

IBM PowerHA SystemMirror for AIX Cookbook

Dino Quintero

Tim Simon

Shawn Bodily

Vera Cruz

Sachin P. Deshmukh

Karim El Barkouky

Youssef Largou

Jean-Manuel Lenez

Vivek Shukla

Kulwinder Singh



Power Systems



IBM Redbooks

IBM PowerHA SystemMirror for AIX Cookbook

January 2024

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

Third Edition (January 2024)

This edition applies to PowerHA SystemMirror for AIX 7.2.7, AIX 7.2 Technology Level (TL) 5, AIX 7.3 TL 0, and AIX 7.3 TL 1

© Copyright International Business Machines Corporation 2009, 2024. 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	xiii
Trademarks	xiv
Prefacexv
Authorsxv
Now you can become a published author, too!xviii
Comments welcomexix
Stay connected to IBM Redbooksxix
Summary of changesxxi
January 2024, Third Editionxxi
Part 1. Introduction	1
Chapter 1. Introducing IBM PowerHA SystemMirror for AIX	3
1.1 What is PowerHA SystemMirror for AIX	4
1.1.1 High availability	6
1.1.2 Cluster multiprocessing	6
1.2 Availability solutions: An overview	7
1.2.1 Downtime	8
1.2.2 Single point of failure	9
1.3 History and evolution	10
1.3.1 PowerHA SystemMirror 7.2.4	11
1.3.2 PowerHA SystemMirror 7.2.5	12
1.3.3 PowerHA SystemMirror 7.2.6	15
1.3.4 PowerHA SystemMirror 7.2.7	16
1.4 High availability terminology and concepts	17
1.4.1 Terminology	17
1.4.2 Concepts	18
1.5 Fault-tolerant versus high availability	19
1.5.1 Fault-tolerant systems	19
1.5.2 High availability systems	19
1.6 Software planning	20
1.6.1 AIX level and related requirements	20
1.6.2 Licensing for other software	23
1.6.3 PowerHA licensing	24
1.7 PowerHA software installation	24
1.7.1 Checking for prerequisites	25
1.7.2 New installation	25
1.7.3 Installing PowerHA	25
Chapter 2. High availability components	27
2.1 PowerHA configuration data	28
2.2 Software components	29
2.3 Cluster topology	30
2.3.1 PowerHA cluster	31
2.3.2 Sites	31
2.3.3 Cluster nodes	32
2.3.4 Networks	32

2.3.5 Cluster Aware AIX	34
2.3.6 Reliable Scalable Cluster Technology	40
2.3.7 Internet Protocol networks	41
2.3.8 The IP Address Takeover mechanism	41
2.3.9 Persistent IP label or address	42
2.3.10 Cluster heartbeat settings	43
2.3.11 Network security considerations	43
2.4 Resources and resource groups	44
2.4.1 Definitions	44
2.4.2 Resources	45
2.4.3 NFS	50
2.4.4 Application controller scripts	51
2.4.5 Application monitors	52
2.4.6 Tape resources	52
2.4.7 User-defined resources and types	52
2.4.8 Resource groups	53
2.5 Smart assists	62
2.6 Other features	62
2.6.1 Notifications	62
2.6.2 Rootvg system event	63
2.6.3 Capacity on Demand and dynamic logical partition support on failover	63
2.6.4 File collections	64
2.6.5 PowerHA SystemMirror Enterprise Edition	64
2.7 Limits	65
2.8 Storage characteristics	66
2.8.1 Shared LVM	66
2.9 Shared storage configuration	66
2.9.1 Shared LVM requirements	67
2.10 PowerHA cluster events	69
Part 2. Planning, installation, and migration	71
Chapter 3. Planning	73
3.1 High availability planning	74
3.2 Planning for PowerHA	74
3.2.1 Planning strategy and example	75
3.2.2 Planning tools	75
3.2.3 Getting started	76
3.2.4 Current environment	76
3.2.5 Addressing single points of failure	77
3.2.6 Initial cluster design	78
3.2.7 Naming conventions	79
3.2.8 Completing the cluster overview planning worksheet	80
3.3 Planning cluster hardware	81
3.3.1 Overview of cluster hardware	81
3.3.2 Completing the cluster hardware planning worksheet	82
3.4 Planning cluster software	82
3.4.1 AIX and RSCT levels	83
3.4.2 Virtual Ethernet and vSCSI support	83
3.4.3 Required AIX file sets	83
3.4.4 PowerHA 7.2.7 file sets	84
3.4.5 AIX files that can be altered by PowerHA	87
3.4.6 Application software	90

