The final page of the wizard displays a summary of the information that you have entered or selected on the previous pages. In Figure 6-25 on page 189, on the example page, General, Deployment, and Paths specifications for the workload partition information appear. Figure 6-25 on page 189 displays the executable script to start the http application. Select **Next**.

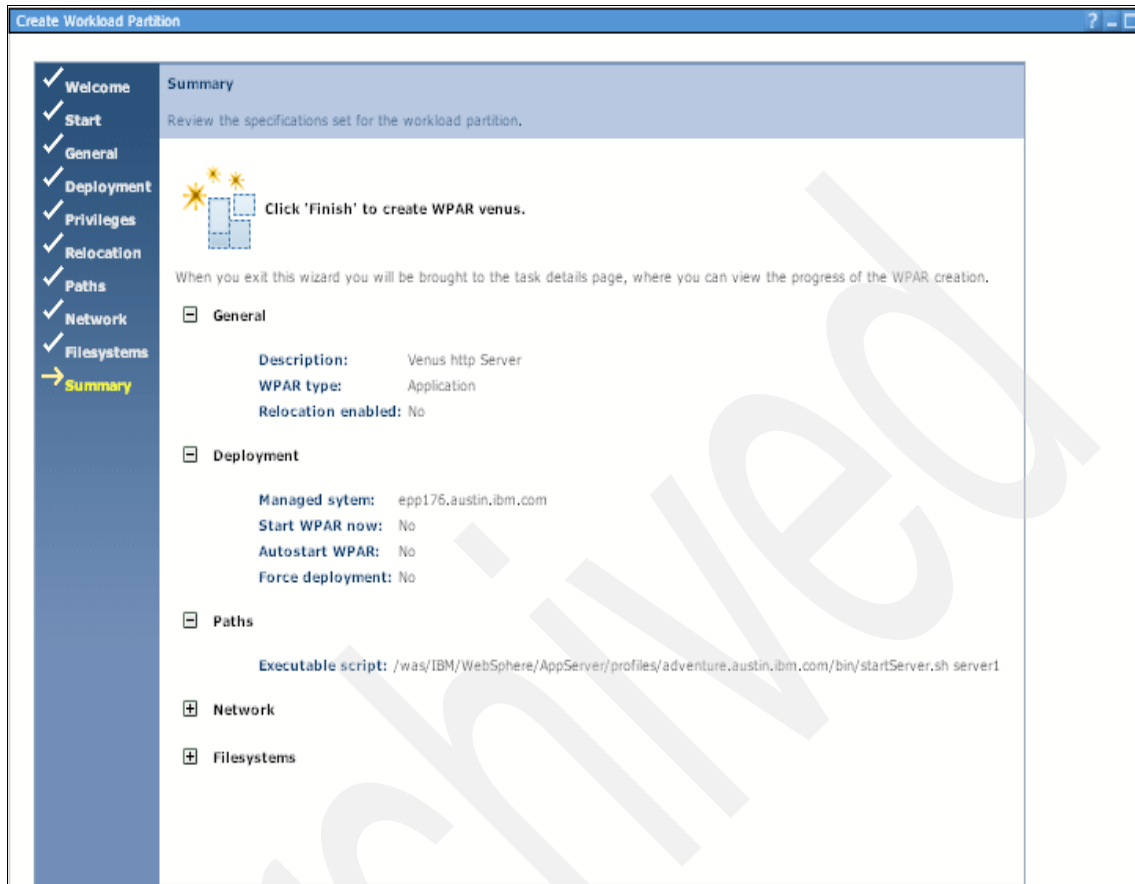


Figure 6-25 Summary: General, Deployment, and Paths (Part 1 of 2)

9. In Figure 6-26 on page 190, the network and file systems information is displayed. To make a change, click Back until you reach the desired page. Click Finish to create the workload partition with the desired settings. There are no settings for the file systems, because venus is a shell from the global environment.

Select **Next**.

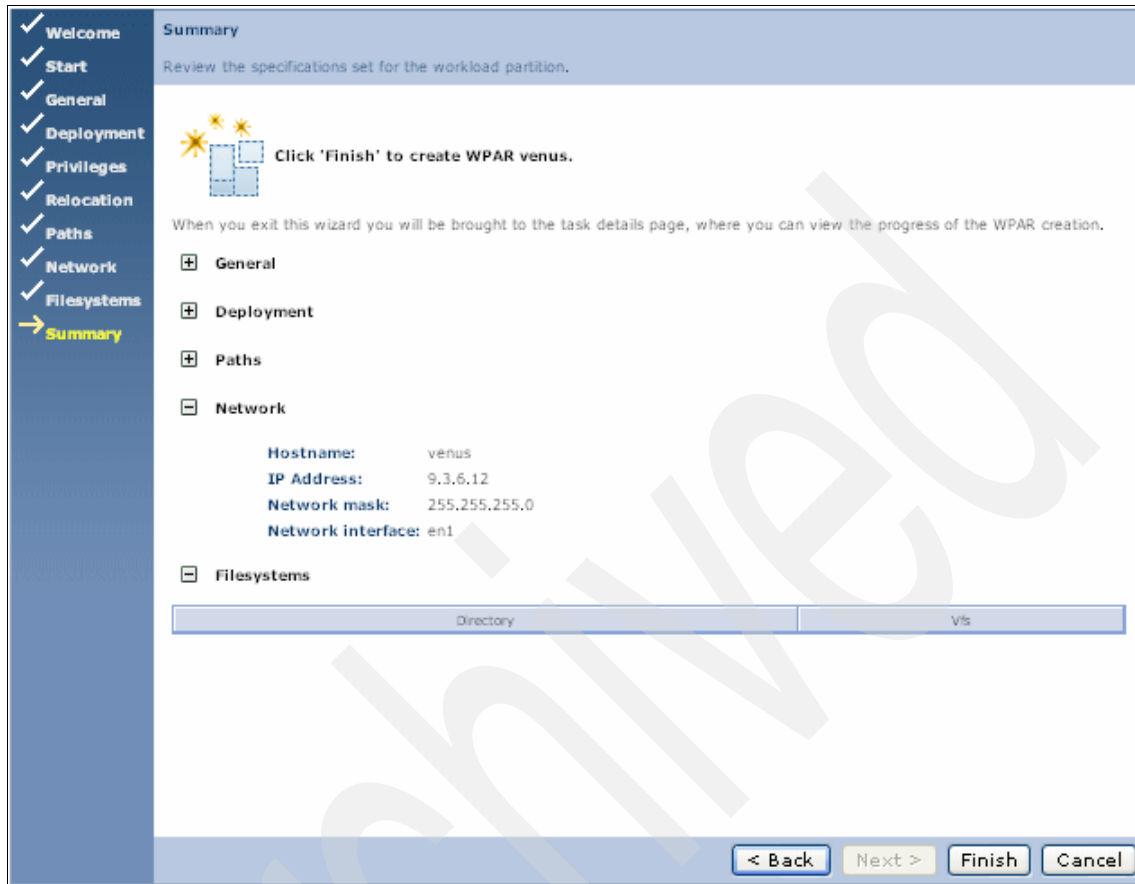


Figure 6-26 Summary network and file systems (Part 2 of 2)

10. To see if the state of the WPAR, click **Resource Views** to the upper left corner. This view shows three options:

- Managed Systems. These are Global systems.
- Workload Partitions. Select this option to see the state of all WPARs.
- Workload Groups. We will discuss this in this chapter.

Figure 6-27 on page 191 shows the status of venus. It is active and hosted on saltgrass. Select **Next**.

<input type="checkbox"/>	Pluto	System	Defined		saltgrass.austin.ibm.co	-	jovian
<input type="checkbox"/>	mars	System	Active		epp182.austin.ibm.com	Red Planet http Server	DefaultWPARGroup
<input type="checkbox"/>	Saturn	System	Transitional		saltgrass.austin.ibm.co	Saturn's http Server	DefaultWPARGroup
<input checked="" type="checkbox"/>	venus	Application	Active		saltgrass.austin.ibm.co	venus http server	DefaultWPARGroup
<input type="checkbox"/>	moon	Application	Active		epp176.austin.ibm.com	yes app	DefaultWPARGroup

Figure 6-27 WPAR Manager interface showing the status of WPARs

Archived

The activation of venus started the WebSphere application. This was done from the executable script we listed in Figure 6-22 on page 186. Open a Web browser and type the name or network address and full path to our application. The http server will serve pages as seen in Figure 6-28.

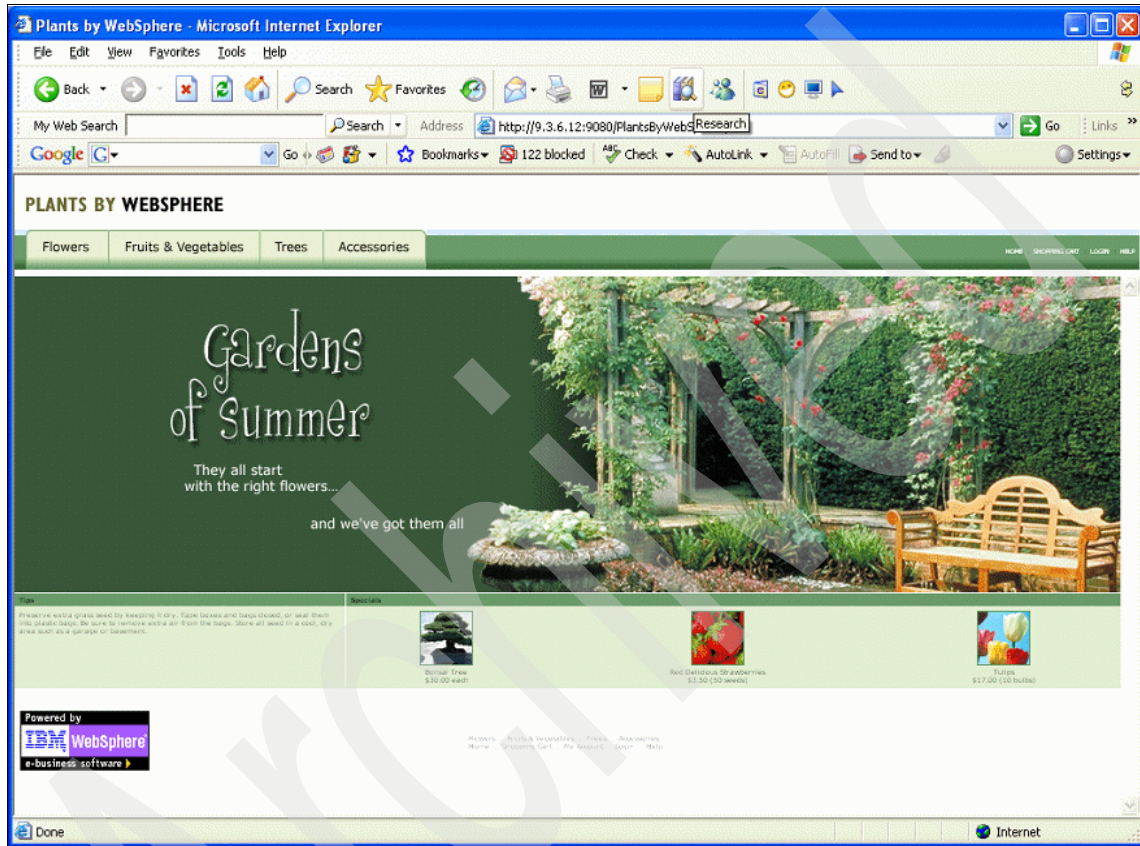


Figure 6-28 Application running inside of application WPAR

6.11 Migrating an application WPAR

In the next activity, the WPAR named `elation`, which is an application WPAR, will be moved from `sal tgrass` (global) to `epp176`, which is another (global) on a different physical system. The applications will not be interrupted. A user only sees a short pause at most from the browser. This does not require a refresh, and no in-flight data will be lost.

In Example 6-9, the output of **lswpar** shows all WPARs in the global environment. Note the WPAR named `elation` is active. This is the application WPAR we will migrate.

Example 6-9 View of all WPARs on saltgrass (global)

```
root: saltgrass:/ --> lswpar
Name      State Type Hostname      Directory
-----
Pluto     D      S      elation.austin.ibm.com  /wpars/elation
Saturn    T      S      adventure.austin.ibm.com /wpars/Saturn
earth     A      S      earth                /wpars/earth
elation   A      A      elation                /
thebe     A      S      thebe                  /wpars/thebe
root: saltgrass:/ -->
```

Moving to the global server `epp176` and running the **lswpar** show that `moon` is the only WPAR on this system, which is shown in Example 6-10.

Example 6-10 View of all WPARs on epp176 (global)

```
root: epp176:/ --> lswpar
Name      State Type Hostname Directory
-----
moon      A      A      moon                  /
root: epp176:/ -->
```

The steps to relocate a workload partition are:

1. At this point, you can log in to the UI and select **Guided Activities** in the upper left corner of the welcome page. Then, select **Relocate Workload Partitions**, which will present the page seen in Figure 6-29. Read the page and then click **Next** to move forward.

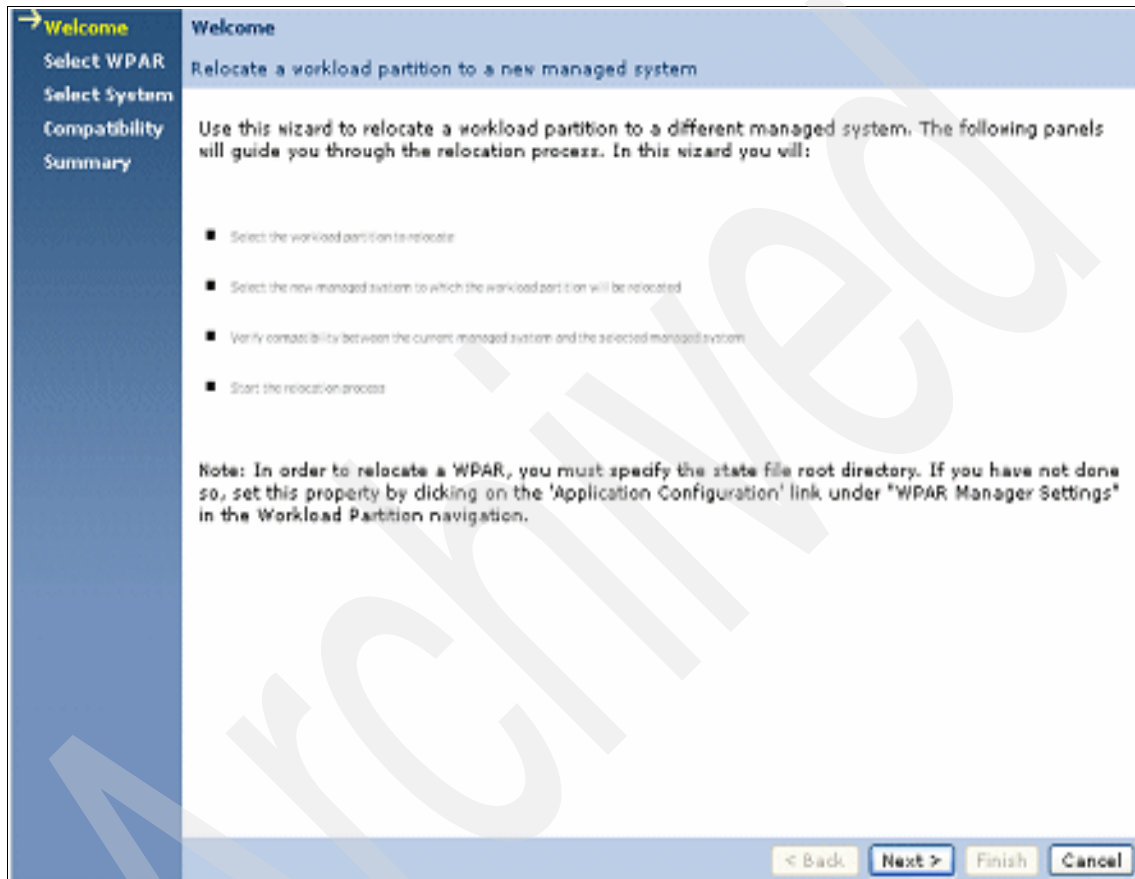


Figure 6-29 The start of relocating a WPAR

2. In Figure 6-30, select **Browse** to search for the WPARs that have been created to allow mobility. Select **Next**.

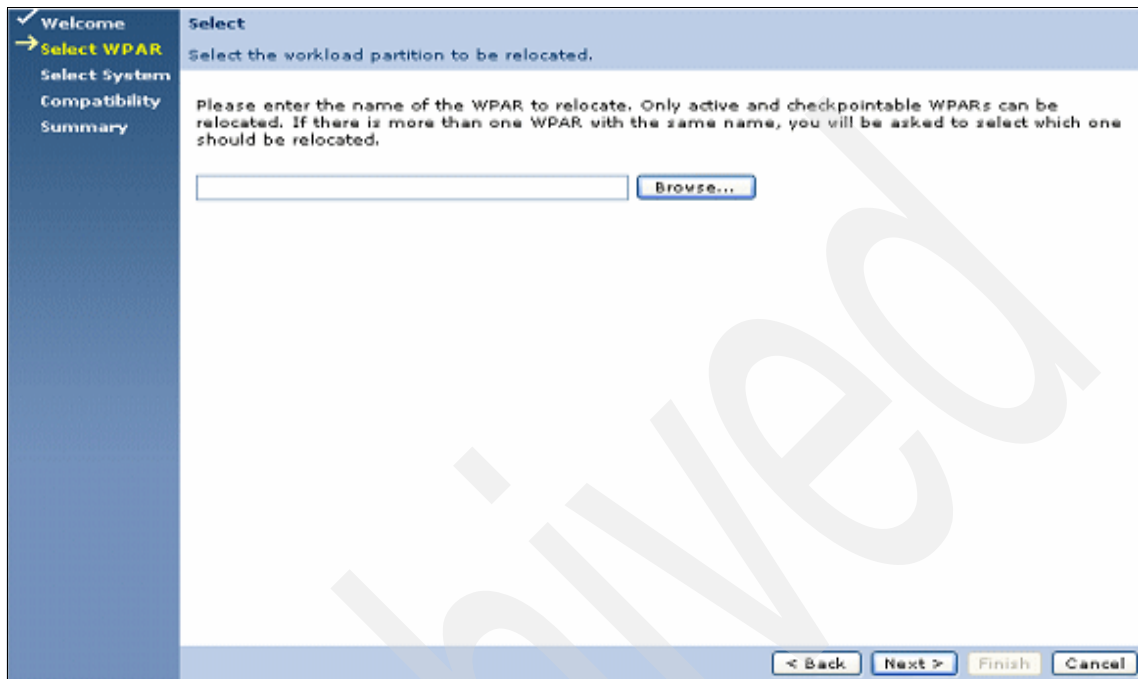


Figure 6-30 Browse for a WPAR to be relocated

3. After selecting the Browse feature, the UI returns an application WPAR named elation that can be relocated. Figure 6-31 shows the WPAR name is elation, Type = Application, State = Active, and Managed system (global) is saltgrass. Figure 6-31 is a subset of the complete window. Select **OK** to move to the next window.