3.4.7 Licensing.....	90
3.4.8 Completing the software planning worksheet.....	91
3.5 Operating system considerations	91
3.6 Planning security.....	92
3.6.1 Cluster security	92
3.6.2 User administration.....	92
3.6.3 High Availability Cluster Multiprocessing group	93
3.6.4 Planning for PoweHA file collections.....	93
3.7 Planning cluster networks	94
3.7.1 Terminology	96
3.7.2 General network considerations	96
3.7.3 IP Address Takeover planning	104
3.7.4 Additional network planning considerations	105
3.7.5 Completing the network planning worksheets.....	106
3.8 Planning storage requirements	108
3.8.1 Internal disks.....	108
3.8.2 Cluster repository disk.....	108
3.8.3 SAN-based heartbeat	109
3.8.4 Shared disks	110
3.8.5 Enhanced concurrent mode volume groups	110
3.8.6 How fast disk takeover works	111
3.8.7 Enabling fast disk takeover.....	112
3.8.8 Shared logical volumes.....	113
3.8.9 Completing the storage planning worksheets.....	114
3.9 Application planning	115
3.9.1 Application controllers.....	117
3.9.2 Application monitoring.....	117
3.9.3 Availability analysis tool	118
3.9.4 Completing the application planning worksheets	118
3.10 Planning for resource groups	121
3.10.1 Resource group attributes.....	121
3.10.2 Completing the planning worksheet	123
3.11 Detailed cluster design	124
3.12 Developing a cluster test plan.....	125
3.12.1 Custom test plan.....	125
3.12.2 Cluster Test Tool.....	127
3.13 Developing a PowerHA 7.2.7 installation plan	127
3.14 Backing up the cluster configuration	128
3.15 Documenting the cluster	129
3.15.1 Native HTML report.....	129
3.16 Change and problem management.....	131
3.17 Planning tools	131
3.17.1 Paper planning worksheets.....	131
3.17.2 Cluster diagram.....	132
Chapter 4. Installation and configuration	133
4.1 Basic steps to implement a PowerHA cluster	134
4.2 Configuring PowerHA	138
4.2.1 General considerations for each configuration method.....	138
4.2.2 Standard configuration path	140
4.2.3 Defining clusters, nodes, and networks	140
4.2.4 Configuring the repository and the heartbeat method.....	141
4.2.5 Creating service IP labels	143

4.2.6	Creating a resource group	143
4.2.7	Creating a shared volume group	144
4.2.8	Creating shared logical volumes	146
4.2.9	Creating a shared jfs2log logical volume	147
4.2.10	Creating a shared file system	148
4.2.11	Creating more application controllers	149
4.2.12	Adding resources into a resource group	149
4.2.13	Verifying and synchronizing a cluster configuration	150
4.3	Installing the SMUI	151
4.3.1	Planning the SMUI installation	151
4.3.2	Installing SMUI clients (cluster nodes)	152
4.3.3	Installing the SMUI server	153
Chapter 5. Migration	159
5.1	Migration planning	160
5.1.1	PowerHA SystemMirror 7.2.7 requirements	160
5.2	Understanding PowerHA 7.2 migration options	162
5.2.1	Migration options	162
5.3	Migration scenarios	163
5.3.1	Migration matrix to PowerHA SystemMirror 7.2.7	163
5.3.2	Rolling migration	163
5.3.3	Snapshot migration	169
5.3.4	Offline migration	174
5.3.5	Nondisruptive migration	175
5.3.6	Migration by using cl_ezupdate	179
5.4	Other migration options	186
5.4.1	Using alt_disk_copy	186
5.4.2	Using nimadm	188
5.4.3	Live Update	190
5.5	Common migration errors	191
5.5.1	Stuck in migration	191
Part 3. Cluster administration	193
Chapter 6. Cluster maintenance	195
6.1	Change control and testing	196
6.1.1	Scope	196
6.1.2	Test cluster	196
6.2	Starting and stopping the cluster	197
6.2.1	Cluster services	198
6.2.2	Starting cluster services	198
6.2.3	Stopping cluster services	201
6.3	Resource group and application management	203
6.3.1	Bringing a resource group offline	203
6.3.2	Bringing a resource group online	205
6.3.3	Moving a resource group	207
6.3.4	Suspending and resuming application monitoring	209
6.4	Scenarios	211
6.4.1	PCI hot-plug replacement of a NIC	211
6.4.2	Service Packs	213
6.4.3	Storage	215
6.4.4	Applications	216
6.5	Updating multipath drivers	217
6.5.1	Cluster-wide update	217