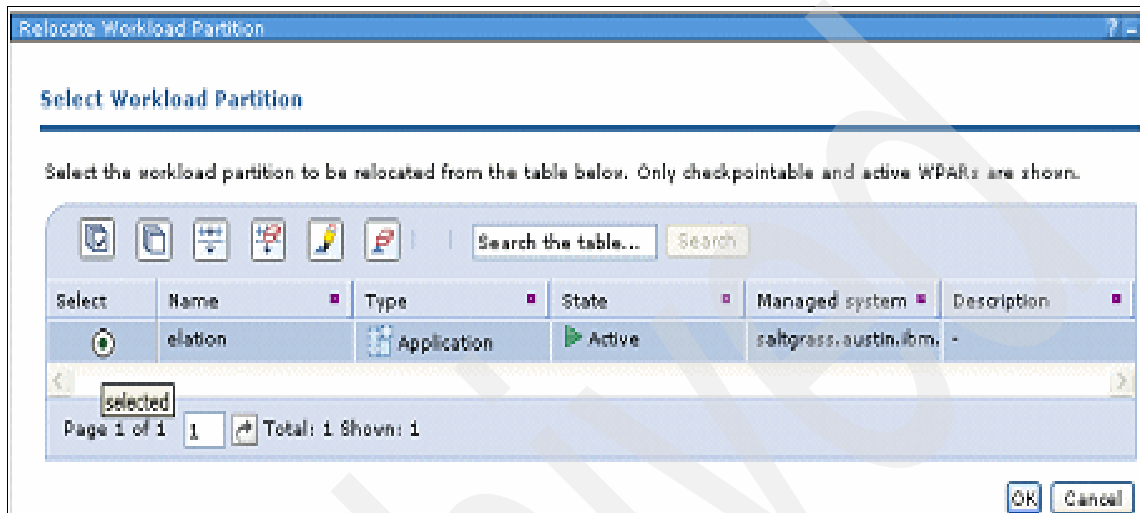


Figure 6-31 Showing a WPAR that is mobile

4. Figure 6-32 shows the WPAR that was selected. Click **Next**.

✓ Welcome
→ Select WPAR
Select System
Compatibility
Summary

Select
Select the workload partition to be relocated.

Please enter the name of the WPAR to relocate. Only active and checkpointable WPARs can be relocated. If there is more than one WPAR with the same name, you will be asked to select one.

elation

< Back

Figure 6-32 WPAR to relocate

- In Figure 6-33, the name of the WPAR to relocate is in the Selected WPAR field and saltgrass is the current managed or Global system. Choose Best available to let the WPAR Manager choose the best available system for relocation, based on previously set policies and the amount of computing resources available on candidate systems. The system chosen as “best available” is displayed below this choice.

Only two-way compatible systems, which are those capable of being moved safely from the departure system to the arrival system and back, are examined when choosing a best available system. If the system cannot find a best available server, the option is disabled and only Manual Selection is available.

If you choose Manual Selection, enter the name of the arrival system. Possible host names will be suggested as you type the host name of your system. Or click Browse to select from a list. All systems and their compatibility status are shown. You can use the table filter to narrow the list of displayed servers. These systems are ranked according to factors, such as availability of computing resources. Select **Next**.

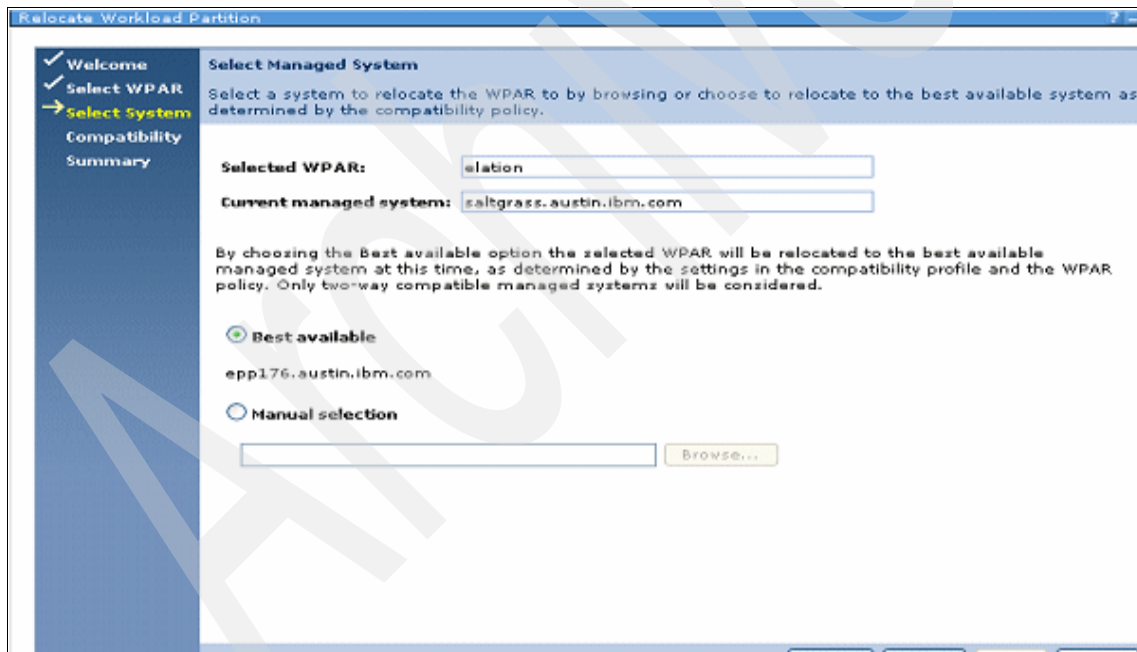


Figure 6-33 Selection of target global system

- Figure 6-34 displays the results of compatibility testing for the managed system selected on the previous page. Select **Finish**.

Compatibility Test Results
Confirm that the current managed system and the selected managed system are compatible.

Managed System saltgrass.austin.ibm.com is two-way compatible with Managed System epp176.austin.ibm.com.

You may relocate WPAR elation from Managed System saltgrass.austin.ibm.com to Managed System epp176.austin.ibm.com. You may also relocate it back from Managed System epp176.austin.ibm.com to Managed System saltgrass.austin.ibm.com in the future.

Summary Outbound critical Outbound optional

Summary of compatibility test results. For detailed results click the tabs above.

	Critical	Optional	Total failures
Outbound tests	Passed	Passed	0
Inbound tests	Passed	Passed	0

< Back Next > Finish Cancel

Figure 6-34 Compatibility Test Results

- In Figure 6-35, the relocate process is now in place and the application, which in our example is a WebSphere http server, is now in the process of moving from saltgrass (global) to epp172 (global) on a different System p server. Select **OK**.

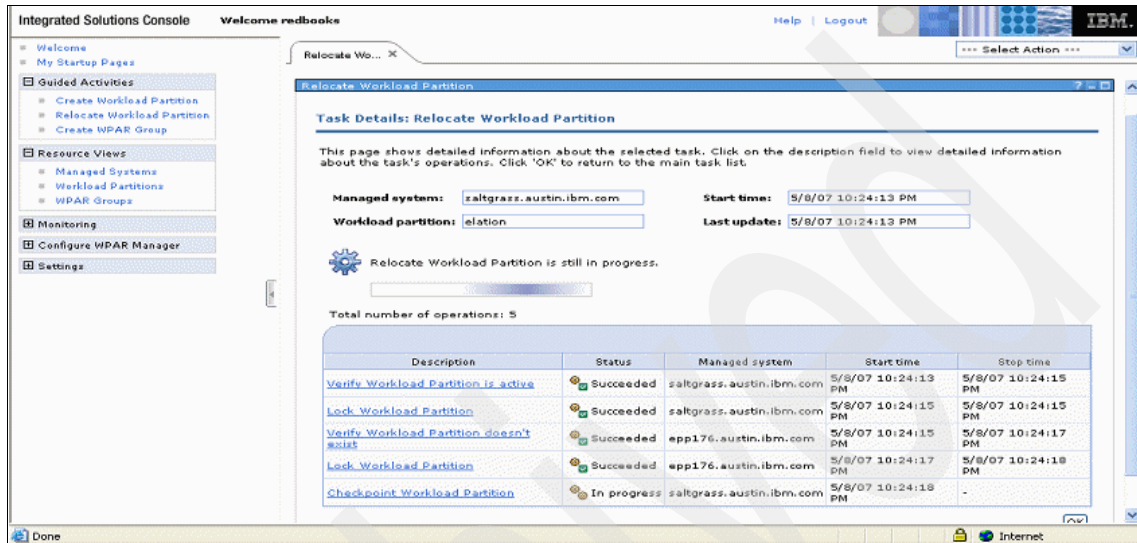


Figure 6-35 Relocate WPAR in progress

Within a few moments, the application will move from one system to another without interrupting in-flight data to the http server. From the client's point of view, the browser might seem to slow down, but no refresh is needed and no connection is lost. This is a very fast process. In Figure 6-36 the Web browser shows the http server's server pages. The URL and network information all stays the same as the migration completes.

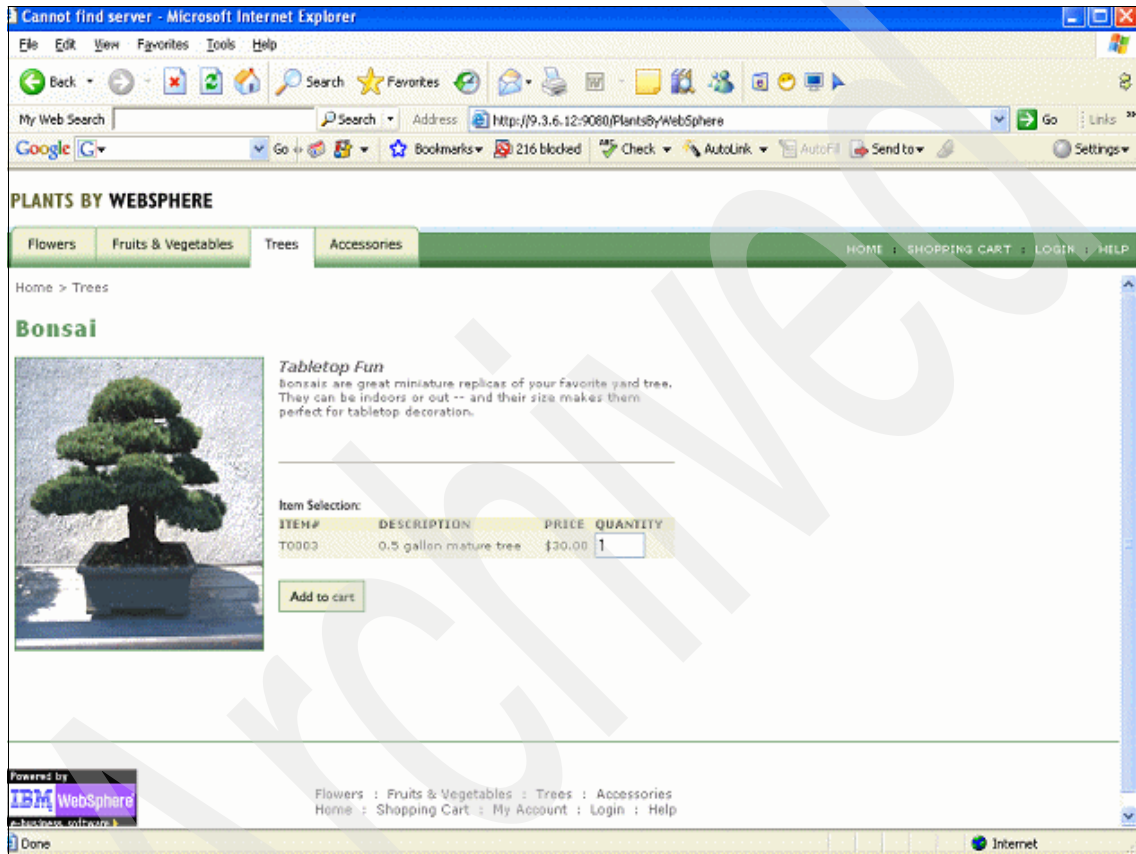


Figure 6-36 HTTP server running on new server

Looking back at the global WPAR listing in Example 6-10 on page 193, the `lswpar` command output listed only moon as a WPAR on epp176 (Global). Now, after completing the WPAR relocation steps just described, the `lswpar` command shows (in Example 6-11 on page 202) that now the application WPAR named e1ation is also an active application WPAR in epp176. The migration has been successful.

Example 6-11 lswpar on epp176

```
root: epp176:/ --> lswpar
Name      State Type Hostname Directory
-----
elation   A      A      elation   /
moon      A      A      moon      /
root: epp176:/ -->
```

In Example 6-12, the **lswpar** command shows that the application WPAR from Example 6-9 on page 193 called *elation* is no longer on this Global system, *saltgrass*.

Example 6-12 lswpar command

```
root: saltgrass:/ --> lswpar
Error finding stanza file for workload partition 'elation'.
Name      State Type Hostname      Directory
-----
Pluto     D      S      elation.austin.ibm.com /wpars/elation
Saturn    T      S      adventure.austin.ibm.com /wpars/Saturn
earth     A      S      earth                /wpars/earth
thebe     A      S      thebe                 /wpars/thebe
root: saltgrass:/ -->
```

6.11.1 Create a WPAR group

You can use groups to organize types of application and system WPARs for ease of administration and deployment. A *workload partition group* specifies a group of workload partitions and the relocation policy settings shared by all WPARs in the group.

The steps to create a WPAR group are:

1. To start the creation of groups, select **Guided Activities** in the upper left corner of the UI. Then select **Create WPAR Group**. This opens a window as seen in Figure 6-37. You must enter the WPAR Group name. This is followed by the WPAR Group description, which is optional. The policy-based relocation setting will enable automatic relocation for all WPARs in this WPAR Group. Select this box to allow all WPARs in the WPAR Group to be automatically migrated based upon the specified policy values. Click each heading listed in Figure 6-37 to open each page and complete the pages to create a new WPAR group.

Create Workload Partition Group

A workload partition group specifies a group of workload partitions, and the relocation policy settings shared by all WPARs in the group. Click each heading below to open each page, and complete the pages to create a new WPAR group.

→ WPAR Group General Properties

Specify a name and optional description for this WPAR group. By default, policy-based relocation is enabled. Uncheck the Automatic Relocation checkbox if you want to disable policy-based relocation for WPARs in this group.

+ WPAR Group name:
solar system http server

WPAR Group description:
the planets

Policy-based relocation:
 Enable policy-based relocation for all WPARs in this WPAR Group

- Add WPARs to WPAR Group
- Specify WPAR Group Policy Settings
- Specify WPAR Group Policy Metrics
- Managed Server Profiles Associated with WPAR Group
- Specify Optional Compatibility Tests

Figure 6-37 WPAR Group General Properties

- Figure 6-38 shows the WPARs currently assigned to this WPAR group. To add WPARs to the group, click **Add New**. This displays a new page listing all WPARs in the default WPAR group. WPARs are initially assigned to the default group when created. Click **Show all WPARs** to view WPARs assigned to other groups, as well as the default group. A WPAR can be a member of only one group. Adding it to the current group removes it from the default group, or any other group to which it might have been assigned. Note that the lists of WPARs on these pages include WPARs that are defined in the WPAR Manager's internal database, whether they are currently deployed on a managed system or not.

Select all the WPARs to add to the WPAR Group, then click **Done**. This returns to the initial page, where the newly-added WPARs are displayed in the list of WPAR group members.

To remove one or more WPARs from the WPAR group, select them in the table, and click **Remove**.

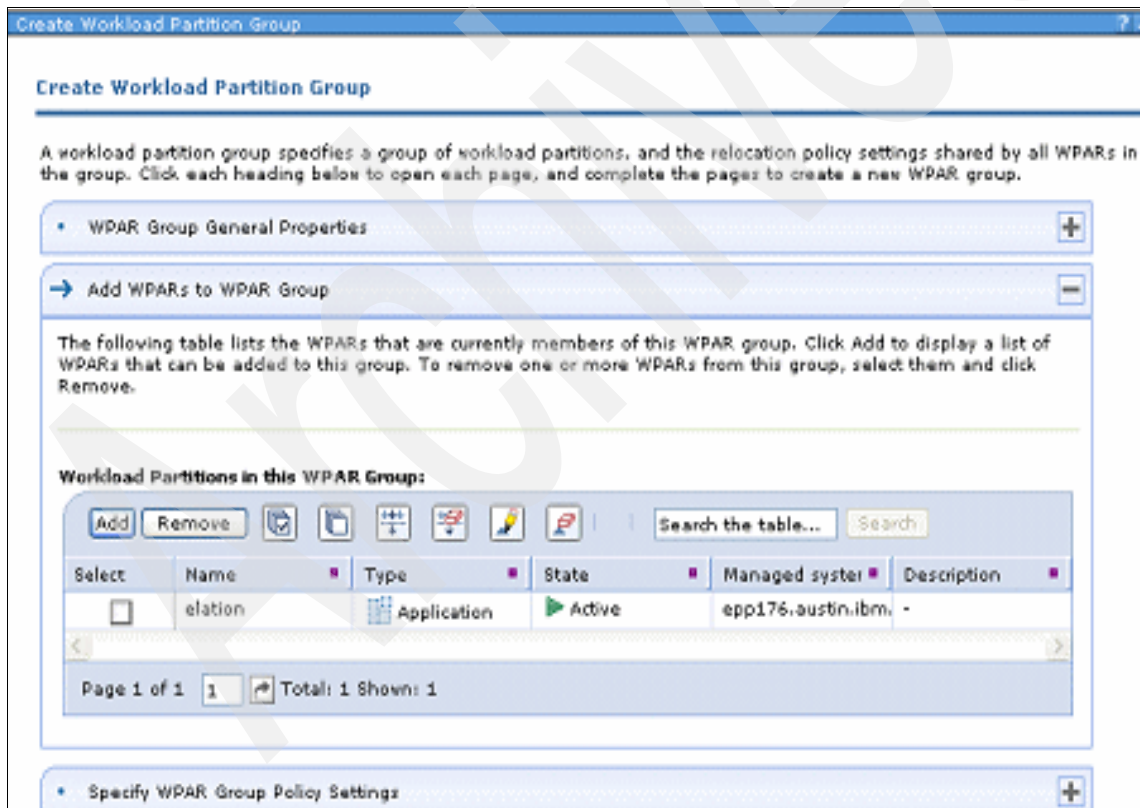


Figure 6-38 Add WPARs to WPAR Group

3. Figure 6-39 is one of two ways in which the relocation policy settings for this WPAR group are defined. On this pane, you can view or change several global policy settings. You can choose between launching a wizard to guide you through the steps or entering data directly into the fields on this pane. For this example, we use the Wizard.

To activate the Wizard interface, click **Launch Wizard**. Otherwise, proceed to the lower part of the pane to enter data.

Periodically, the average performance state of all WPAR instances deployed in the WPAR group is calculated. If the average performance state exceeds the limits specified by the Maximum and Minimum policy threshold values, a relocation event is generated, based on other properties defined in the WPAR Group definition. Maximum and minimum policy thresholds are unitless measures ranging between 0 and 100.

This multi-page table contains a number of time-out settings related to WPAR Manager operations. If any time-out value is exceeded, the operation is considered to have failed. The task of which the operation was a part is canceled, and the recovery process is started. If you need to change the default time-out values, you can adjust them in this table.

Create Workload Partition Group

A workload partition group specifies a group of workload partitions, and the relocation policy settings shared by all WPARs in the group. Click each heading below to open each page, and complete the pages to create a new WPAR group.

- WPAR Group General Properties +
- Add WPARs to WPAR Group +
- Specify WPAR Group Policy Settings -
- Specify WPAR Group Policy Metrics +

Specify the performance state thresholds for the WPAR group. The averaged performance state for all the WPARs in the group is compared to these thresholds. A performance state value higher than the maximum threshold defines when the WPARs in the group are "busy." A value less than the minimum threshold defines when the WPARs in the group are "idle." Threshold values can range between zero and one. The averaging period is the time window over which performance metrics are averaged for performance state analysis.

Thresholds:

- **Maximum Threshold:** 0.80
- **Minimum Threshold:** 0.20
- **Averaging Period (minutes):** 5

Figure 6-39 WPAR Group Policy Settings

4. The table in Figure 6-40 shows the performance metrics, which are currently being monitored for this WPAR group, and the limits, weights, and other characteristics for each metric.

To add metrics to be monitored for this WPAR group, click **Add New**. This displays a new page where you can select from a list of available metrics.

After adding a metric, the new metric is displayed in the table along with the characteristics that you defined.

To stop monitoring one or more metric, select the check box for each metric and click **Remove**. The removed metrics will no longer appear in the table.

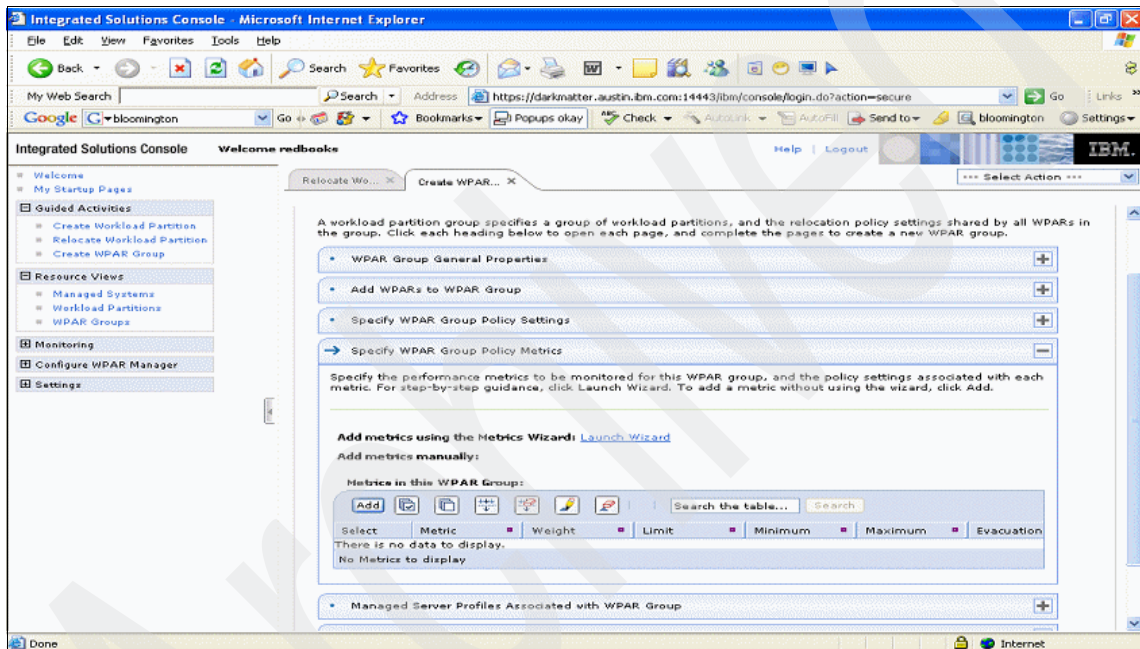


Figure 6-40 WPAR Group Policy Metrics

5. After selecting the Launch Wizard in Figure 6-40 on page 206, an introduction to add performance metrics window appears as seen in Figure 6-41. This is an information page only.

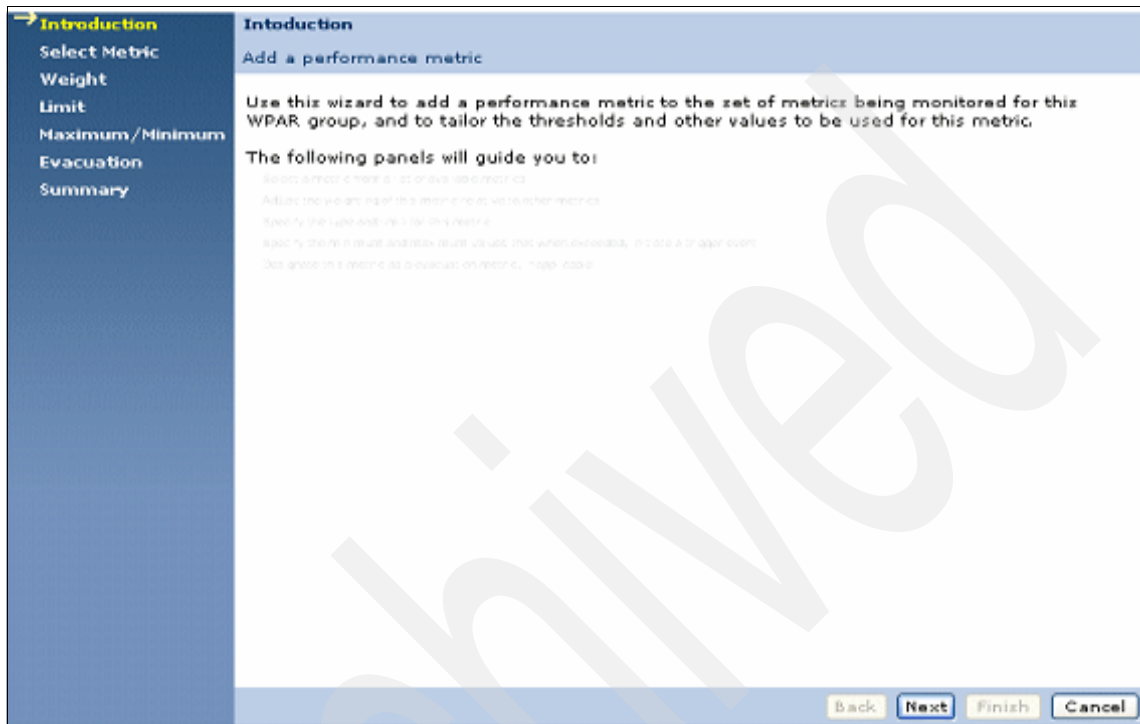


Figure 6-41 Introduction to the WPAR Metrics Wizard

6. Various metrics are available in Figure 6-42. Select the desired performance metric and click **Next**.

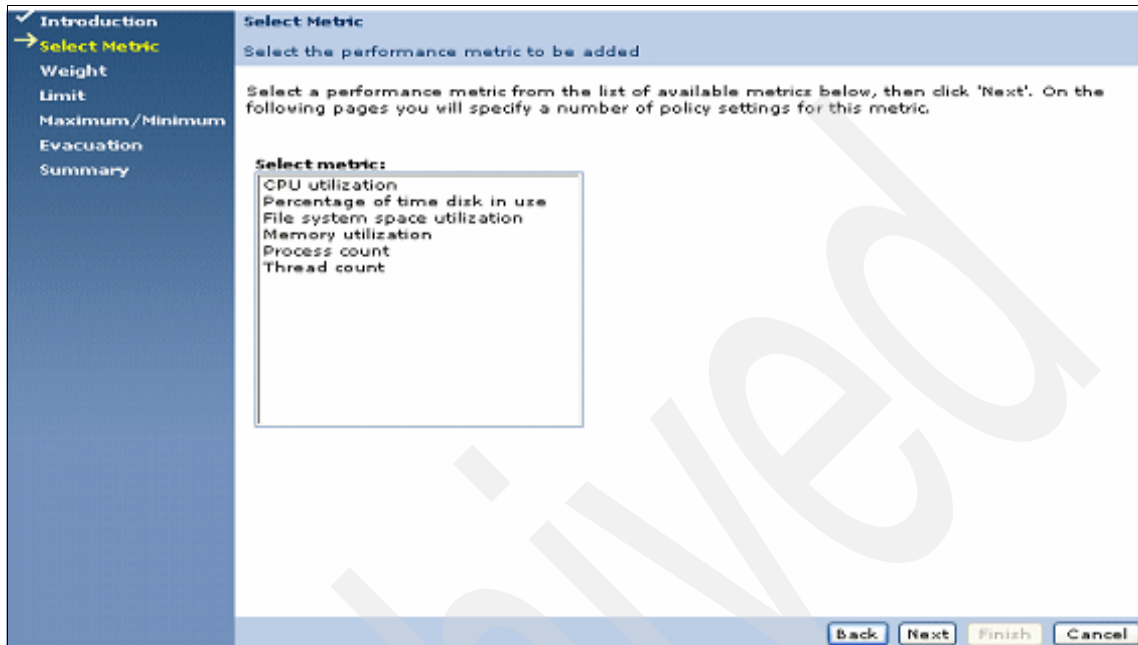


Figure 6-42 Selecting the metrics

- Specific weights can be set as seen in Figure 6-43. To find more information about a topic at any point, there is a question mark on the window in the upper right side of the topic page. This opens a slide as seen in Figure 6-43.

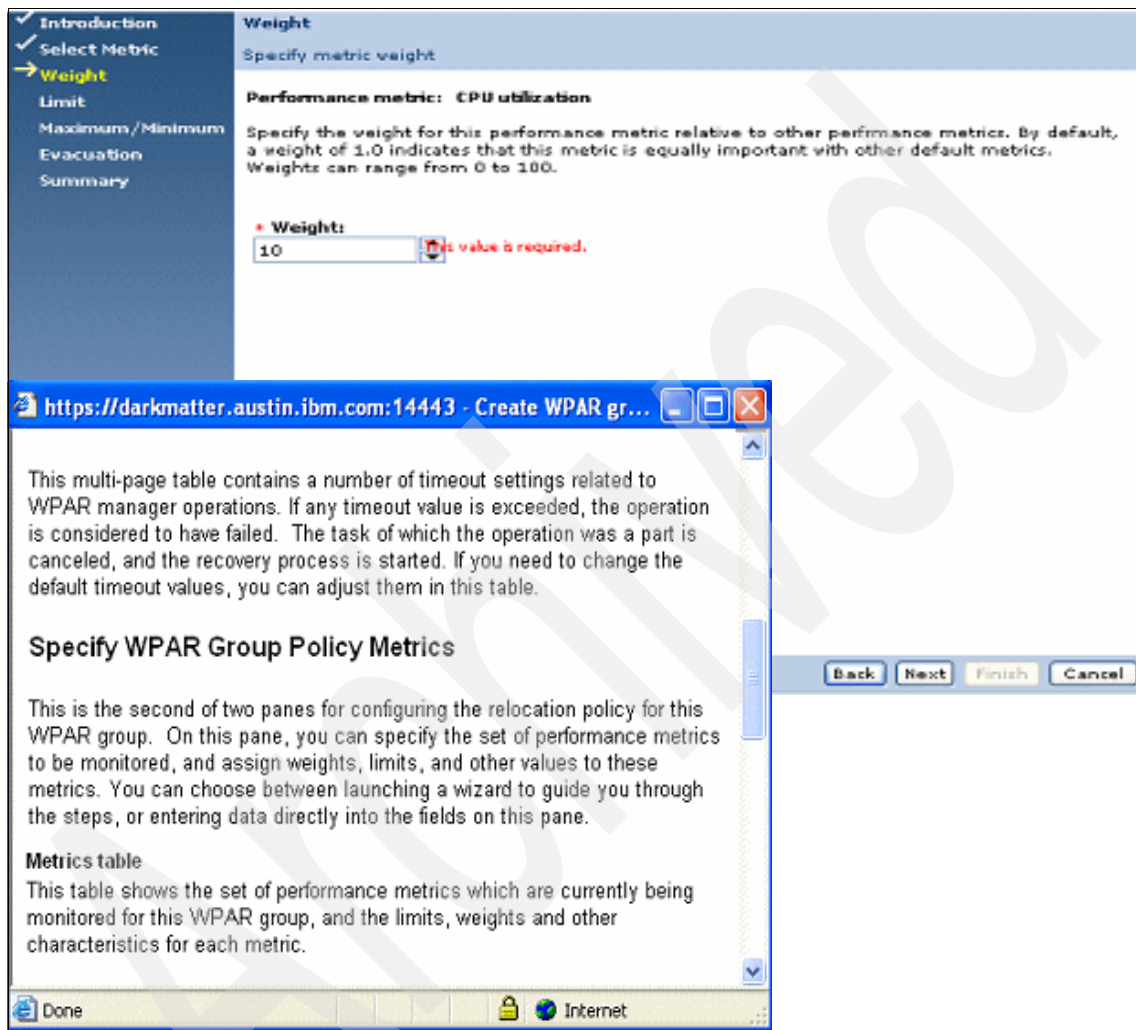


Figure 6-43 Help page within the metrics topic

8. For CPU utilization, Select the metric type and if necessary, the limit value in Figure 6-44. Select **Next**.

Limit

Specify metric type and limit

Performance metric: CPU utilization

Specify the metric type, and if necessary, the Limit value for this performance metric. If the Policy Metric is a rate value, then a Limit must be specified to derive a percentage of utilization to calculate a weighted average. If the metric is already a measure of utilization, the Limit value must be set to one (1).

Examples:

Process count is a rate value, so an explicit, accurate Limit must be specified. For example, if you specify a limit of 100, the number of processes is divided by 100 to calculate a weighted average.

Number of processes is a rate value, so an explicit, accurate Limit must be specified. If you specify a limit of 100, the number of processes is divided by 100 to calculate a weighted average.

This metric is a 'utilization' metric (e.g., CPU, Memory)

This metric is a 'rate' metric (e.g., process count, thread count) that requires a limit

+ Limit:

1

Back Next Finish Cancel

Figure 6-44 Performance metric for CPU utilization

9. Figure 6-45 shows the minimum and maximum threshold values for the CPU utilization performance metric. Select **Next**.

The screenshot shows a configuration window titled "Minimum and Maximum" with the subtitle "Specify metric minimum and maximum". On the left is a navigation pane with the following items: Introduction (checked), Select Metric (checked), Weight (checked), Limit (checked), Maximum/Minimum (highlighted with a yellow arrow), Evacuation, and Summary. The main content area is titled "Performance metric: CPU utilization" and contains the following text: "Specify the minimum and maximum threshold values for this performance metric. The minimum or maximum represent the allowed value for the metric before the Threshold State value is forced down or up to at least the value of this metric. This allows individual metrics such as CPU utilization to override the Threshold State calculation and present the individual Metric value for policy-based decisions." Below this text are two input fields: "Minimum:" with a value of 10 and "Maximum:" with a value of 50. At the bottom right of the window are four buttons: "Back", "Next", "Finish", and "Cancel".

Figure 6-45 Minimum and Maximum settings

10. The evacuation metric is set as seen in Figure 6-46. This is designed to help with predictive failures, such as a hard drive that is starting to fail and has a slow response time. This allows all WPARs in the group to be moved to another machine if the threshold limits are exceeded. This provides higher availability for applications. Select **Next**.

Introduction
✓ Select Metric
✓ Weight
✓ Limit
✓ Maximum/Minimum
→ Evacuation
Summary

Evacuation
Specify whether this is an evacuation metric

Performance metric: CPU utilization
Check this field to specify that when the Limit value for this metric is exceeded, all WPARs in this group will be evacuated, or moved off the managed system on which they are currently located. Use the Evacuation designation for predictive failure metrics, such as failed disk reads/writes, that indicate the possibility of imminent failure of the managed system.

This is an evacuation metric. This is an evacuation metric.

Back Next Finish Cancel

Figure 6-46 Evacuation metric enabled

11. Figure 6-47 shows the selections we have made in this walk-through of the wizard. Click **Finish**.

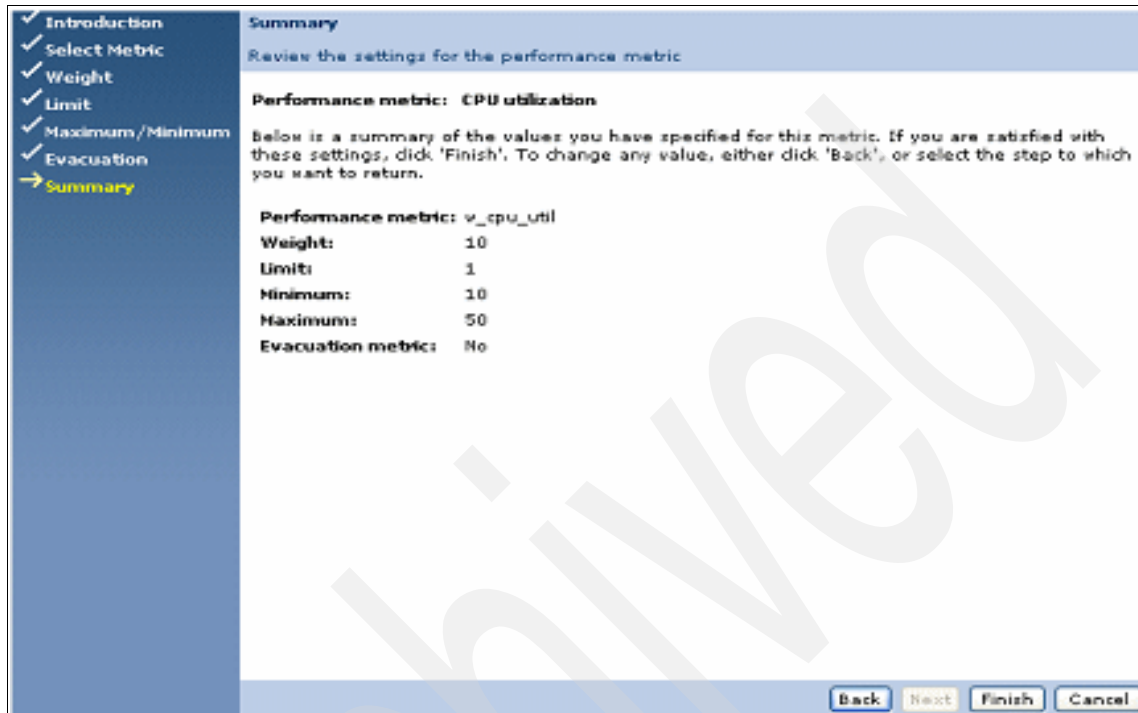


Figure 6-47 Summary of WPAR metric wizard selections

12. After the wizard is complete, it shows the new metric. You can add new metrics at any time by selecting the add function as seen in Figure 6-48.

Specify the performance metrics to be monitored for this WPAR group, and the policy settings associated with each metric. For step-by-step guidance, click [Launch Wizard](#). To add a metric without using the wizard, click [Add](#).

Add metrics using the Metrics Wizard: [Launch Wizard](#)

Add metrics manually:

Metrics in this WPAR Group:

Select	Metric	Weight	Limit	Minimum	Maximum	Evacuation
<input type="checkbox"/>	CPU utilization	10	1	10	50	—

Page 1 of 1 1 Total: 1 Shown: 1

Figure 6-48 Review the newly created CPU utilization metric and the capability to add more metrics

13. The next tab is Managed Server Profiles Associated with the WPAR Group. Figure 6-49 displays a list of server profiles that will be used for relocation of WPARs in this WPAR group. A *server profile* is a description of a server hardware configuration that might represent one or more physical servers in the environment. If two servers have identical hardware configurations, they share the same server profile.