6.5.2 Individual node update	219
6.5.3 Steps for maintenance on PowerHA before Version 7.1.3 SP1	221
6.6 Repository disk replacement.....	222
6.6.1 Automatic Repository Update.....	222
6.6.2 Manual repository swap	222
6.7 Critical volume groups.....	224
6.8 Cluster Test Tool.....	226
6.8.1 Test duration.....	227
6.8.2 Considerations	227
6.8.3 Automated testing.....	228
6.8.4 Custom testing	232
Chapter 7. Cluster management	251
7.1 Cluster Single Point of Control	252
7.1.1 The C-SPOC SMIT menu.....	253
7.2 File collections.....	254
7.2.1 Predefined file collections	254
7.2.2 Managing file collections.....	257
7.3 User administration	261
7.3.1 C-SPOC user and group administration	261
7.3.2 Password management	269
7.3.3 Encrypted File System management	274
7.4 Shared storage management	275
7.4.1 Updating LVM components.....	275
7.4.2 Enhanced concurrent volume group LVM limitations	278
7.4.3 Dynamic volume expansion	278
7.4.4 C-SPOC Storage menu	282
7.4.5 Examples	283
7.4.6 C-SPOC command-line interface	299
7.5 Time synchronization	300
7.6 Cluster verification and synchronization	300
7.6.1 Cluster verification and synchronization by using SMIT	300
7.6.2 Cluster verification and synchronization by using a CLI (clmgr)	303
7.6.3 Dynamic cluster reconfiguration with DARE	306
7.6.4 Changing between multicast and unicast	308
7.6.5 Verification log files	309
7.6.6 Running automatic corrective actions during verification	310
7.6.7 Automatic cluster verification	311
7.7 Monitoring PowerHA	312
7.7.1 Cluster status checking utilities.....	313
7.7.2 Other cluster monitoring tools.....	318
7.7.3 Topology information commands	319
7.7.4 Resource group information commands	322
7.7.5 Log files.....	324
7.7.6 Using the clanalyze log analysis tool	330
7.7.7 SMUI log file viewing.....	332
7.7.8 Error notification	333
7.7.9 Application monitoring	333
7.7.10 Measuring application availability	344
Chapter 8. Cluster security	347
8.1 Cluster security	348
8.1.1 The /etc/cluster/rhosts file	348