When you select one or more server profiles to associate with this WPAR group, you are specifying that only servers with the selected server profiles will be considered as possible targets for WPAR relocation. This pane also allows you to assign ranks or priorities to server profiles. A managed system with a server profile that is ranked High will receive relocation preference over a managed system having a profile with a Medium or Low ranking. Click **OK**.

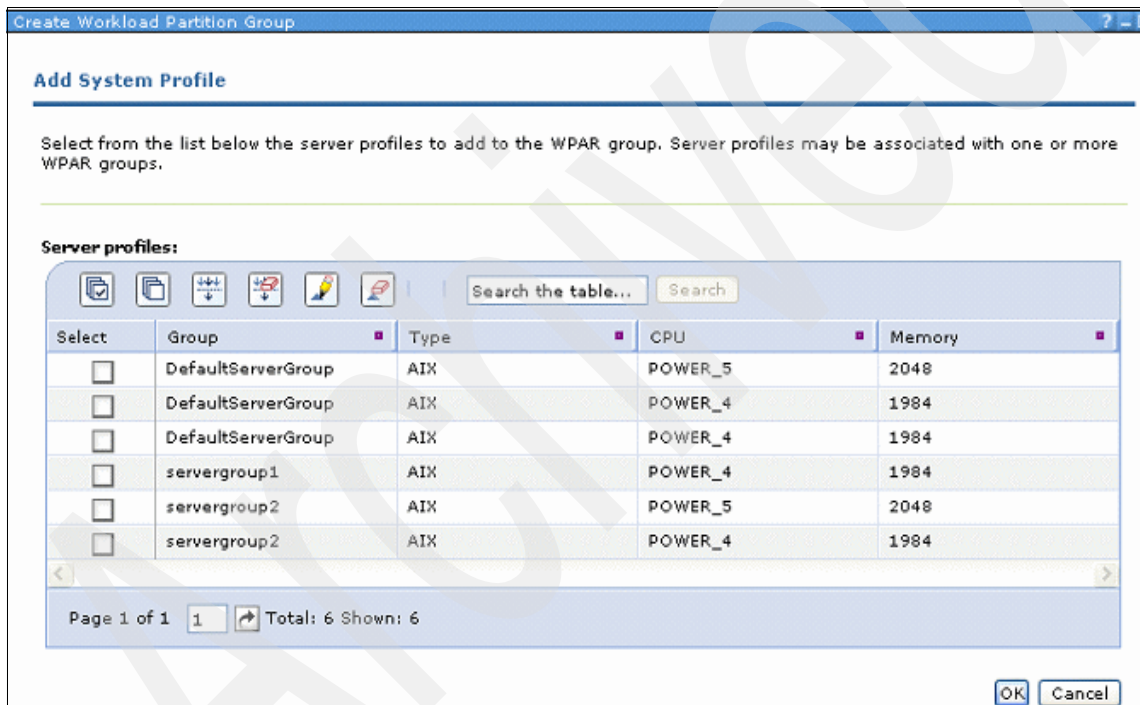


Figure 6-49 Add System Profile window

In addition to the required tests for WPAR compatibility, you can specify additional tests if needed to ensure safe relocation of your applications. As seen in Figure 6-50, if you know, for example, that the amount of memory assigned to the WPAR cannot be reduced, you can select **Arrival system must have at least as much physical memory as departure system** as an optional compatibility test.

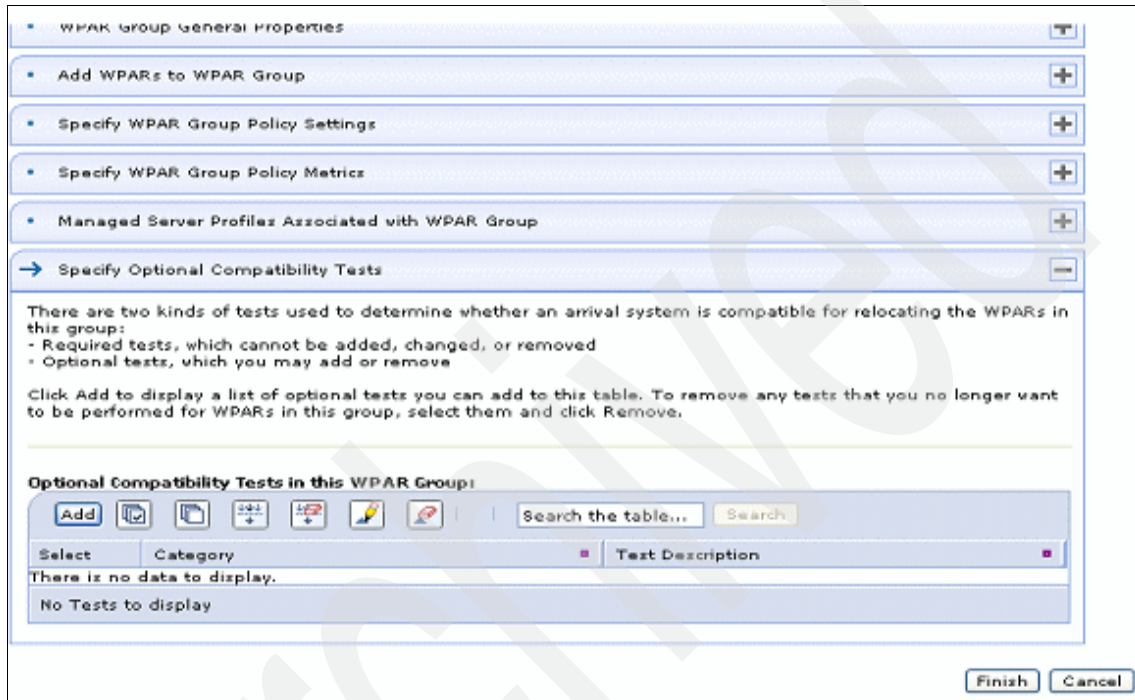


Figure 6-50 Optional Compatibility Tests in this WPAR Group

After you have selected one or more optional tests, they are evaluated in the same way as the required tests when determining whether a potential arrival system is compatible for WPAR relocation. You can review these selections as seen in Figure 6-51.

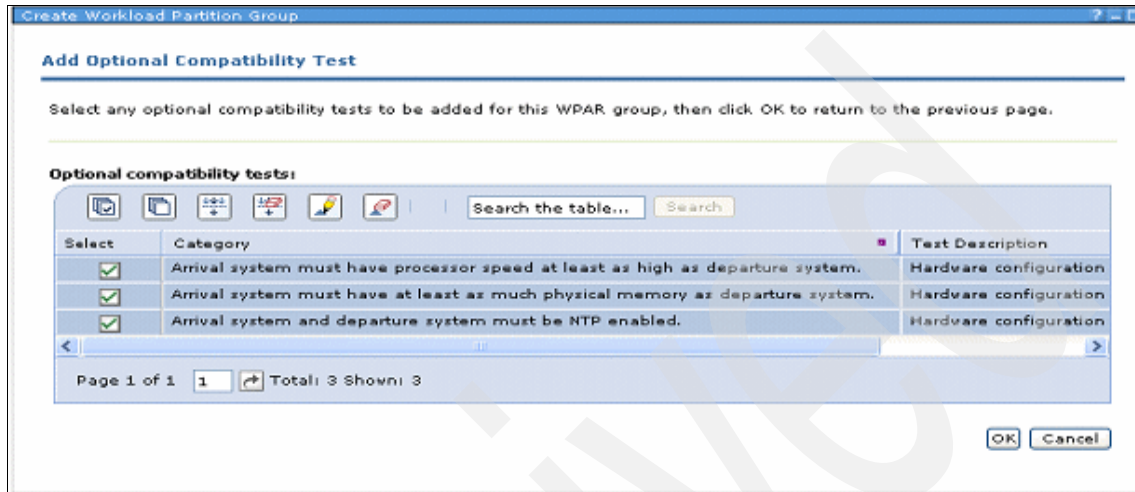


Figure 6-51 Review the optional compatibility tests

In Figure 6-52, the window reviews the groups that have been created and that will be managed. The jovian group has been created to handle the optional automatic relocation feature.

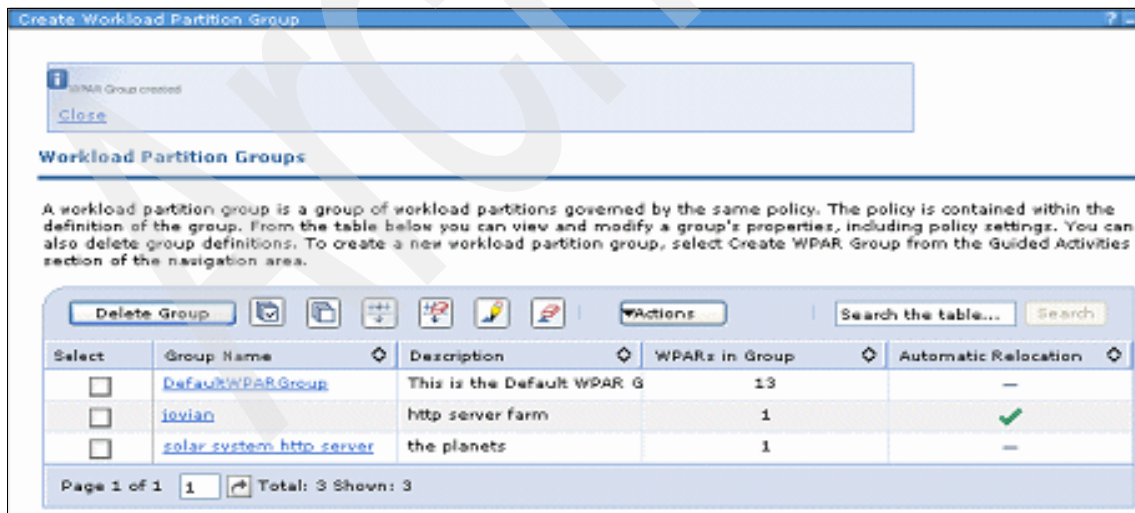


Figure 6-52 List of WPAR Groups

6.11.2 Resource Views

When returning to the main page, select **Resource Views** → **Managed Systems** in the menu on the upper left side of the page as seen in Figure 6-53. This shows the Global Environment systems known to the WPAR Manager for AIX Mobility Agent. The drop-down menu shows different tasks that can be managed from the UI.

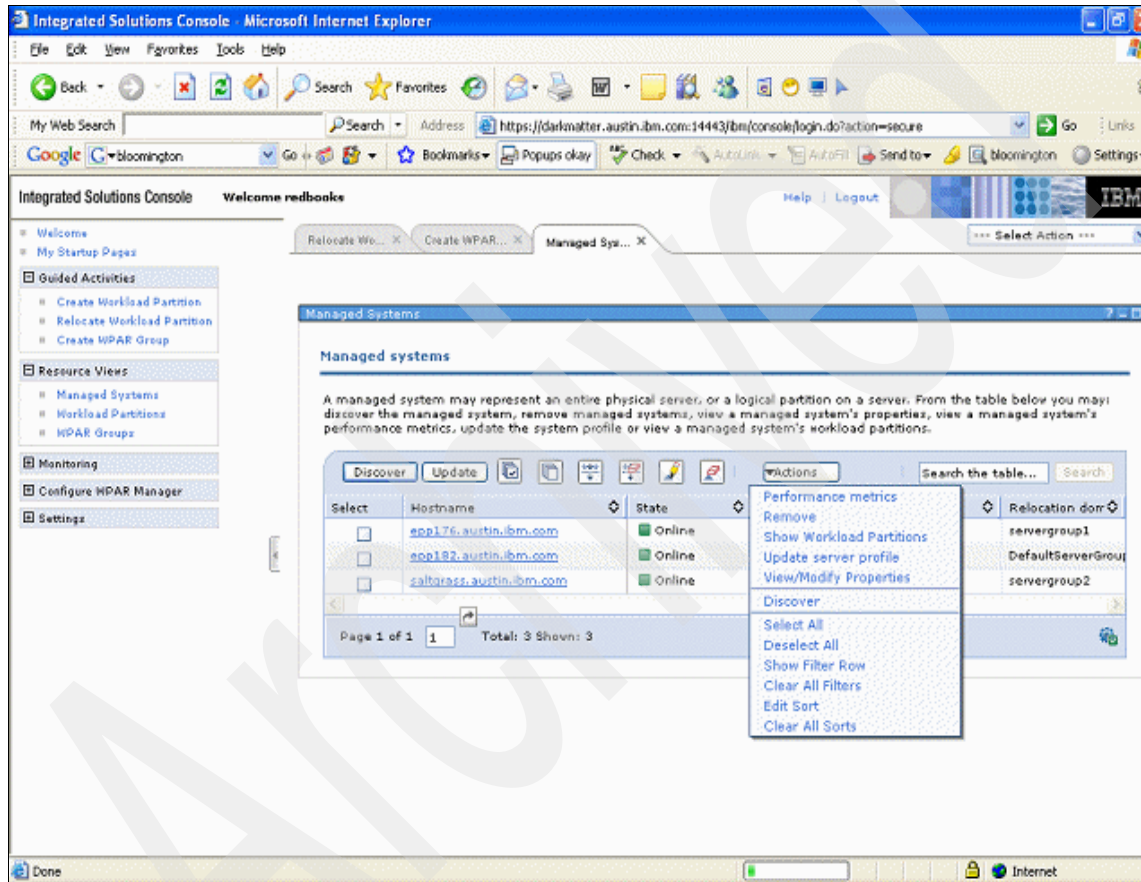


Figure 6-53 Resource Views → Managed Systems

6.11.3 Deploy application WPAR

To deploy an applications WPAR:

1. Select **Resource Views** → **Workload Partitions**. A window appears similar to Figure 6-54. This shows all the application and system WPARs and associated profiles that have been created. Some WPARs are active, others are not deployed. For this example, an application WPAR named yes is not deployed and the goal is to start the application by checking the box in front of yes and selecting **Actions** → **Deploy**.

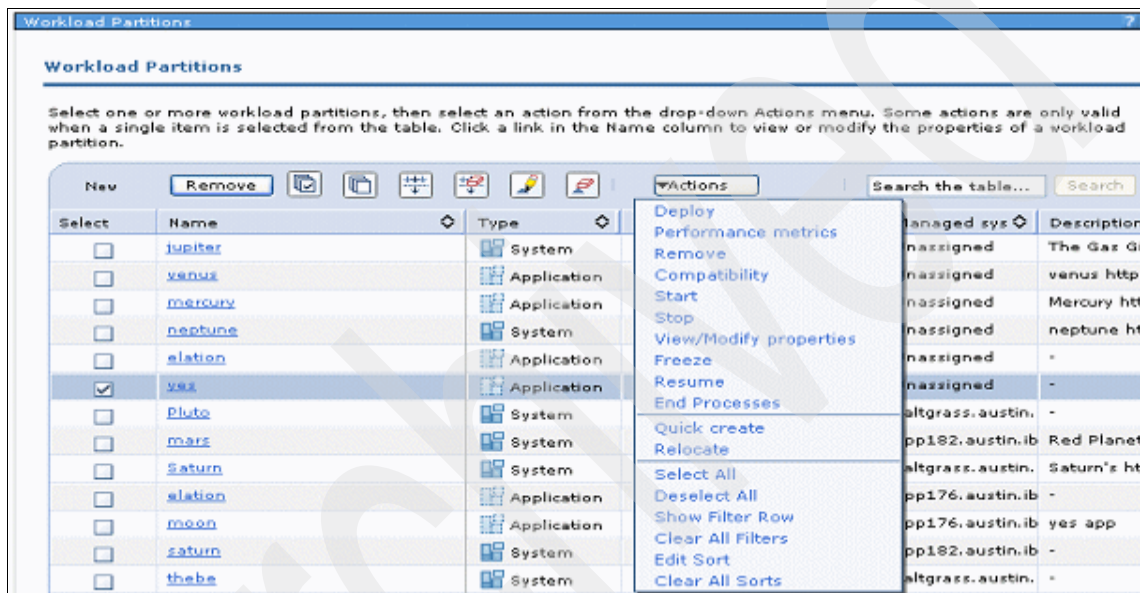


Figure 6-54 Workload Partitions window

- As seen in Figure 6-55, in the list box next to Systems, select a Global system. For our example, we selected the Global system, saltgrass. Click **OK**.

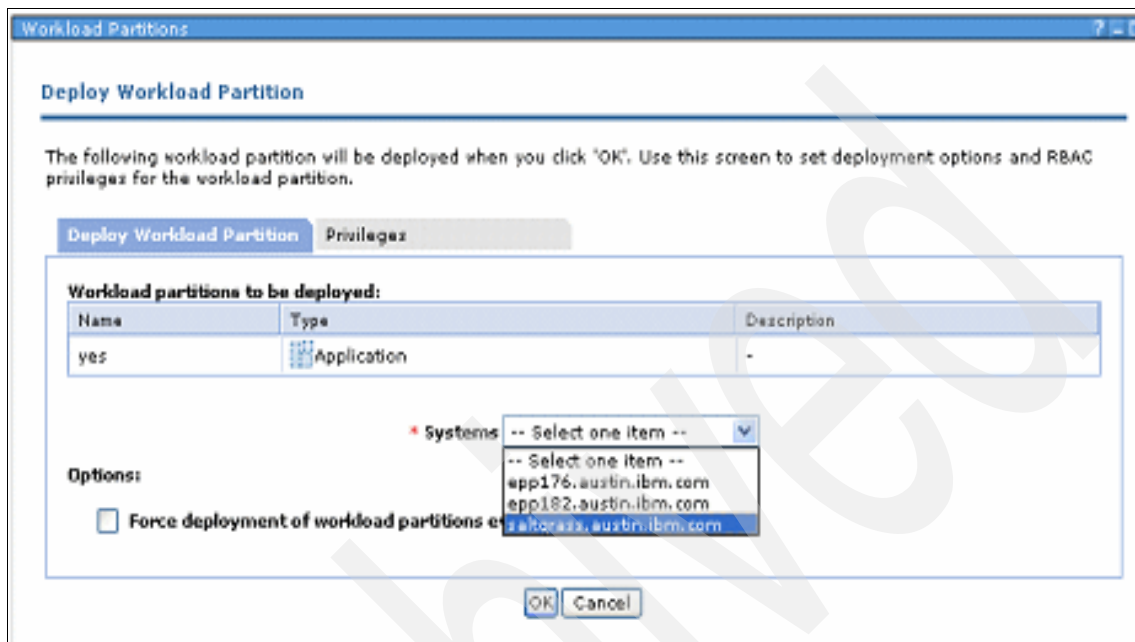


Figure 6-55 Select Global Deploy Workload Partition

Example 6-13 shows the output of the `lswpar` command on saltgrass. This shows that the application has been deployed and started. The user script was `/usr/bin/yes > /dev/null` to start the application WPAR and show that the CPU is busy.

Example 6-13 `lswpar` command on Global

```

root: saltgrass:/ --> lswpar
Name      State Type Hostname          Directory
-----
Pluto     D      S      elation.austin.ibm.com /wpars/elation
Saturn    T      S      adventure.austin.ibm.com /wpars/Saturn
earth     A      S      earth              /wpars/earth
elation   A      A      elation            /
thebe     A      S      thebe              /wpars/thebe
yes       A      A      yes                /
root: saltgrass:/
-->

```


- From the command line, we typed the **topas** command on the Global of saltgrass. Figure 6-56 shows new information in AIX V6.x for the total WPAR and the names of active WPARs, such as yes, earth, and thebe. Figure 6-56 shows the applications are running in the WPAR.

```

9.3.6.189 - PuTTY
Topas Monitor for host: saltgrass
Tue May 8 22:51:40 2007 Interval: 2

EVENTS/QUEUES FILE/TTY
Cswitch 552 Readch 869.0K
Syscall 18486 Writech 0.0G
Reads 1025 Rawin 0
Writes 7680 Ttyout 168
Forks 171 Igets 0
Execs 170 Namei 1027
Runqueue 2.5 Dirblk 0
Waitqueue 0.0

Kernel 46.5 |#####|
User 53.5 |#####|
Wait 0.0 |
Idle 0.0 |

Network KBPS I-Pack O-Pack KB-In KB-Out
eni 0.3 1.5 0.5 0.1 0.2
lo0 0.0 0.0 0.0 0.0 0.0

Disk Busy% KBPS TPS KB-Read KB-Writ
hdisk0 0.0 0.0 0.0 0.0 0.0

PAGING MEMORY
Faults 18335 Real,MB 1984
Steals 0 % Comp 41.7
PgspIn 0 % Noncomp 21.6
PgspOut 0 % Client 21.6
PageIn 0
PageOut 0
Sios 0 PAGING SPACE
Size,MB 512
% Used 1.4
% Free 98.5

NFS (calls/sec)
ServerV2 0
ClientV2 0 Press:
ServerV3 0 "h" for help
ClientV3 0 "q" to quit

Total WPARs 6
Active WPARs 4

WLH-Class (Active) CPU% Mem% Disk-I/O%
System 50 20 0
Shared 0 3 0

Name PID CPU% PgSp Class
yes 319636 49.7 0.1 yes
ksh 311380 16.4 0.5 System
java 291038 0.3 50.6 System
topas 499746 0.1 1.4 System
xmvlm 278560 0.1 3.2 thebe
xmvlm 454846 0.1 2.4 earth
wlmsched 53274 0.0 0.1 System
gil 49176 0.0 0.1 System
xmgc 36882 0.0 0.1 System
rpc.lock 184470 0.0 0.2 System

```

Figure 6-56 Topas showing new WPAR information

Figure 6-57 returns to the UI showing that the yes application WPAR is now active on the saltgrass server.

Workload Partitions

Select one or more workload partitions, then select an action from the drop-down Actions menu. Some actions are only valid when a single item is selected from the table. Click a link in the Name column to view or modify the properties of a workload partition.

Select	Name	Type	State	Mobile	Managed sys	Description
<input type="checkbox"/>	jupiter	System	Undeployed	✓	Unassigned	The Gas Giant
<input type="checkbox"/>	venus	Application	Undeployed	—	Unassigned	venus http s
<input type="checkbox"/>	mercury	Application	Undeployed	—	Unassigned	Mercury http
<input type="checkbox"/>	neptune	System	Undeployed	—	Unassigned	neptune htt
<input type="checkbox"/>	station	Application	Undeployed	✓	Unassigned	-
<input type="checkbox"/>	yes	Application	Active	—	saltgrass.austin.	-
<input type="checkbox"/>	Pluto	System	Defined	✓	saltgrass.austin.	-
<input type="checkbox"/>	mars	System	Active	—	epp182.austin.ib	Red Planet I
<input type="checkbox"/>	Saturn	System	Transitional	✓	saltgrass.austin.	Saturn's htt
<input type="checkbox"/>	station	Application	Active	✓	epp176.austin.ib	-
<input type="checkbox"/>	moon	Application	Active	—	epp176.austin.ib	yes app
<input type="checkbox"/>	saturn	System	Active	—	epp182.austin.ib	-
<input type="checkbox"/>	thebe	System	Active	—	saltgrass.austin.	-

Figure 6-57 The yes WPAR is now Active

Stopping the WPAR is the same process as starting. As seen in the Figure 6-58, select the check box of the yes WPAR and select the **Actions** → **Stop**.

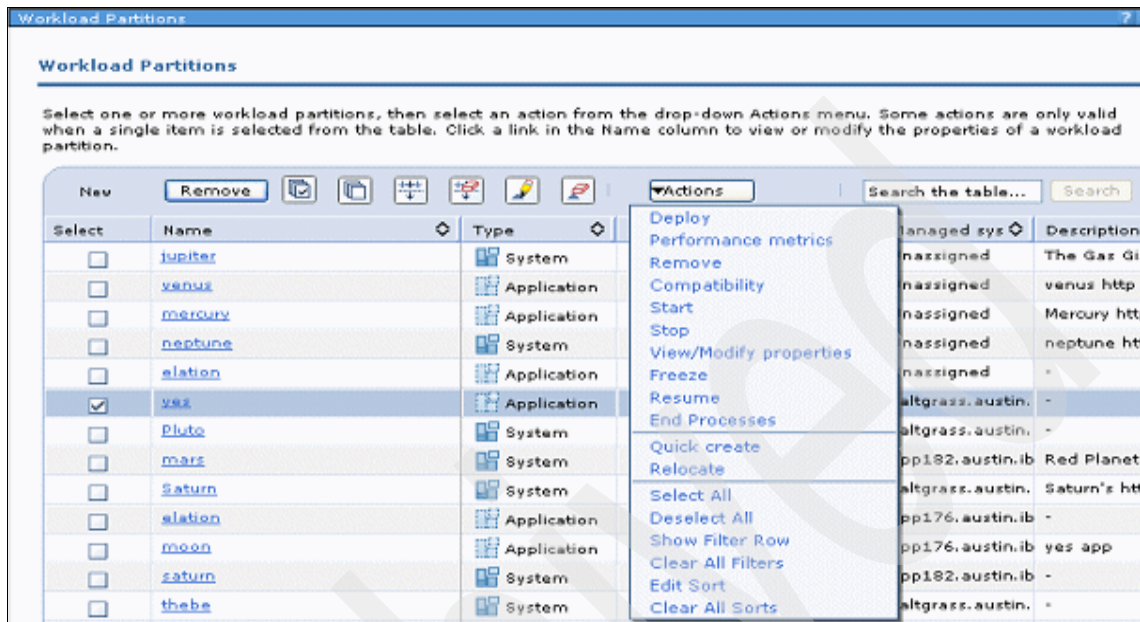


Figure 6-58 Stopping the WPAR

In Figure 6-59, the window shows the three ways to stop a WPAR. Click the desired method of stopping the WPAR and click **OK**.

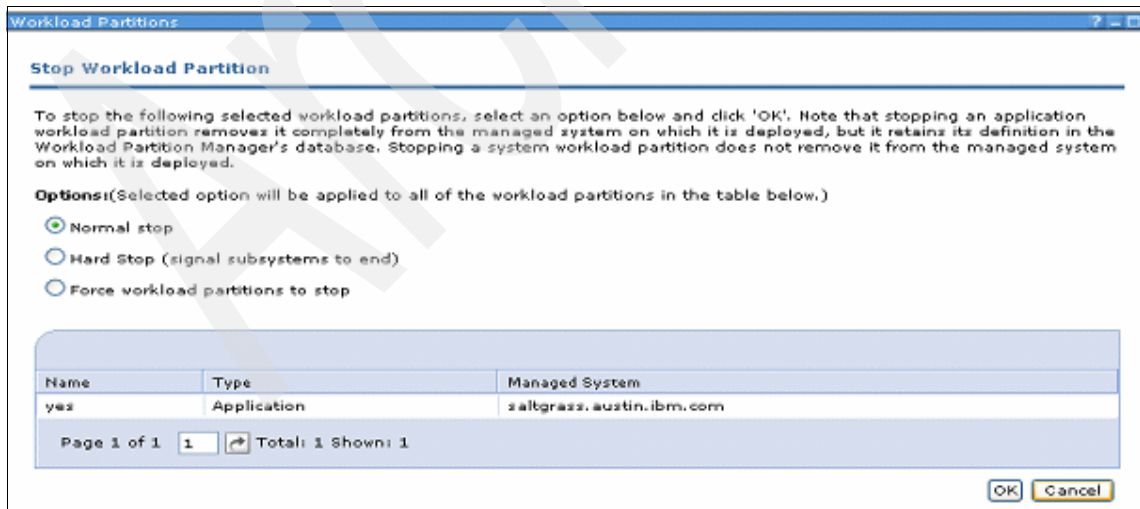


Figure 6-59 Stop Workload Partition window

Figure 6-60 shows that the yes application WPAR is now Undeployed, which means that the WPAR has stopped successfully.

The screenshot shows the 'Workload Partitions' management interface. At the top, a message box states: 'The stop wpar task initiated successfully.' Below this is a table with columns: Select, Name, Type, State, Mobile, Managed sys, and Description. The 'yes' WPAR is highlighted, and its 'State' is 'Undeployed'. The 'Mobile' column for 'yes' has a checked checkbox. The 'Managed sys' column for 'yes' is 'Unassigned'. The 'Description' column for 'yes' is '-'. Other WPARs listed include jupiter, venus, mercury, neptune, elation, Pluto, mars, Saturn, and elation.

Select	Name	Type	State	Mobile	Managed sys	Description
<input type="checkbox"/>	jupiter	System	Undeployed	✓	Unassigned	The Gas Gia
<input type="checkbox"/>	venus	Application	Undeployed	—	Unassigned	venus http s
<input type="checkbox"/>	mercury	Application	Undeployed	—	Unassigned	Mercury http
<input type="checkbox"/>	neptune	System	Undeployed	—	Unassigned	neptune htt
<input type="checkbox"/>	elation	Application	Undeployed	✓	Unassigned	-
<input type="checkbox"/>	yes	Application	Undeployed	—	Unassigned	-
<input type="checkbox"/>	Pluto	System	Defined	✓	saltgrass.austin.	-
<input type="checkbox"/>	mars	System	Active	Checked	epp182.austin.ib	Red Planet I
<input type="checkbox"/>	Saturn	System	Transitional	✓	saltgrass.austin.	Saturn's htt
<input type="checkbox"/>	elation	Application	Active	✓	epp176.austin.ib	-

Figure 6-60 Application WPAR yes is Undeployed

In Figure 6-61 on page 225, our goal is to remove the yes application WPAR. The check box in front of yes must be checked. Then, select **Remove** at the top of the box above the name.

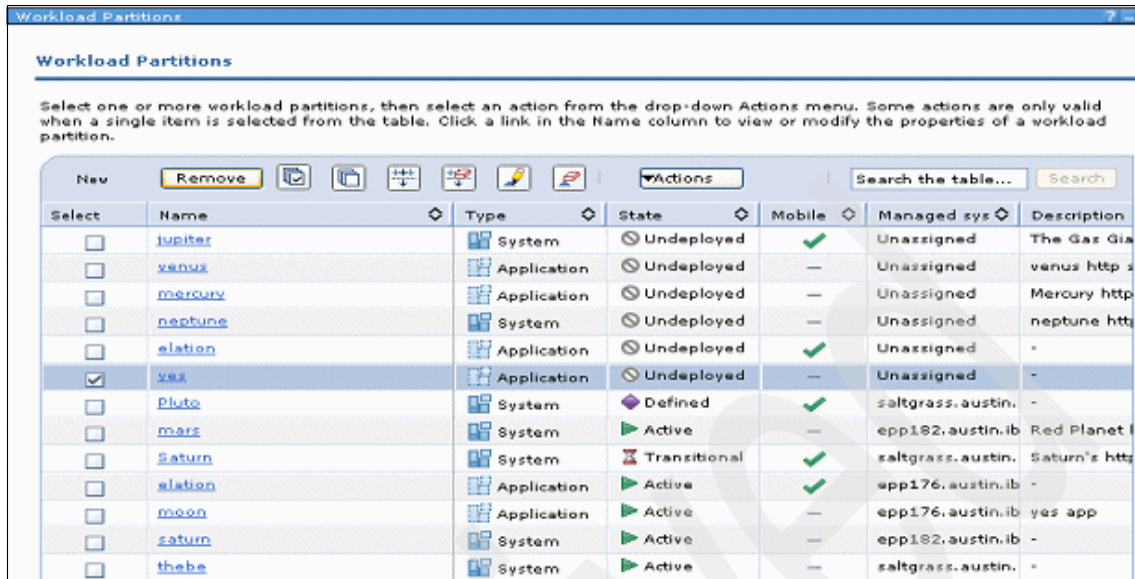


Figure 6-61 Remove a WPAR

Figure 6-62 shows you the WPAR to be removed. This is the system's way of asking you to verify that you are sure that you want to remove this WPAR. If you are sure, click **OK**.

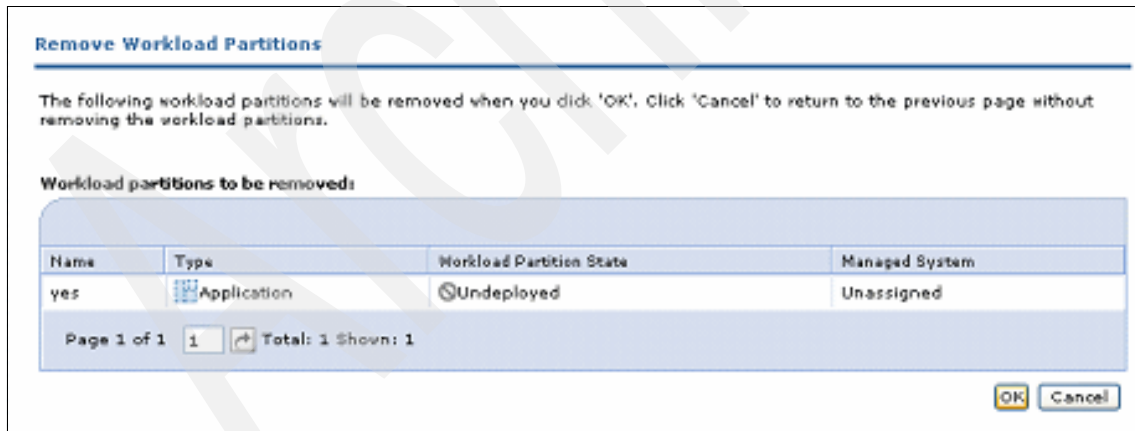


Figure 6-62 Workload Partitions to be removed

When looking at **topas** and the command line of **lswpar** in Figure 6-63, you can see that **yes** is no longer running. The Total WPARs is still 6, but the Active WPARs are now 3. This is different than what you saw in Figure 6-56 on page 221 where there were 6 Total WPARs and 4 Active. The **lswpar** shows all WPARs with their associated state. The **yes** WPAR is no longer there.

```

9.3.6.189 - PuTTY
Disk Busy% KBPS TPS KB-Read KB-Writ Faults 20310 Real,MB 1984
hdisk0 0.0 0.0 0.0 0.0 0.0 PgsIn 0 % Comp 41.6
PgsOut 0 % Noncomp 21.6
PgsIn 0 % Client 21.6
PgsOut 0
PageIn 0
PageOut 0
Sios 0
PAGING SPACE
Size,MB 512
% Used 1.4
% Free 98.5
Name PID CPU% PgSp Class NFS (calls/sec)
ksh 311380 20.5 0.4 System ServerV2 0
xnmwlm 258174 0.3 75.9 System ClientV2 0
java 291038 0.3 50.6 System ServerV3 0
topas 499746 0.1 1.4 System ClientV3 0
syncd 90206 0.1 0.5 System
wlmsched 53274 0.0 0.1 System
gil 49176 0.0 0.1 System
rpc.lock 184470 0.0 0.2 System
xntpd 82126 0.0 0.4 System
netm 45078 0.0 0.1 System
Total WPARs 6
Active WPARs 3
Signal 2 received

root: saltgrass:/ --> lswpar
Name State Type Hostname Directory
-----
Pluto D S elation.austin.ibm.com /wpars/elation
Saturn T S adventure.austin.ibm.com /wpars/Saturn
earth A S earth /wpars/earth
elation B A elation /
thebe A S thebe /wpars/thebe
root: saltgrass:/ -->

```

Figure 6-63 Topas and lswpar from saltgrass

6.11.4 Task Activity

To check the log of activities on the main page, select **Monitoring** on the upper left side and then select **Task Activity**. This reveals Figure 6-64. This area is useful for debugging a WPAR that does not start or stop properly.

Description	Status	Managed system	WPAR	Start time
Stop Workload Partition	Succeeded	saltgrass.austin.ibm.cc	yes	5/8/07 10:54:29 PM
Deploy Workload Partition	Succeeded	saltgrass.austin.ibm.cc	yes	5/8/07 10:51:40 PM
Register discovered Workload Partition	Succeeded	saltgrass.austin.ibm.cc	elation	5/8/07 10:28:21 PM
Update Workload Partition Configuration	Succeeded	epp176.austin.ibm.cor	elation	5/8/07 10:27:27 PM
Relocate Workload Partition	Failed	saltgrass.austin.ibm.cc	elation	5/8/07 10:24:13 PM
Deploy Workload Partition	Succeeded	saltgrass.austin.ibm.cc	elation	5/8/07 10:18:55 PM
Stop Workload Partition	Succeeded	saltgrass.austin.ibm.cc	venus	5/8/07 10:11:43 PM
Register discovered Workload Partition	Succeeded	saltgrass.austin.ibm.cc	earth	5/8/07 5:47:49 PM
Register discovered Workload Partition	Succeeded	saltgrass.austin.ibm.cc	thebe	5/8/07 2:55:55 PM
Register discovered Workload Partition	Succeeded	epp182.austin.ibm.cor	saturn	5/8/07 1:45:19 PM

Figure 6-64 Application Tasks log

6.11.5 Application configuration

To view or change settings related to the Workload Partition Manager application, go to the main page, select **Configure WPAR Manager** on the upper left side of the main page and then select **Application Configuration**. This presents a window as seen in Figure 6-65 on page 228.

Relocating a WPAR generates a set of files that includes logs and the state file that is used to save the full context of the applications running within the WPAR. The state file root directory is used to indicate where these files are going to be generated during the relocation workflows. For system WPARs, this directory refers to a location relative to the root directory(/) of the WPAR itself. For application WPARs, this directory refers to a location relative to the root directory of the managed system that is hosting the WPAR.

The Metric collection interval in seconds field is required. This setting determines the interval during which performance metrics are collected. The default collection interval is every 60 seconds. The minimum allowed interval is every 30 seconds.

Choosing a shorter interval allows you to view changes in resource usage over shorter periods of time, but increases overhead because more frequent processing of the metrics is required. Choosing a longer interval reduces server overhead, but it might not provide the resolution needed to determine whether WPAR relocations are warranted.