8.1.2 Additional cluster security features	349
8.1.3 Cluster communication over a virtual private network.	349
8.2 Using encrypted internode communication from CAA.	349
8.2.1 Self-signed certificate configuration	351
8.2.2 Custom certificate configuration	354
8.2.3 Symmetric fixed key only configuration.	356
8.2.4 Symmetric key distribution by using an asymmetric key pair	356
8.3 Secure remote command execution	357
8.4 PowerHA and firewalls	357
8.5 Federated security for cluster-wide security management	358
8.5.1 Federated security components	358
8.5.2 Federated security configuration requirement.	360
8.5.3 Federated security configuration details	362
Part 4. Advanced topics with examples	371
Chapter 9. IBM PowerHA and IBM PowerVM	373
9.1 Virtualization	374
9.2 Virtual I/O Server.	375
9.2.1 PowerHA and virtual storage	376
9.2.2 PowerHA and virtual Ethernet.	377
9.2.3 PowerHA and single root I/O virtualization and virtual network interface card	378
9.2.4 PowerHA and SAN heartbeat	378
9.3 Resource Optimized High Availability	380
9.3.1 Concepts and terminology	380
9.3.2 Planning	382
9.3.3 Configuring ROHA	385
9.3.4 Troubleshooting HMC verification errors.	405
9.4 ROHA testing	406
9.4.1 Example 1: Setting up a ROHA cluster without On/Off CoD.	406
9.4.2 Testing Example 1: Setting up one ROHA cluster without On/Off CoD	412
9.4.3 Example 2: Setting up one ROHA cluster with On/Off CoD	427
9.4.4 Testing Example 2 with On/Off CoD.	432
9.5 Live Partition Mobility	438
9.5.1 Performing LPM with SANcomm defined	439
Chapter 10. Extending resource group capabilities	441
10.1 Resource group attributes.	442
10.2 The settling time attribute	442
10.2.1 Behavior of the settling time attribute	442
10.2.2 Configuring the settling time for resource groups	443
10.2.3 Displaying the current settling time	443
10.2.4 Settling time scenarios	444
10.3 Serial processing order	446
10.3.1 Configuring a serial (acquisition and release) processing order.	446
10.3.2 Serial processing order scenario.	447
10.4 Node distribution policy	449
10.4.1 Configuring a resource group node-based distribution policy	449
10.4.2 Node-based distribution scenario	450
10.5 Dynamic node priority	452
10.5.1 Configuring a resource group with a predefined RMC-based DNP policy	453
10.5.2 How predefined RMC-based dynamic node priority functions	455
10.5.3 Configuring a resource group with an adaptive failover DNP policy	458
10.5.4 Testing adaptive failover dynamic node priority	462

10.6 Delayed fallback timer	463
10.6.1 Delayed fallback timer behavior	464
10.6.2 Configuring delayed fallback timers	464
10.6.3 Displaying delayed fallback timers in a resource group	465
10.7 Resource group dependencies	466
10.7.1 Resource group parent-child dependency	467
10.7.2 Resource group location dependency	468
10.7.3 Start After and Stop After dependencies	471
10.7.4 Combining various dependency relationships.	472
10.7.5 Displaying resource group dependencies.	473
10.7.6 Resource group dependency scenario	474
Chapter 11. Customizing resources and events	479
11.1 Overview of cluster events	480
11.2 System events	480
11.3 User-defined resources and types	481
11.3.1 Creating a user-defined resource type	482
11.3.2 Creating a user-defined resource	484
11.3.3 Adding a user-defined resource to a resource group	485
11.4 Writing scripts for custom events	486
11.5 Pre-event and post-event commands	486
11.5.1 Parallel processed resource groups and pre-event and post-event scripts	487
11.5.2 Configuring pre-event or post-event scripts	487
11.6 Automatic error notification	488
11.6.1 Disk monitoring consideration.	489
11.6.2 Setting up automatic error notification.	489
11.6.3 Listing automatic error notification	490
11.6.4 Removing automatic error notifications.	490
11.6.5 Using error notification	490
11.6.6 Customizing the event duration	492
11.6.7 Defining new events	493
Chapter 12. Network considerations	497
12.1 Multicast considerations	498
12.1.1 Multicast concepts	498
12.1.2 Multicast guidelines.	500
12.2 Distribution preference for service IP aliases	501
12.2.1 Configuring a service IP distribution policy	502
12.2.2 Example scenarios with a service IP distribution policy	504
12.3 Cluster tunables	505
12.3.1 Changing cluster-wide tunables	506
12.3.2 Resetting cluster tunables to the cluster defaults	508
12.3.3 Changing the network settings	508
12.4 Site-specific service IP labels	509
12.5 Understanding the netmon.cf file	517
12.5.1 The netmon.cf format for virtual Ethernet environments.	517
12.5.2 netmon.cf examples	519
12.5.3 Implications	519
12.6 Using poll_uplink	520
12.7 Understanding the chosts file	522
12.7.1 Creating the chosts file.	522
Chapter 13. Cross-Site Logical Volume Manager stretched campus cluster	525
13.1 Cross-site LVM mirroring overview	526