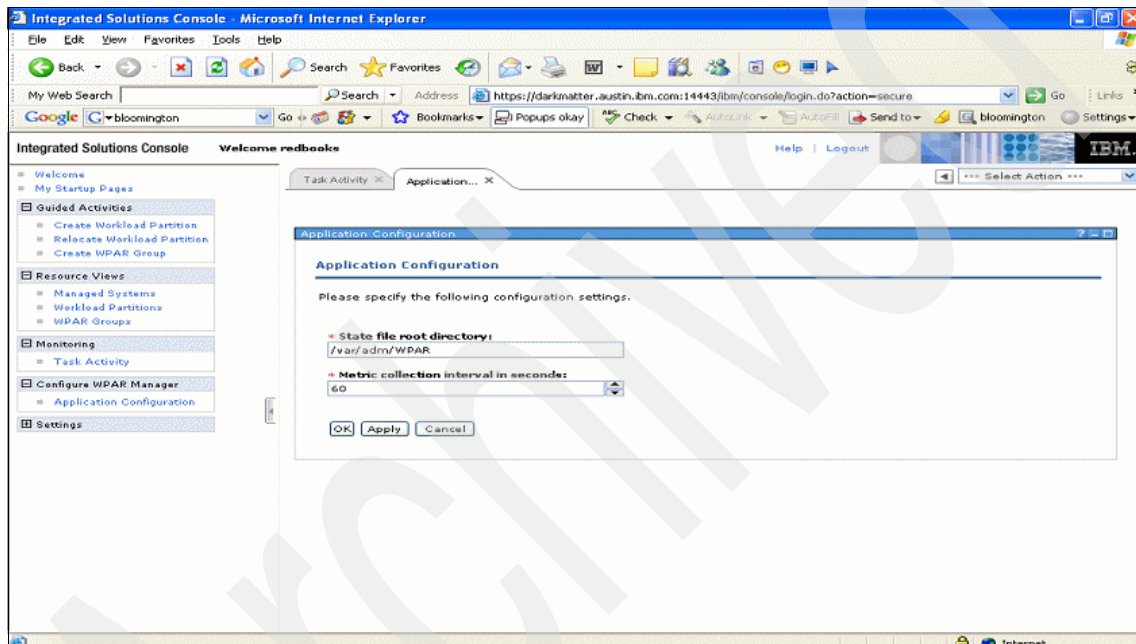


Figure 6-65 Application Configuration

6.11.6 Logging and tracing

To view or change settings related to Console Logging and Tracing, go to the main page and select **Settings** on the upper left side and then **Console Logging and Tracing**. This presents a window as seen in Figure 6-66 on page 230.

This configuration is used to control the settings for recording and maintaining runtime information for the console. This window has two sections for configuring console logging and tracing.

The General Properties section is used to configure log and trace settings for the overall console.

The Detail Levels section is used to configure trace settings for specific packages.

In the General Properties section, you can change several options. “Append log file” appends log messages to a single log file. If you do not select this option, a new log file is created each time that you start Integrated Solutions Console. Logging messages are written to the file `error-log-0.xml` of the `/logs` subdirectory of the console installation. This file is always locked by the console to write log messages.

There is an “Append trace file” option, which traces messages to a single log file. If you do not select this option, a new trace file is created each time that you start Integrated Solutions Console. Trace messages are written to the file `trace-log-0.xml` of the `/logs` subdirectory of the console installation. This file is always locked by the console to write trace messages.

The following fields are available for more detailed configuration of the log and trace output:

- ▶ Maximum file size indicates the maximum size of the log or trace file in bytes. This value must be an integer between 100000 and 4000000. Do not include any special characters in this value, such as commas or periods.
- ▶ Maximum number of historical files specifies the number of log or trace files to keep after a new file is started. This value must be an integer of at least 1 and no more than 6. Historical files are stored in the `/logs` subdirectory with the following name format: The log file format is `error-log-number.xml` and the format for the trace file is `trace-log-number.xml`.
- ▶ The lowest number indicates the most recent log file. For example, if you set the value of this field to 3, then `trace-log-3.xml` is the oldest available trace file. The next time the server is started, this file is discarded and `trace-log-2.xml` is renamed to `trace-log-3.xml`.

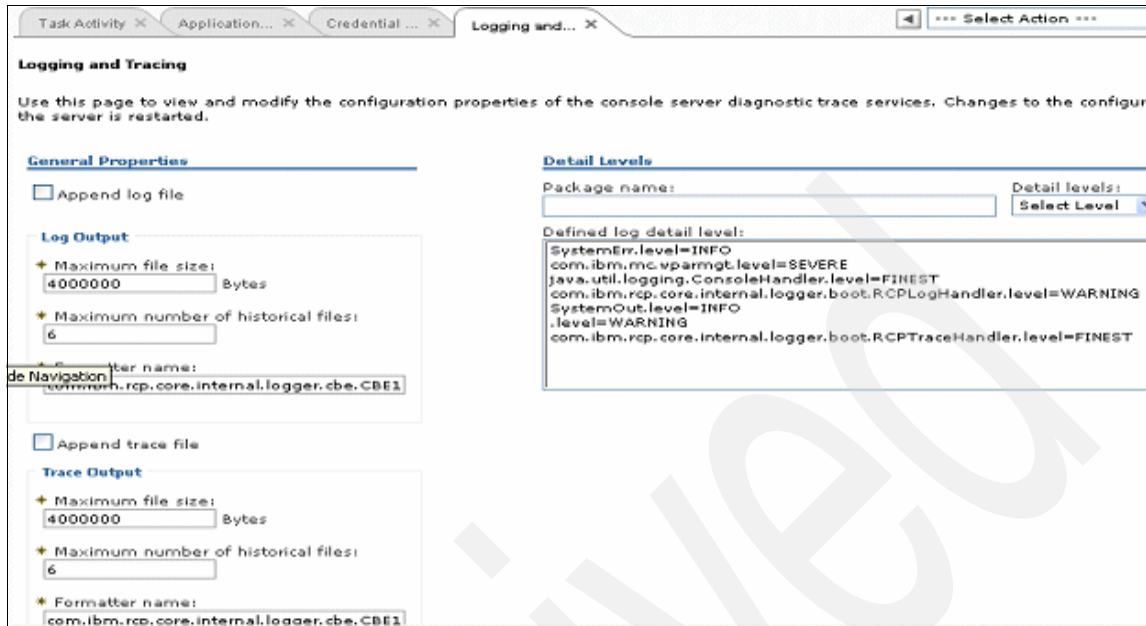


Figure 6-66 Console Logging and Tracing configuration

6.12 Introduction to mobility command-line interface

Relocating a workload partition requires performing actions within two LPARs:

- ▶ The source partition where the WPAR is executing, on which a snapshot of the code state and its runtime environment has to be saved
- ▶ The target partition, where the snapshot and environment must be restored, so that the application code can resume execution

In addition, you might have to perform other actions, such as preparing the target global environment of the target system by defining file systems or checking for relocation prerequisites, such as operating system code compatibility. The manual relocation of an application is a task that requires the system administrator to perfectly understand the application and all the steps involved in its migrations.

Important: The WPAR Manager automates the relocation of WPARs and encapsulates within one operation the whole sequence of required tests, verifications, checkpoints, and pause and restart operations that are performed in the correct order for a successful relocation.

The WPAR Manager performs compatibility tests between the source and target global environment that are not provided by the command-line interface (CLI) commands.

Using the WPAR Manager is therefore the only recommended way to perform a WPAR relocation.

We expect that partition mobility tasks using the command-line interface will take place infrequently in normal production environments, and we recommend that system administrators performing CLI mobility tasks have a solid understanding of the WPAR partition mobility operation and its potential effects on the running application in the WPAR. There can be many conditions that must be met for successful migration, which might require detailed knowledge of any dependencies in the application that is being migrated.

6.13 Explanation of mobility CLI commands

The mobility feature CLI consists of four commands that complement the base AIX WPAR CLI to allow the system administrator to stop a WPAR at a point in time, capture its state, store it in a set of files, and restart execution of the workload partition:

chkptwpar

The **chkptwpar** command captures a snapshot of all tasks executing within one WPAR. It first interrupts all processes so they reach a *quiescence point*, then it stores a copy of the processes' context in a state file. Depending on the optional argument passed to the command, the WPAR then either:

- Resumes its execution within the same LPAR, returning to the Active state
- Goes to the Paused or Frozen state
- Terminates

killwpar

The **killwpar** command is used to kill all tasks belonging to a *Paused* WPAR. It cleans up the global environment from all processes and related information regarding this WPAR. When a WPAR is migrated from one source LPAR

to a target LPAR, it might be left in the Paused state on the source LPAR. After it is successfully restarted on the target LPAR, the Paused WPAR can be removed from the source LPAR by using the **killwpar** command.

restartwpar

The **restartwpar** command *creates* a workload partition from a state file previously generated using a **chkptwpar** command (For a system WPAR, the partition must have been defined before it can be restarted). The *creation* of the WPAR consists of restoring all tasks that make the WPAR in the LPAR memory, so that it is ready to resume execution of the processes at the exact point where they were saved during the checkpoint. After the WPAR is created, the **restartwpar** command either puts the WPAR in the Paused state if called with the **-p** option or, by default, resumes execution of the application and puts it in the Active state.

resumewpar

The **resumewpar** command resumes execution of a *Paused* or *Frozen* WPAR and puts it into the *Active* state.

6.14 Comparison of CLI commands

The WPAR command-line interface provides the system administrator with many commands. Several are part of the base AIX support, and others are part of the WPAR Manager support. Although the names of the commands are chosen to be as self-explanatory as possible, it is worthwhile mentioning the differences between several of these commands.

startwpar and restartwpar

The **startwpar** command is part of AIX, while **restartwpar** is a WPAR Manager command.

The **startwpar** command only applies to a system WPAR. It can be seen as booting a system partition from scratch. It starts the init process, which in turn starts all daemons and processes of the application environments.

The **restartwpar** command restores an application in the middle of a previous execution that was captured at a point in time with a checkpoint action.

rebootwpar and resumewpar

The **rebootwpar** command is part of AIX, while **resumewpar** is a WPAR Manager command.

The **rebootwpar** command stops all processes running within a workload partition. For a system WPAR, it is then equivalent to a **startwpar** command, launching the init process and starting all daemons and processes from scratch. All environmental connections from the previous execution of the WPAR are lost.

The **resumewpar** command applies to a partition that has been *Paused* or *Frozen*. It resumes execution of all WPAR processes at the state where they were checkpointed.

stopwpar, killwpar, and rmwpar

The **stopwpar** and **rmwpar** commands are part of AIX, while **killwpar** is a WPAR Manager command.

The **stopwpar** and **killwpar** commands stop execution of a workload partition, while **rmwpar** deletes all of a workload partition's definitions.

The normal way to end execution of a workload partition is to use the **stopwpar** command, which:

- ▶ Ends all processes executing within the partition
- ▶ Unloads WLM classes
- ▶ Deactivates the partition IP addresses
- ▶ Unmounts file systems

The **killwpar** command only works on a *paused* WPAR and is the normal way to stop execution of a partition that has reached the *Paused* state.

The **rmwpar** command deletes all definitions of the WPAR in the workload partition database, as well as the WLM profile of the partition, and it deletes the local file systems belonging to this partition.

6.15 Step-by-step relocation of a workload partition

This section presents you with an example of a WPAR relocation between two LPARs to help you understand the successive actions that are performed during a relocation. As stated in Section 6.12, "Introduction to mobility command-line interface" on page 230, do not use this example for relocation of WPARs in normal circumstances.

Our example is the relocation of a system WPAR:

1. First, create the WPAR to support mobility. It must be checkpointable, therefore, create it with the `-c` option of the `mkwpar` command. All writable file systems must be NFS-mounted.

Important: As a best practice, we recommend that all file systems used in the WPAR are mounted over NFS. Even if the file system is read only, it must still be shared between the departure and arrival nodes in a relocation scenario. The file systems need to be identical on the source and target nodes, regardless of their read/write attributes. For example, a process on a source node can have a file opened on a read-only file system. But if the same file is not present on the target node, the relocation will fail because the file descriptor cannot be reopened on the target node.

2. Now start the WPAR and its hosted applications.
3. Create a specification file from the executing WPAR:

```
mkwpar -e MobileWPAR -o MobileWPAR.spec -w
```

The `-w` option indicates that a specification file must be created for the partition identified by the `-e` option and stored in the file specified by the `-o` option.

Example 6-14 shows the NFS-mounted file system's definition extracted from the workload partition specification file. It also shows the `/proc` file system exception.

Example 6-14 Example of NFS-mounted file system stanza in WPAR specification file

```
mount:
    dev = "/big/wpars/MobileWPAR/home"
    directory = "/home"
    vfs = "nfs"
    mountopts = "bg,intr"
    host = "gracyfarms.itsc.austin.ibm.com"

mount:
    dev = "/proc"
    directory = "/proc"
    vfs = "namefs"
    mountopts = "rw"
```

4. Create an empty directory to store the state file that will contain an image of the WPAR state during the relocation. This empty directory must be in an NFS file system that is accessible from both the source and target LPARs that host

the WPAR. Furthermore, this directory must be writable from within the WPAR. For example, you can create this empty directory in the /etc directory of the WPAR. The path to this file system must be given as an absolute path name of the global environment.

Note: If you specify a state file directory that does not exist yet when using the `chkptwpar` command, `chkptwpar` will create it. If the directory already exists when running `chkptwpar`, it must be empty.

Tip: In most cases, a file system of a size equal to two times the memory space used by the WPAR is sufficient. In the case of the WPAR opening a large number of files, the state file will require more disk space.

5. Define the partition on the target system using the specification file created on the source system:

```
mkwpar -p -f MobileWPAR.spec
```

Make sure you do *not* use the `-s` option so that the WPAR remains in the *Defined* state. *Do* use the `-p` flag to avoid recreating the file systems.

6. Check that the NFS file systems holding the WPAR data are correctly exported to the target global environment. You do not need to mount them, because the WPAR restart action will mount all files based on the information archived in the specification files used to define the WPAR.

Tip: The NFS file systems must be exported to and mountable from the source and target global environment, as well as the mobile WPAR. In other words, the /etc/exports stanza in the NFS server for each file system must contain the IP address or host name of the source, global environment, the target global environment, and the WPAR.

7. Make sure the target global environment is compatible with the source global environment. This includes at least compatible processor architecture, identical levels of AIX, including patches, the same level of the same edition of installed software, and access to the same network VLANs.
8. Capture a snapshot of the WPAR partition state by checkpointing it:

```
chkptwpar -k -d /wpars/MobileWPAR/tmp/statefiledir -o  
/wpars/MobileWPAR/tmp/mobileWPARLOG MobileWPAR
```

In this example, the WPAR is killed (`-k` option) after the checkpoint and goes to the defined state. Depending on the application architecture, other checkpoint options can be used to perform the checkpoint in several stages and freeze or pause the application before capturing the state file.

Tip: It is a good practice to always specify the path to the logfile. This file must be on an NFS directory that is accessible from the source and target directory, as well as from within the WPAR.

9. Check the completion of the checkpoint action. The **chkptwpar** command needs to return a 0 exit value. The **lswpar** command must show the WPAR in the *Defined* state.
10. On the target system, restart the execution of the WPAR, specifying both the statefile and logfile as defined in the checkpoint command:

```
restartwpar -d /wpars/MobileWPAR/tmp/statefiledir -o  
/wpars/MobileWPAR/tmp/mobileWPARLOG MobileWPAR
```
11. Use the **lswpar** command on the target system to check that the WPAR is in the *Active* state and verify that the applications have resumed executing.

In addition to this simple example, the WPAR mobility commands offer options to let the system administrator define log files or execute a user-defined script before the restart of an application WPAR is complete. Execution of a user-defined script is a valid option only for application WPARs in restart-pause transitions.

The **restartwpar** command does not return after the restart of the application WPAR is finished, but when the restart is finished *and* the restarted application has finished its normal course of execution. So, the caller of the **restartwpar** command for an application WPAR has no way of knowing exactly when the application is about to resume its execution. The **lswpar** command shows the wpar is in the state "T" while it is restarted and "A" when the application has resumed. When the caller of **restartwpar** passes a script through the **-u** option, the script is invoked after the successful reconstruction of the wpar when it is in the *Paused* state.

Resource control

This chapter presents methods for WPAR resource control and how those methods can be used to properly allocate CPU and memory resources to all workload partitions, thereby minimizing interference between workload partitions in terms of resource utilization and performance.

The workload partition resource control is based on the *Workload Manager* (WLM) technology built into the AIX kernel since AIX 4.3.3. Because the workload partition resource control commands encapsulate and hide WLM details, the system administrator does not need to have an in-depth knowledge of WLM in order to use workload partition resource control. This chapter contains the following sections:

- ▶ 7.1, “Workload partition resource control overview” on page 238
- ▶ 7.2, “Workload partition resource control attributes” on page 240
- ▶ 7.3, “Default values of resource control attributes” on page 241
- ▶ 7.4, “Practical examples of resource allocation” on page 242
- ▶ 7.5, “Use case scenario” on page 249
- ▶ 7.6, “WPAR resource control changes to WLM” on page 249
- ▶ 7.7, “Questions about WLM and WPAR resource control” on page 251

7.1 Workload partition resource control overview

In the context of workload partition, *resource control* can be used to specify:

- ▶ The amount of CPU and memory resources allocated to a workload partition
- ▶ The number of processes and threads that are allowed in a workload partition
- ▶ The amount of virtual memory that a single process within a workload partition can consume
- ▶ Which *resource set* (rset) to use

Note: The control of resource allocation to each WPAR is performed at the global environment level by the system administrator; therefore, resource control commands are not available from within a workload partition.

We can specify the resource control with the **-R** option of the **mkwpar**, **chwpar**, **wparexec**, and **lswpar** commands.

7.1.1 CPU and memory allocation to a WPAR

There are two approaches to specify CPU and memory allocation:

- ▶ **Share-based:** Each workload partition receives its share of resource according to the ratio of its share to the total shares of all currently active workload partitions.

Notice that the amount of resources received by a WPAR in this approach can vary significantly depending not only on the number of shares it was allocated, but also on how many other workload partitions are currently active in the system and how many shares they have.

- ▶ **Percentage-based:** Three values can be specified in this approach:
 - **Minimum%:** The minimum amount of resource that a workload partition is guaranteed to have available. If it consumes less than this amount, WLM will raise its priority so that it can use more resources.
The total amount of minimum% for all currently active workload partitions cannot exceed 100%.
 - **Soft maximum%:** The maximum amount of resource that a workload partition can have when there is contention for the resource. If there is no contention, this maximum limit can be surpassed.
 - **Hard maximum%:** The maximum amount of resource that the workload partition can have, even if there is no contention for the resource. The

workload partition will never get more resource than its hard maximum, even if it is the only workload partition running in the global environment.

Important: Note these rules:

- ▶ The total minimum% of all active workload partitions must not exceed 100%.
- ▶ The minimum% must not exceed the soft maximum%.
- ▶ The soft maximum% must not exceed the hard maximum%.

7.1.2 Processes and threads in a WPAR

The number of processes, as well as the number of threads, can be controlled in the workload partition. For an example, see Section 7.4.3, “Processes and threads” on page 247.

7.1.3 Virtual memory of a single process within a WPAR

The maximum amount of virtual memory that a process can have in a workload partition can be controlled. For an example, see Section 7.4.4, “Process virtual memory” on page 248.

7.1.4 Resource sets

A *resource set* is used to define a subset of processors in the system. If a resource set is specified for a workload partition, it can use the processors within the specified resource set only.

Note: A resource set is normally used by compute intensive or High Performance Computing (HPC) applications in order to achieve processor affinity.

7.2 Workload partition resource control attributes

Table 7-1 lists the attributes that can be set with the **-R** option.

Table 7-1 Attributes for **-R** option

Attribute=value	Description
active={yes no}	Use active=yes to activate resource control for a workload partition. Using active=no only saves the other specified attributes on the command, but will not activate the resource control.
rset=<rset>	Configure workload partition to use only resource set <rset>. The default is to use all CPUs in the system.
shares_CPU=<n>	Specify <n> as the number of CPU shares for this workload partition. Shares value can be from 1 to 65535.
CPU=<min>%-<sMax>%,<hMax>%	Specify <min>% as the minimum CPU%, <sMax>% as the soft maximum CPU%, and <hMax>% as the hard maximum CPU% for this workload partition.
shares_memory=<n>	Specify <n> as the number of memory shares for this workload partition. Shares value can be from 1 to 65535.
memory=<min>%-<sMax>%,<hMax>%	Specify <min>% as the minimum memory%, <sMax>% as the soft maximum memory%, and <hMax>% as the hard maximum memory% for this workload partition.
totalProcesses=<n>	Specify <n> as the total number of processes allowed in this workload partition.
totalThreads=<n>	Specify <n> as the total number of threads allowed in this workload partition. The total thread limit must be at least as large as the total process limit.
procVirtMem=<n>[MB GB TB]	Specify <n> megabytes, gigabytes, or terabytes as the maximum amount of virtual memory that a single process in this workload partition can consume.

7.3 Default values of resource control attributes

Because workload partition resource control makes use of Workload Manager (WLM) technology, the default values for its resource control attributes are inherited from WLM.

However, some of these default values might need to be adjusted according to the business decision that led to using workload partitions. For example, if CPU intensive applications and I/O intensive applications are consolidated within the same LPAR in different WPARs, it might be necessary to set a minimum CPU percentage for the I/O applications so that they can have a guaranteed amount of computing power even during the peak activities of the CPU intensive applications.

Table 7-2 lists the default value for each attribute and the recommended setting for them.

Table 7-2 Default value and recommendation

Attribute	Default value	Recommendation
active=	yes	Use default value for <i>all</i> workload partitions (because it will be difficult for WLM to effectively control the resources if there is a workload partition that is not under its control).
rset=	none	Use default value of no resource set (unless your workload is comprised of compute intensive, HPC types of applications).
shares_CPU=	unlimited	Explicitly set for <i>all</i> workload partitions according to the relative importance of your workload partition (the more share, the more CPU resource it gets).
CPU=	0%-100%,100%	<ul style="list-style-type: none"> ▶ Specify some minimum value for your important workload partition so that it receives CPU resource under contention. ▶ Use the default value for soft/hard maximum.
shares_memory=	unlimited	Explicitly set for <i>all</i> workload partitions according to the relative importance of your workload partition (the more share, the more memory resource it gets).

Attribute	Default value	Recommendation
memory=	0%-100%,100%	<ul style="list-style-type: none"> ▶ Specify the minimum value for your important workload partition so that it receives memory resource under contention. ▶ Use the default value for soft/hard maximum.
totalProcesses=	unlimited	Use the default value.
totalThreads=	unlimited	Use the default value.
procVirtMem=	unlimited	Use the default value (unless you want to control the amount of virtual memory that your process consumes).

Important: The recommended minimum attributes to specify for **-R** option are:

```
-R shares_CPU=<xxx> shares_memory=<yyy>
```

After this initial setting, the administrator can use the **wlmstat** command from the global environment to monitor the utilization of each WPAR and adjust them accordingly.

Because you can do the resource control dynamically, it does not incur any downtime for any application running in any WPAR.

7.4 Practical examples of resource allocation

With workload partition resource control, the system administrator can modify any of the resource settings dynamically. The change is effective immediately.

Resource control for a workload partition can be set:

- ▶ During the creation of the workload partition using the **-R** option of the **mkwpar** or **wparexec** command
- ▶ While the workload partition is running using the **chwpar** command with the same **-R** option

All of the workload partition resource control settings can be done either via the command-line interface or SMIT as in *smitty wpar* under "Change/Show General Characteristics."

All of these settings apply to both system and application workload partitions.

7.4.1 CPU

There are two approaches to control the usage of the CPU: share-based and percentage-based.

The share-based approach means that you are allocating the available resource to the workload partitions based on their relative importance. (This amount of resource is sometimes called the “*target*” because it is the amount that WLM tries to give to the workload partition.)

For example, if WPAR A has 5 shares and WPAR B has 10 shares:

Total shares=5+10=15

WPAR A=5/15=33%

WPAR B=10/15=66%

WLM will try to give 33% of the resources to WPAR A and 66% to WPAR B.

If a new WPAR C is added with 15 shares:

Total shares=5+10+15=30

WPAR A=5/30=17%

WPAR B=10/30=33%

WPAR C=15/30=50%

Now, the target amount of resources for WPAR A and B changed to 17% and 33% respectively, due to the fact that WPAR C, with 15 shares (or 15/30=50% of the total resources), has been introduced to the global environment.

By default, the number of shares for each workload partition is unlimited, which means that each is allowed to use as much CPU as it desires. The recommendation is to explicitly set the number of shares according to the relative importance of each workload partition that runs in the global environment.

If you want to change the number of shares for each workload partition, use the `chwparr -R shares_CPU` command as shown in Example 7-1 on page 244.

Example 7-1 Change CPU shares

```
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
RSet:
CPU Shares:                         unlimited
CPU Limits:                           0%-100%,100%
Memory Shares:                         unlimited
Memory Limits:                         0%-100%,100%
Per-Process Virtual Memory Limit:     unlimited
Total Processes:                       unlimited
Total Threads:                         unlimited
# chwpar -R shares_CPU=10 MySysWpar
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
RSet:
CPU Shares:                         10
CPU Limits:                           0%-100%,100%
Memory Shares:                         unlimited
Memory Limits:                         0%-100%,100%
Per-Process Virtual Memory Limit:     unlimited
Total Processes:                       unlimited
Total Threads:                         unlimited
```

If a minimum resource amount or a limit of the maximum amount of a resource must be set for a workload partition, you need to use percentage-based controls.

The minimum value corresponds to a guaranteed capacity for the workload partition. This means that the workload partition is allocated at least that minimum processing power if it requires the resource. However, if it is idle, the resources can be used by other workload partitions.

The soft maximum is the maximum limit that the workload partition cannot exceed if there is contention for the resource. If the resource is idle, then it can exceed this amount.

The hard maximum is the absolute maximum limit that the workload partition cannot exceed.

Tip: We recommend that you start with the share-based approach to control the resources. In almost all cases, it will satisfy your requirements.

Apply the percentage-based approach *only* for cases that cannot be addressed via the share-based approach, for example:

- ▶ Use minimum limit to insure the amount of resource required. For example, an application that has low resource usage but requires a fast response time must have some minimum limit specified for memory, so that some of its pages still remain in memory to prevent the case where all pages got stolen during a period of inactivity and need to be paged-in.
- ▶ Use maximum limit to contain some resource-intensive, low-priority jobs. For example, a backup job must have its maximum resources set so that it will not interfere with other more important workloads.
- ▶ Avoid using maximum limit for memory, because it might cause paging activities, which impact the system performance. Instead of specifying the memory maximum limit for a workload partition, it is better to specify the memory minimum limit for other workload partitions running in the same global environment instead.

Example 7-2 shows the syntax of the **chwp** **-R** **command** in order to change percentage-based limits and how the change is reported by **lswpar**.

Example 7-2 Change CPU limits

```
# chwp -R CPU=6%-30%,45% MySysWpar
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
RSet:
CPU Shares:                         10
CPU Limits:                         6%-30%,45%
Memory Shares:                        unlimited
Memory Limits:                        0%-100%,100%
Per-Process Virtual Memory Limit:    unlimited
Total Processes:                      unlimited
Total Threads:                        unlimited
```

In this example, if MySysWpar is the only workload partition in the system, the target CPU resource will be 10/10=100%.

However, because we have set the percentage-based limits for the CPU usage to 6%-30%,45%, the WPAR is not able to use more than 45% (hard limit) of the CPU resources.

Note: Resource usage of a workload partition can be with shares, percentages, or both methods in conjunction.

When both methods are in use, the percentage-based control has precedence over the shared-based control. In other words, the percentage-based method will limit the range of values that the shared-based method can have.

7.4.2 Memory

As with the CPU resource controls, memory resource controls can be defined in shares or percentages. They have the same meaning and usage as the CPU resource controls.

Example 7-3 shows how to change the memory share to 20 and also set the minimum memory to 5%.

Example 7-3 Change memory shares and limits

```
# chwpar -R shares_memory=20 MySysWpar
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
RSet:
CPU Shares:                            10
CPU Limits:                            6%-30%,45%
Memory Shares:                        20
Memory Limits:                         0%-100%,100%
Per-Process Virtual Memory Limit:      unlimited
Total Processes:                       unlimited
Total Threads:                         unlimited

# chwpar -R memory=5%-100%,100% MySysWpar
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
RSet:
```


CPU Shares:	10
CPU Limits:	6%-30%,45%
Memory Shares:	20
Memory Limits:	5%-100%,100%
Per-Process Virtual Memory Limit:	unlimited
Total Processes:	unlimited
Total Threads:	unlimited

7.4.3 Processes and threads

The number of processes, as well as the number of threads, can be controlled in the workload partition.

If the maximum number of process attributes is not set, it is still limited by the *maxuproc* attribute for sys0 in the global environment for non-root users and unlimited for root.

Example 7-4 shows how to set the total number of processes to 128 and also set the total number of threads to 512.

Example 7-4 Set the number of processes and threads

```
# chwpar -R totalProcesses=128 MySysWpar
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                yes
RSet:
CPU Shares:            10
CPU Limits:            6%-30%,45%
Memory Shares:         20
Memory Limits:         5%-10%,20%
Per-Process Virtual Memory Limit: unlimited
Total Processes:      128
Total Threads:         unlimited

# chwpar -R totalThreads=64 MySysWpar
*****
**ERROR: resources.totalThreads (64) must be >= totalProcesses (128)
*****

# chwpar -R totalThreads=512 MySysWpar
# lswpar -R MySysWpar
=====
```

```
MySysWpar - Active
```

```
=====
Active:                               yes
RSet:
CPU Shares:                            10
CPU Limits:                            6%-30%,45%
Memory Shares:                         20
Memory Limits:                         5%-10%,20%
Per-Process Virtual Memory Limit:      unlimited
Total Processes:                       128
Total Threads:                       512
=====
```

In this example, the error message illustrates that the system enforces that the number of threads defined by the system administrator must be greater or equal to the number of processes.

7.4.4 Process virtual memory

The maximum amount of virtual memory that a process can have in a workload partition can be controlled.

Example 7-5 shows how to set the maximum amount of process virtual memory to 1 TB.

Example 7-5 Set the maximum amount of process virtual memory

```
=====
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
RSet:
CPU Shares:                            unlimited
CPU Limits:                            0%-100%,100%
Memory Shares:                         unlimited
Memory Limits:                         0%-100%,100%
Per-Process Virtual Memory Limit:      unlimited
Total Processes:                       unlimited
Total Threads:                         unlimited
# chwpar -R procVirtMem=1TB MySysWpar
# lswpar -R MySysWpar
=====
MySysWpar - Active
=====
Active:                               yes
```

RSet:	
CPU Shares:	unlimited
CPU Limits:	0%-100%,100%
Memory Shares:	unlimited
Memory Limits:	0%-100%,100%
Per-Process Virtual Memory Limit:	1048576MB
Total Processes:	unlimited
Total Threads:	unlimited

7.5 Use case scenario

This is a use case scenario for WLM and resource control for WPARs.

A client running WebSphere Application Server has two application servers in a logical partition. The client's first reaction is to use WLM to assign resources to each of the application servers so that they do not interfere with each other.

However, the WPAR command-line interface provides an easier way to achieve this goal in just two commands. This is done by launching each application server as the argument of a `wparexec` command, along with resource control variable settings.

For example, assuming the client's WebSphere application can be started with the `/home/script/startserver.sh` and the client wants to respectively allocate 30% and 70% of the resources to the development server and the production server, the client can simply use the commands:

```
wparexec -R shares_CPU=30 shares_memory=30 /home/script/startserver.sh  
devServer
```

```
wparexec -R shares_CPU=70 shares_memory=70 /home/script/startserver.sh  
prodServer
```

Note that because share-based control only is used, if one of the servers does not start, the other is not constrained (by the percentage-based control) and can use all available resources in the system.

7.6 WPAR resource control changes to WLM

WLM functions in AIX have been enhanced to support resource control for workload partitions.

All WLM concepts addressed in this section are described in more detail in *AIX 5L Workload Manager (WLM)*, SG24-5977.

7.6.1 Number of user-defined superclasses

Prior to AIX V6, WLM supported only 69 superclasses: five predefined and 64 user-defined.

WLM implementation in AIX V6 has now been extended to support 8192 user-defined superclasses in order to support the maximum number of active workload partitions at any one time (8192).

Important: Currently, WLM is not supported inside a WPAR.

7.6.2 Resource limit granularity

As the number of superclasses increases beyond 100, it becomes necessary to be able to specify the resource limit with a finer granularity than an integer number percentage.

Resource control for workload partition allows up to two decimal digits after the decimal point (instead of just an integer value) in the percentage-based specification.

Thus, each value in the percentage-based limit specification, *<min>%*, *<sMax>%*, and *<hMax>%*, now can have up to two decimal digits after the decimal point.

7.6.3 Changes to WLM commands

The `wlmstat` command now behaves differently depending on whether it is run in the global environment or in a workload partition:

- ▶ The `wlmstat` command in the global environment displays the information about all classes that are not associated with the workload partition.
- ▶ Displaying information about classes associated with a workload partition uses the `-@` option in the `wlmstat` command, as shown in Example 7-6.

Example 7-6 wlmstat command to display WPAR statistics

```
# wlmstat -@ 2
          CLASS   CPU   MEM   DKIO
MySysWpar  0.00  0.04  0.00
MyCompetingWpar  0.24  0.04  0.00
```

TOTAL	0.24	0.08	0.00
CLASS	CPU	MEM	DKIO
MySysWpar	0.00	0.04	0.00
MyCompetingWpar	0.44	0.04	0.00
TOTAL	0.44	0.08	0.00

...

7.7 Questions about WLM and WPAR resource control

For those who are familiar with WLM concepts, here are several frequently asked questions about WLM and workload partition resource control.

7.7.1 The tier

Question: The tier is not specified with the `-R` attribute, so how is the tier used by WLM for a workload partition?

Answer: All workload partitions are put in tier 0.

Question: Can I change the tier for a workload partition?

Answer: No, currently this is not supported.

7.7.2 Superclass and subclass

Question: The superclass and subclass are not specified with `-R` attribute, so what are the superclass and subclass used by WLM for a workload partition?

Answer: A superclass with the same name as the workload partition (from `-n` option) will be created for each workload partition. A subclass is not explicitly defined, so all processes in the workload partition are in the Default subclass.

7.7.3 Class assignment rule

Question: The class assignment rules are not specified with `-R` attribute, so how can WLM classify the process to a superclass?

Answer: The first process in each workload partition, `init`, is classified by the kernel to a superclass that has the same name as that workload partition. All

children of this init process and all processes in the workload partition will inherit this superclass classification.

In summary, all processes in the same workload partition are classified into one superclass, and processes in different workload partitions are classified into different superclasses.

7.7.4 Enabling resource control when creating a WPAR

Question: Do I need to enable resource control when I create a workload partition?

Answer: Resource control is enabled by default. We highly recommend that you keep it enabled.

If you do not enable resource control (specify `-R active=no`), you will not have WLM available to control and adjust the amount of resources consumed by processes in the workload partition.

Thus, it means that any process can freely consume resources as much as it can without any restriction or limitation. This is likely to cause high interference among the workload partition.

By enabling resource control, you are enforcing the resource consumption restriction and limitation, and thus, more likely to obtain better performance isolation among workload partitions.

Important: Besides enabling the resource control, at a minimum, you must also specify `shares_CPU` and `shares_memory`, because the default WLM settings for them are unlimited (or no control from WLM).

7.7.5 Enabling resource control for selected WPARs

Question: Can I enable resource control for only some of the WPARs?

Answer: We highly recommend that you keep the default setting of `-R active=yes` for *all* workload partitions, because WLM can most effectively control and manage resources if all of the workload partitions are under its control.

7.7.6 Using WLM commands directly for resource control

Question: Can I use WLM commands directly to set or change the resource settings of a workload partition?

Answer: No. The supported resource control settings for workload partition are through:

- ▶ **-R** options of the **mkwpar**, **wparexec**, or **chwpar** command
- ▶ WPAR Manager GUI

7.7.7 Using WLM to manage other workloads

Question: Can I use WLM to manage other workloads in the global environment?

Answer: Yes, WLM in the global environment is supported.

However, the administrator has to be aware that there are WLM superclasses created for each resource-controlled workload partition and that resource controls for workload partitions have to be assigned and changed only through the WPAR-related tools and commands.

The **wlmstat** command has been extended so that resource utilization for workload partitions can be monitored through the same interface that is used for user-defined WLM superclasses and subclasses.

Archived



Device management

In this chapter, we explain which devices are available to WPARs and how you can add or remove them from the WPARs. This chapter contains the following sections:

- ▶ 8.1, “Device visibility in the workload partition” on page 256
- ▶ 8.2, “Supported device categories in a WPAR environments” on page 259

8.1 Device visibility in the workload partition

One of the major differences between operating in the global environment and in the workload partition is the device visibility. The `/etc/wpars/devexports` file describes the available devices to WPARs and which devices are exported by default. That is, this file defines which devices go into a default WPAR. In Example 8-1, we show part of the contents of that file where you can see that some devices are commented out.

Example 8-1 Part of the `/etc/wpars/devexports` file

```
...
# The black hole device that throws away any input
device:
  devname = "/dev/null"

# Device that generates zero output
device:
  devname = "/dev/zero"

# STREAMS:
#device:
#  devname = "/dev/echo"
...
```

You must edit this file if you have different needs. By default, all the devices that are not commented out with the number sign or hash sign (#) character are exported with the `mkwpar` command. Each stanza starts with **device:** and continues with three parameters:

device name	Required. Examples are <code>/dev/zero</code> , <code>/dev/null</code> (attribute name: "devname")
device type	Clone or pseudo (attribute name: "devtype")
automatic export option	yes or no. If export is not specified, yes is assumed. (attribute name: "export")

Editing the `/etc/wpars/devexports` file will affect all WPARs that you create by default. Alternatively, you can pass different devices to the `mkwpar` command with the `-D` flag. In Example 8-2 on page 257, we show how to prevent the creation by default of the `/dev/zero` device.

Example 8-2 Editing the /etc/wpars/devexports file

```
...
# Device that generates zero output
device:
  devname = "/dev/zero"
  export = no
...
```

After the creation of a WPAR, the **mkwpar** command no longer helps you add new devices to the WPAR. Instead, we use the **chwpar** command as shown in Example 8-3 to change the available devices inside the WPAR.

Example 8-3 Changing the WPAR's devices

```
# lswpar
Name          State  Type  Hostname  Directory
-----
MySysWpar    A      S     MySysWpar /wpars/MySysWpar
# clogin MySysWpar
# ls -l /dev/zero
ls: 0653-341 The file /dev/zero does not exist.
# exit
# chwpar -D devname=/dev/zero export=yes MySysWpar
# clogin MySysWpar
# /usr/lib/methods/wpar_cfgpseudo
Importing device /dev/null, type = 1
Importing device /dev/tty, type = 1
Importing device /dev/console, type = 1
Importing device /dev/clone, type = 1
Importing device /dev/sad, type = 3
Importing device /dev/xti/tcp, type = 3
Importing device /dev/xti/tcp6, type = 3
Importing device /dev/xti/udp, type = 3
Importing device /dev/xti/udp6, type = 3
Importing device /dev/xti/unixdg, type = 3
Importing device /dev/xti/unixst, type = 3
Importing device /dev/error, type = 1
Importing device /dev/errorctl, type = 1
Importing device /dev/audit, type = 1
Importing device /dev/nvram, type = 1
Importing device /dev/zero, type = 1
# ls -l /dev/zero
crw-rw-rw-  1 root    system    2,  3 May 11 14:37 /dev/zero
```

The **chwp** command writes into the `/etc/wpars/<name of the WPAR>.cf`; thus, making it permanent. When you reboot or stop and start the WPAR, those devices will start with the WPAR. Example 8-4 shows the output of device-related commands running in the system workload partition.