13.1.1 Requirements	527
13.1.2 Planning considerations	527
13.2 Test environment	530
13.3 Configuring a cross-site LVM cluster	531
13.3.1 Topology creation	531
13.3.2 Resource group creation.....	535
13.3.3 Defining resources	539
13.3.4 Defining an application controller	541
13.3.5 Defining and creating volume groups	542
13.3.6 Creating mirrored logical volumes and file systems	544
13.3.7 Cluster configuration validation.....	546
13.4 Testing	552
13.4.1 Local node failure within the primary site	553
13.4.2 Rolling node failures that are promoted to a site failure	554
13.4.3 Primary site local storage failure.....	556
13.4.4 Primary site remote storage failure.....	557
13.4.5 Primary site all storage failure.....	558
13.4.6 Primary site failure	559
Chapter 14. IBM PowerHA and IBM Power Virtual Server.....	563
14.1 Introduction	564
14.1.1 What is IBM Power Virtual Server.....	564
14.1.2 IBM Power and IBM Power Virtual Server HADR options	564
14.1.3 Cloud HADR for IBM Power Virtual Server.....	566
14.2 Disaster recovery replication methods for cloud	567
14.2.1 Storage-based data mirroring	567
14.2.2 OS-based data mirroring.....	568
14.2.3 Geographic Logical Volume Manager concepts	569
14.2.4 Block-storage based replication	571
14.2.5 File-storage based replication.....	572
14.3 Hybrid and multi-cloud deployment models	573
14.3.1 Hybrid cloud	573
14.3.2 Multiple public clouds	573
14.3.3 Cold disaster recovery	574
Chapter 15. Geographic Logical Volume Manager wizard	575
15.1 Introduction	576
15.1.1 Prerequisites.....	576
15.1.2 Limitations.....	577
15.1.3 Logs	577
15.2 Creating a cluster by using the SMUI	577
15.3 Adding to an existing cluster	585
15.3.1 Using SMIT	586
15.3.2 Using the SMUI to add an asynchronous GLVM configuration.....	589
15.4 GLVM cluster configuration.....	595
15.4.1 Topology	595
15.4.2 Resource group	596
15.4.3 Application controller.....	597
15.4.4 Application monitor	598
15.4.5 File collection	598
15.4.6 RPV servers and clients	599
15.4.7 GMVG attributes	600
15.4.8 Mirror pools.....	600

15.4.9 Logical volumes mirrored	601
15.4.10 AIO cache logical volumes	602
15.4.11 Split and merge policy.....	603
15.5 Using the command-line interface.....	603
Part 5. Appendixes	607
Appendix A. Paper planning worksheets	609
Internet Protocol network planning worksheets	610
Internet Protocol network interface worksheet.....	611
Fibre Channel disks worksheets	612
Shared volume group and file system worksheet	613
NFS-exported file system or directory worksheet.....	614
Application worksheet	615
Application server worksheet.....	616
Application monitor worksheet (custom)	616
Resource group worksheet	617
Cluster events worksheet.....	618
Cluster file collections worksheet.....	619
Appendix B. Cluster Test Tool log.....	621
Sample output from the Cluster Test Tool log	622
Abbreviations and acronyms	641
Related publications	643
IBM Redbooks	643
Online resources	643
Help from IBM	644

Notices

This information was developed for products and services offered in the US. This material might be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

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 grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing, IBM Corporation, North Castle Drive, MD-NC119, Armonk, NY 10504-1785, US

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 jurisdictions 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 websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

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

The performance data and client examples cited are presented for illustrative purposes only. Actual performance results may vary depending on specific configurations and operating conditions.

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.

Statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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 actual people or business enterprises 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. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at <http://www.ibm.com/legal/copytrade.shtml>

The following terms are trademarks or registered trademarks of International Business Machines Corporation, and might also be trademarks or registered trademarks in other countries.

AIX®	IBM FlashSystem®	pureScale®
Cognos®	IBM Security®	Redbooks®
Db2®	IBM Spectrum®	Redbooks (logo)  ®
DS8000®	IBM Z®	Storwize®
FileNet®	Power8®	SystemMirror®
HyperSwap®	Power9®	Tivoli®
IBM®	PowerHA®	WebSphere®
IBM Cloud®	PowerVM®	XIV®

The following terms are trademarks of other companies:

The registered trademark Linux® is used pursuant to a sublicense from the Linux Foundation, the exclusive licensee of Linus Torvalds, owner of the mark on a worldwide basis.

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

Red Hat, OpenShift, are trademarks or registered trademarks of Red Hat, Inc. or its subsidiaries in the United States and other countries.

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

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

Preface

This IBM Redbooks® publication can help you install, tailor, and configure the new IBM PowerHA® 7.2.7 product, and understand new and improved features such as migrations, cluster administration, and advanced topics like using Resource Optimized High Availability (ROHA); creating a cross-site Logical Volume Manager (LVM) stretched campus cluster; and running IBM PowerHA SystemMirror® in the IBM Power Virtual Server environment.

With this book, you can gain a broad understanding of the IBM PowerHA SystemMirror architecture. If you plan to install, migrate, or administer a high availability (HA) cluster, this book is right for you.

This book can help IBM AIX® professionals who seek a comprehensive and task-oriented guide for developing the knowledge and skills that are required for PowerHA cluster design, implementation, and daily system administration. It provides a combination of theory and practical experience.

This book is targeted toward technical professionals (consultants, technical support staff, IT architects, and IT specialists) who are responsible for providing HA solutions and support with the IBM PowerHA SystemMirror Standard on IBM Power servers.

Authors

This book was produced by a team of specialists from around the world working at IBM Redbooks, Austin Center.

Dino Quintero is a Systems Technology Architect with IBM Redbooks. He has 28 years of experience with IBM Power technologies and solutions. Dino shares his technical computing passion and expertise by leading teams developing technical content in the areas of enterprise continuous availability, enterprise systems management, high-performance computing (HPC), cloud computing, artificial intelligence (including machine and deep learning), and cognitive solutions. He is a Certified Open Group Distinguished Technical Specialist. Dino is formerly from the province of Chiriquí in Panama. Dino holds a Master of Computing Information Systems degree and a Bachelor of Science degree in Computer Science from Marist College.

Tim Simon is an IBM Redbooks Project Leader in Tulsa, Oklahoma, US. He has over 40 years of experience with IBM®, primarily in a technical sales role working with customers to help them create IBM solutions to solve their business problems. He holds a Bachelor of Science degree in Math from Towson University in Maryland. He has worked with many IBM products and has extensive experience creating customer solutions by using IBM Power, IBM Storage, and IBM Z® throughout his career.

Shawn Bodily is an eight-time IBM Champion for Power. He is a Senior IT Consultant for Clear Technologies in Dallas, Texas. He has 30 years of IBM AIX experience, and spent the last 26 years specializing in high availability and disaster recovery (HADR) solutions that are primarily focused around IBM PowerHA. He has written and presented extensively about HA and storage at technical conferences, webinars, and on site to customers. He is an IBM Redbooks Platinum Author who has co-authored over a dozen IBM Redbooks publications and IBM Redpaper publications. He is also the only author to work on every version of this book.

Vera Cruz is a consultant for IBM Power in IBM ASEAN Technology Lifecycle Services. She has 28 years of IT experience doing implementation, performance management, high availability (HA) and risk assessment, and security assessment for IBM AIX and IBM Power across diverse industries, including banking, manufacturing, retail, and government institutions. She has been with IBM for 8 years. Before joining IBM, she worked for various IBM Business Partners in the Philippines and Singapore by working as a Tech Support Specialist and Systems Engineer for IBM AIX and IBM Power. She holds a degree in Computer Engineering from the Cebu Institute of Technology University in Cebu, Philippines.