Example 8-4 lsdev -C command output in a system workload partition

```
# lsdev -C
inet0 Available Internet Network Extension
iscsi0 Defined iSCSI Protocol Device
lo0 Defined Loopback Network Interface
lvdd Defined LVM Device Driver
pty0 Available Asynchronous Pseudo-Terminal
rcm0 Defined Rendering Context Manager Subsystem
sys0 Available System Object
# ls -la /dev
total 56
drwxrwxr-x 5 root system 4096 May 03 16:48 .
drwxr-xr-x 17 root system 4096 May 03 16:48 ..
drwxrwx--- 2 root system 4096 May 03 16:47 .SRC-unix
-rw--w--w- 1 root system 0 May 03 16:48 Global
srwxrwxrwx 1 root system 0 May 03 16:46 SRC
cr--r----T 1 root system 8, 0 May 03 16:46 audit
crw-rw-rw- 1 root system 12, 0 May 03 16:46 clone
crw--w--w- 1 root system 4, 0 May 03 16:06 console
crw--w--w- 1 root system 6, 0 May 03 16:46 error
crw----- 1 root system 6, 1 May 03 16:46 errorctl
srw-rw-rw- 1 root system 0 May 03 16:46 log
crw-rw-rw- 1 root system 2, 2 May 03 16:48 null
crw-r--r-T 1 root system 3, 0 May 03 16:46 nvram
crw-rw-rw- 1 root system 12, 20 May 03 16:46 ptc
drwxr-xr-x 2 root system 16384 May 03 16:46 pts
crw-rw-rw- 1 root system 22, 0 May 03 16:46 ptyp0
...
crw-rw-rw- 1 root system 22, 15 May 03 16:46 ptypf
crw-r--r-- 1 root system 11, 0 May 03 16:46 random
crw-rw-rw- 1 root system 12, 13 May 03 16:46 sad
crw-rw-rw- 1 root system 1, 0 May 03 16:46 tty
crw-rw-rw- 1 root system 23, 0 May 03 16:46 tty0
...
crw-rw-rw- 1 root system 23, 15 May 03 16:46 ttypf
crw-r--r-- 1 root system 11, 1 May 03 16:46 urandom
drwxr-xr-x 2 root system 256 May 03 16:46 xti
crw-rw-rw- 1 root system 2, 3 May 03 16:46 zero
```

8.2 Supported device categories in a WPAR environments

There are five classes of devices in a WPAR environment:

- Global** Globally exported devices are devices that are configured in the global environment and are made available for use in the workload partition. Examples of global devices are `/dev/zero` and `/dev/null`.
- Global environment only** Global environment only devices, such as `/dev/kmem`, `/dev/bus`, or `/dev/sysdump`, are not available inside a WPAR. These types of devices are reserved for the global environment only.
- Physical devices** Physical devices, such as disks, CD-ROMs, or devices dependent on physical devices, such as logical volumes, are only available in the global environment.
- Clone** Clone devices and those that use the streams subsystem are devices that depend on minor and major numbers, hash keys and, in some cases, a channel number. These devices are created as special files that access the global clone device and return the usable device handle. Devices that provide a transport layer interface to socket-based protocols are an example of clone devices.
- Minor number dependent** There are several types of devices that are minor number dependent: specific or finite devices and local or buffer-based devices. Some devices have a unique minor number inside the WPAR but not in the global environment. The WPAR accesses them through a dictionary created when the device is created. Errors and traces are examples of local or buffer-based devices. These are basically two-dimensional arrays of buffers with the buffer and the WPAR keyed in.

Archived

Appendixes

In Appendix A, “Modified AIX commands” on page 263, we provide a list of commands that have been modified to support workload partitions.

Archived



Modified AIX commands

This appendix contains a list of commands in Table A-1 on page 264 that have been modified to support workload partitions. This list is copied from the *WPAR System Administration Guide*.

AIX modified commands

The commands in Table A-1 have been modified to support workload partitions.

Table A-1 AIX Modified Commands

Command	Flags/arguments	Behavior in a workload partition	Behavior in the Global Environment
acctcom	"- @ wparname"	Fails with workload partition not found message unless workload partition name is "Global". In this case, fails with "cannot open /var/adm/pacct" message	Executes normally displaying accounting records for workload partition "wparname".
acctcom	"- @ no argument"	Fails with "cannot open /var/adm/pacct" message	Executes normally displaying accounting records for all workload partitions. A workload partition name is displayed for each record.
acctctl	All options	Fails with a "not owner" message	Executes normally if user has correct privilege
clogin	-C wparname user cmd args	Not allowed in a workload partition	Runs cmd in the workload partition or login if no command is specified
df	All options	Displays information about workload partition mounted file systems only. Paths are displayed relative to the workload partition root.	Displays information about all file systems. Paths are absolute.
domainname	No arguments	Displays domainname of workload partition	Displays domainname of system
domainname	{newdomainname}	If executed by workload partition root sets domainname of workload partition	If executed by global root, sets domainname of system
hostid	No arguments	Displays host ID of workload partition	Displays host ID of system
hostid	{IP address hex number}	If executed by workload partition root, sets host ID of workload partition	If executed by global root, sets host ID of the system

Command	Flags/arguments	Behavior in a workload partition	Behavior in the Global Environment
hostname	No arguments	Displays host name of workload partition	Displays host name of system
hostname	{newhostname}	If executed by workload partition root and host name privilege allowed for workload partition, sets host name of workload partition	If executed by global root, sets host name of the system
ifconfig	All display options (-a -l)	Display information about the workload partition	Display information about the global
ioo		Non-functional in workload partition	no change in behavior
ipcrm	no "-@" argument	Removes IPC objects associated with the workload partition	Removes IPC objects associated with global
ipcrm	"-@ wparname"	Invalid unless workload partition name = Global. In this case, removes IPC object within the WPAR	Removes IPC objects associated with workload partition "wparname".
ipcs	no "-@" argument	Displays information about IPC objects created by processes within the wpar	Displays information about ipc objects created by by global processes. No workload partition associated objects are displayed.
ipcs	"-@"	Displays IPC information within the workload partition	Displays information about all IPC objects in the system. The name of the workload partition associated with the object is listed.
ipcs	"-@ wparname"	Displays no ipc information unless wparname = Global. Global case displays information about ipc objects associated with processes within the workload partition.	Displays information about IPC objects associated with processes within the workload partition named "wparname"
mkclass	all options	This command will only update "/etc/wlm" dir. Will fail updating kernel data	No change in behavior

Command	Flags/Arguments	Behavior in a workload partition	Behavior in the Global Environment
mount	No arguments	Displays only workload partition mounted file systems relative to the workload partition root.	Displays all mounted file systems with absolute paths
mount	With arguments	Only Network File System (NFS) mounts without cachefs allowed. Forced nosuid, nodev	No change in behavior
netstat	All options except -a -i	Fails - Cannot open /dev/kmem	Displays information about the whole system
netstat	"-a"	Displays all connections for the workload partition	Displays connection information for the global
netstat	"-l"	Displays all addresses for the workload partition	Displays all addresses for the global
netstat	new arg "-@" wparname	Not allowed in a workload partition	Displays either connection or address information for the workload partition or all workload partitions if wparname is not specified
nfso		Non-functional in workload partition	No change
no	All options except -a	Fails with an error message. -a options execute normally.	Executes normally if user has correct privilege
projctl	All options except qproj(s)	Fails with a "not owner" message. qproj(s) executes normally	Executes normally if user has correct privilege
ps	"-e"	Displays everything within the workload partition	Displays everything within the system. Processes are not screened from view unless a specific -@ <wparname> is also included.
ps	"-@"	Displays process information for processes in the workload partition. Workload partition name is included in output.	Displays process information for all processes in the system. The name of the workload partition is displayed on output.

Command	Flags/Arguments	Behavior in a workload partition	Behavior in the Global Environment
ps	"-@ wparname"	Displays no process information unless wparname = Global. Global case displays information about processes within the workload partition. The name of the workload partition is provided on output.	Displays information about processes associated with the workload partition named "wparname". The name of the workload partition is provided on output.
ps	"-o wpar"	Produces a wpar name header and the name of the workload partition associated with the process. This name is always "Global."	Produces a wpar name header and the name of the workload partition in which the process is executing.
schedo		Non-functional in workload partition	No change in behavior
uname	"-n"	Displays name of the workload partition	Displays node name of the system
vmo		Non-functional in workload partition	No change in behavior
wlmstat	all options	Will not work in this release	No change in behavior
wlmstat	"-@"	Will not work in workload partition	Will display data for meta class (workload partition class)
wlmtune	all options	Will not work in workload partition	No change in behavior
wlmcntrl	all options	Will not work in workload partition	No change in behavior

Archived

Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

IBM Redbooks

For information about ordering these publications, see “How to get IBM Redbooks publications” on page 270. Note that some of the documents referenced here might be available in softcopy only:

- ▶ *Advanced POWER Virtualization on IBM System p5: Introduction and Configuration*, SG24-7940
- ▶ *Advanced POWER Virtualization on IBM System p Virtual I/O Server Deployment Examples*, REDP-4224
- ▶ *AIX 5L Workload Manager (WLM)*, SG24-5977
- ▶ *AIX Logical Volume Manager from A to Z: Troubleshooting and Commands*, SG24-5433
- ▶ *AIX V6 Advanced Security Features Introduction and Configuration*, SG24-7430

Other publications

These publications are also relevant as further information sources:

- ▶ *IBM Workload Partitions Manager for AIX*, found at:
<http://publib.boulder.ibm.com/infocenter/pseries/v6r1/topic/com.ibm.aix.wpar1pp/wpar1pp.pdf>

Online resources

These Web sites are also relevant as further information sources:

- ▶ AIX Collaboration Center
<http://www-03.ibm.com/servers/aix/collaborationcenter/index.html>

- ▶ AIX V6 Open Beta

<http://www14.software.ibm.com/iwm/web/cc/earlyprograms/ibm/aix6beta>

How to get IBM Redbooks publications

You can search for, view, or download IBM Redbooks publications , Redpapers, Technotes, draft publications and Additional materials, as well as order hardcopy IBM Redbooks publications, at this Web site:

ibm.com/redbooks

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Index

Symbols

"WPAR database" 50
/etc/wpars 53

A

active relocation 12
Agent Manager 138
agentKeys.jks 160
agentTrust.jks 160
allocation
 memory 246
 Process virtual memory 248
 Processes and threads 247
alt_install 95
alternate disk installation 95
application mobility 31
Application WPAR
 defined 11
Application WPARs 27
APV 126

B

backup 102
base directory 29, 71
basedir 29
bffcreate 86

C

CAS 126
 Agent Manager 138
 Common Agent 137
 Resource manager 139
CAS Agent Manager daemon 153
CAS framework 136
CertificateRevocationList 160
changing a WPAR 69
changing WPAR hostname 70
checkpoint state 50
Checkpoint statefile 50
checkpoint statefile 53
chkptwpar 48, 231
chwpar 46

CLI 45
clic 100
clogin 47
cloning 102, 106
command line interface 45
Commands
 chkwpar 48
 chwpar 46
 CLI 48
 clogin 43, 47
 halt 48
 killwpar 48
 Issavewpar 46
 lswpar 42, 46
 mcrk_admin 48
 mkrset 49
 mkwpar 46, 62
 mkwpardata 46
 rc.wpars 46
 reboot 48
 rebootwpar 46
 restartwpar 48
 restwarfiles 46
 restwpar 46
 resumewpar 48
 rmwpar 46
 rsh 47
 savewpar 46
 shutdown 48
 ssh 47
 startwpar 42, 46
 stopwpar 46
 syncroot 46
 syncwpar 46
 telnet 47
 topas 48
 wparexec 46
Common Agent 137
Common Agent Services 126
common agent services framework 136
complex WPAR creation 81
configure-agent 160
CPU Allocation 238
create a user inside a WPAR 112

D

- database 50
- datavg 108
- DB2 141
- default access
 - users and root 111
- devexportsfile 105
- devices
 - supported categories for WPAR 259
 - visibility 256
- Dynamic Logical Partitions 6

E

- Eclipse 45
- effective utilization 135
- EFS 100
- encrypted filesystem 100
- encrypted filesystems 102
- encryption keys 101

F

- Files
 - /etc/hosts 73
 - /etc/resolv.conf 73
 - /etc/wpars 53
 - devexports 52
 - secattr 52
- filesets 45
- filesystems
 - application WPARs 27
 - partition mobility 32
 - system WPARs 29
- firewall 23

G

- global environment 8, 10, 26
 - adapters and I/O 26
 - considerations 26

H

- HACMP 126
- halt 48
- host name 70
- hot-migration 12

I

- I/O buffer cache 116
- image.data 108
- inheritprivs 113
- init 42
- inittab 154
- innateprivs 113
- installation
 - filesets 45
- installing openssh 89
- installp 88
- interface 73
- ioo 122

K

- keystores 100
- killwpar 48, 231

L

- live application mobility 31
- load balancing 131
- Logging and Tracing 228
- Logical Partitions 6
- lpd 119
- lssavewpar 46
- lssrc 119
- lswpar 46

M

- measured utilization 135
- Memory Allocation 238
- micropartitions 7
- mksysb 51
- mkwpar 46, 62
- mkwpardata 46
- mobility CLI
 - CLI 230
- mounting a temporary filesystem 86
- mounting nfs inside WPAR 88
- MPIO 28

N

- network settings
 - changing 73
- Networking 21
- networks 72
- NFS considerations 102

NFS filesystem 32
nfsd 122
NIM 33
 using 94

O

openssh 88
openssl 100

P

Partition Load Manager 128
Partition mobility 12
partition mobility 31
 and NFS 34
 file system setup 32
 NFS file systems example 35
 topology requirements 33
performance metrics 127
PKI technology 100
Planning 19
 filesystems 23
 firewall 23
 General considerations 21
 Networking 21
 NFS filesystems 32
 TCPIP Ports 23
 WPAR manager 21
planning
 global environment 26
PLM 128
print spooling 115
privilege commands 15
privileges 114
processes 239
processes in the global environment 118
ps 118
PV_FS_CHOWN 113
PV_FS_MOUNT 114
PV_KER_WPAR 113

Q

qdaemon 119
QUEUE 115

R

raso 122
RBAC 43, 113

 privileges 114
rc.wpars 46
reboot 48
rebootwpar 46, 233
recovery 51, 84
 root password 112
Redbooks Web site 270
 Contact us xv
relocation 12
 active 12
 step by step 233
relocation flow 130
relocation states 130
removal 84
resource allocation 15, 242
resource control 72
resource control attributes 240
resource control overview
 WPAR resource control 238
resource limit granularity 250
resource management 49
Resource manager 139
restartwpar 48, 232
restore 102–103
restwpar 46, 103, 105
restwparfiles 46
resumewpar 48, 232–233
rmwpar 46
role based access control 113
root password recovery 112
rootvg 108
rsh 47

S

savevg 102
savewpar 46, 102–103
schedo 122
server group 132
Service Level Agreements 13
Service Update Management Assistant 94
shared /usr with writable filesystem 98
shutdown 48
SMIT 44, 46
smitty 44
specification file 52
specification files 81
Specifications file 50
spooler 119

- SRC 119
- ssh 47
- startsrc 119
- startwpar 46
- State
 - Active 56
 - Broken 57
 - Defined 56
 - Frozen 57
 - Loaded 57
 - Paused 57
 - transition 57
 - Transitional 57
- state 56
- state file 50, 53
- state management 55
- states 56
 - during relocation 130
- stopwpar 46, 233
- subnet 73
- SUMA 94
- superclass 250
- swvpdmgr 92
- synchronization 106
- syncroot 46, 90
- syncwpar 46, 90, 100
- System WPAR 28
 - defined 11
- system WPAR
 - create using WPAR Manager 162
- System WPARs 28

T

- TCPIP Ports 23
- telnet 47
- threads 239
- topas 48, 120
- Trusted Computing Base 54
- tuning tools 122

U

- uptime
 - Commands
 - uptime 48
- user-defined superclass 250
- users and groups 111
- using NIM 94
- utilization 135

V

- virtual memory 239
- virtualization
 - overview 5
 - WPAR based 7
- virtualization technologies 5
- vmo 122
- vscsi 28

W

- WLM 4, 44, 49
 - concepts 251
 - considerations 49
- wlmstat 123
- Workload Partition
 - defined 8
- workload partition
 - when to use 13
- WPAR
 - application 11
 - application migration 192
 - application to system comparison 42
 - base directory 71
 - changing 69
 - changing network settings 73
 - checkpoint state 50
 - complex WPAR creation 81
 - create from other working WPARs 83
 - general considerations 21
 - hostname 70
 - listing in global environment 30
 - load balancing 131
 - LPAR comparison 36
 - management options 44
 - managing users and groups 111
 - mobility 126
 - networks 72
 - printing from 115
 - recovery 51
 - relocation 12
 - relocation flow 130
 - Relocation Policy 132
 - relocation states 130
 - resource allocation 15
 - resource control 72
 - resource control attributes 240
 - resource optimization 14
 - server group 132

- specification 52
- state 56
- state management 55
- states 56
- system 11
- viewing processes inside 117
- WPAR Agent 160
- WPAR Manager
 - Command line interface 48
 - components 136
 - DB2 setup 146
 - installation 140
 - monitoring 127
 - overview 126
 - ports 140
 - relocation 128
 - workload balancer 132
- WPAR manager
 - deployment overview 21
- WPAR Manager Configurator 148
- WPAR Manager daemon 153
- WPAR mobility
 - topology requirements 33
- WPAR relocation
 - Fully-automatic 129
 - manual 129
 - Semi-automatic 129
- wparexec 46
- wparamgt.mgr.rte 140
- WPARAMGTDB 142
- WPMConfig 148
- WPMConfig.log 153
- writesrv 119

Archived



Introduction to Workload Partition Management in IBM AIX Version 6.1

(0.5" spine)
0.475" <-> 0.875"
250 <-> 459 pages



Redbooks

Introduction to Workload Partition Management in IBM AIX Version 6.1

Overview, applicability, and planning information for system architects

Understanding and applying WPAR technology for efficient IBM System p resource utilization

Walk-through examples for AIX system administrators

This IBM Redbooks publication presents Workload Partitions (WPARs), a set of completely new software-based system virtualization features introduced in IBM AIX Version 6.1.

Adding WPARs to the AIX operating system provides a new level of system virtualization capability. WPARs complement the already existing AIX and System p virtualization features, by allowing:

- ▶ The partitioning of an AIX instance into multiple environments, each hosting applications and providing isolation from applications executing in the other environments
- ▶ The ability to checkpoint and restart execution of applications without modification of the application code
- ▶ The capability to relocate a live application into a different AIX instance, whether it is hosted in an LPAR of the same physical server or executing on a different physical server
- ▶ Fine-grained control of resource utilization by each application executing within an AIX instance

**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