Sachin P. Deshmukh is the Global Power and AIX Platform Lead for Kyndryl, and is based in the US. His areas of expertise include IBM AIX operating system provisioning and support, IBM PowerHA, virtualization, and the IBM Cloud® platform. He provides guidance, oversight, and assistance to global delivery teams that support Kyndryl accounts. As a member of the Critical Response Team, he works on major incidents and high severity issues. He participates in proactive Technical Health Checks and Service Management Reviews. He interacts with automation, design, procurement, architecture, and support teams for setting delivery standards and creating various best practices documentation. He creates and maintains the IBM AIX Security Technical Specifications for Kyndryl. He is also certified on various other platforms such as Amazon Web Services (AWS) Solutions Architect (Associate), AWS Cloud Practitioner, and Red Hat Certified System Administrator. Before moving to Kyndryl in 2021, he had been with IBM since 1999. He has been closely associated with IBM AIX and the IBM Power platform for close to 30 years.

Karim El Barkouky is a Senior IT Management Consultant who works in MEA - Technology Services- Lab Services. He worked in IBM Systems as an L2 remote support - global PowerHA SME in Cairo, Egypt. He has 8 years of experience in the industry with expertise in several implementations and consultancy tasks for various HA solutions, such as IBM PowerHA SystemMirror, IBM Spectrum® Scale, Cluster Aware AIX (CAA), Reliable Scalable Cluster Technology (RSCT), Geographic Logical Volume Manager (GLVM), VM Recovery Manager (VMRM), the SUSE Linux Enterprise Server - SUSE/HA extension, and container orchestrators, such as Red Hat OpenShift. He is a recognized trainer who has delivered various IBM AIX and HA training sessions across the MEA. He has experience in IBM Power servers and the Power software family, such as IBM PowerVM®, IBM PowerVC, IBM PowerSC, and Linux on Power.

Youssef Largou is the founding director of PowerM, a platinum IBM Business Partner in Morocco. He has 21 years of experience in systems, HPC, middleware, and hybrid cloud, including IBM Power, IBM Storage, IBM Spectrum, IBM WebSphere®, IBM Db2®, IBM Cognos®, IBM WebSphere Portal, IBM MQ, Enterprise Service Bus (ESB), IBM Cloud Paks, and Red Hat OpenShift. He has worked within numerous industries with many technologies. Youssef is an IBM Champion for 2020, 2021, and 2022 and an IBM Redbooks Gold Author. He designed many reference architectures. He has been an IBM Beacon Award Finalist in Storage, Software-Defined Storage, and LinuxONE five times. He holds an Engineer degree in Computer Science from the Ecole Nationale Supérieure des Mines de Rabat, and an Excecutif MBA from EMLyon.

Jean-Manuel Lenez has been a Presales Engineer since 1999 with IBM Switzerland. He specializes in UNIX, Power, IBM AIX, and IBM i server technologies, and associated products such as PowerVM, PowerHA, PowerSC, Linux on Power, and IBM Cloud. He is heavily involved in his presales mission, where he leads projects with major customers regarding various subjects, such as artificial intelligence, deep learning, SAP HANA, server consolidation, and HADR.

Vivek Shukla is a Presales Consultant for cloud, AI, and cognitive offerings in India. He is an IBM Certified L2 (Expert) Brand Technical Specialist. He has over 20 years of IT experience in Infrastructure Consulting, IBM AIX, and IBM Power servers and storage implementations. He also has hands-on experience with IBM Power servers, IBM AIX, and system software installations, request for proposal understandings, statement of work preparations, sizing, performance tuning, root cause analysis, disaster recovery (DR), and mitigation planning. He has written several Power FAQs and is the Worldwide Focal for Techline FAQs Flash. He holds a Master's degree in Information Technology from IASE University and a Bachelor's degree (BTech) in Electronics & Telecommunication Engineering from IETE, New Delhi. His area of expertise includes IBM Power Enterprise Pools, Red Hat OpenShift, IBM Cloud Paks, and hybrid cloud.

Kulwinder Singh is a Technical Support Professional with the IBM India Systems Development Lab, IBM India. He has over 25 years of experience in IT infrastructure management. He supports customers as IBM AIX L2 development support for IBM AIX, PowerHA, and IBM VMRM HADR on Power. He holds Bachelor of Computer Application degree from St. Peter's University. His areas of expertise include IBM AIX, HA, and dynamic node priority (DNP) solutions, IBM Spectrum Protect storage, and storage area networks (SANs).

The following group of individuals were part of the residency that created the updates to this Redbooks publication and two additional documents. Thanks for their support in the residency.

Felipe Bessa is an IBM Brand Technical Specialist and Partner Technical Advocate on IBM Power in Brazil.

Carlos Jorge Cabanas Aguero is a Consultant with IBM Technology Lifecycle Services in Argentina.

Dishant Doriwala is a Senior Staff Software engineer, and the Test Lead for the VMRM for HADR product. He works in IBM Systems Development Labs, Hyderabad, India.

Alexander Ducut is a Technical Sales Manager with IBM in the Philippines.

Ash Giddings is based in the UK and is a Product Manager for Maxava.

Santosh S Joshi is a Senior Staff Software Engineer in IBM India Systems Development Lab, IBM India.

Juan Prada is an IBM i Senior System Administrator in Madrid, Spain.

Antony Steel is a senior technical staff member working with IBM Australia.

Yadong Yang is an IBM IT Management Consultant on IBM Power. He works for IBM Technology Services in the US.

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

Jeff Boleman, Principal Product Manager, IBM Power Virtual Server Infrastructure as a Service (IaaS)

IBM Systems, Vera Cruz, CA

Ramya Bommineni, IBM - VMRM Development Partner

IBM Systems, IBM India

Uma Maheswara Rao Chandolu, Director - PowerHA IBM AIX and VM Recovery Manager HADR

IBM Systems, IBM India

Jes Kiran Chittigala, HADR Architect for Power VMRM, Master Inventor

IBM India System Development Labs

Steven Finnes, IBM Power HA Product Offering Manager

IBM Systems, Rochester, MN

Abhilash Kadivendi, IBM - VMRM Development Partner

India

Adhish Kapoor, VMRM Developer

IBM India System Development Labs

Brian Nordland, IBM i High Availability

IBM Systems, US

Pandi Jai Sree, IBM - VMRM Development Partner

India

Srikanth Thannerru, Advisory Software Engineer, VMRM

IBM India System Development Labs

Douglas Yakesch, Power User Technologies Build Tools Team Lead

IBM Systems, Austin, TX

Vijay Yalamuri

IBM India

Tom Weaver, PowerHA, CAA - IBM AIX Development Support

IBM Systems Technology Lifecycle Services, Austin, TX

Scot Stansell, IBM AIX Technical Specialist - PowerHA Team

IBM Systems Technology Lifecycle Services, Coppell, TX

Now you can become a published author, too!

Here is an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an IBM Redbooks residency project and help write a book in your area of expertise, while honing your experience using leading-edge technologies. Your efforts will help to increase product acceptance and customer satisfaction, as you expand your network of technical contacts and relationships. Residencies run from two to six weeks in length, and you can participate either in person or as a remote resident working from your home base.

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 Redbooks form found at:

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbooks@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, IBM Redbooks
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- ▶ Find us on LinkedIn:

<http://www.linkedin.com/groups?home=&gid=2130806>

- ▶ Explore new Redbooks publications, residencies, and workshops with the IBM Redbooks weekly newsletter:

<https://www.redbooks.ibm.com/Redbooks.nsf/subscribe?OpenForm>

- ▶ Stay current on recent Redbooks publications with RSS Feeds:

<http://www.redbooks.ibm.com/rss.html>

Summary of changes

This section describes the technical changes that were made in this edition of the book and in previous editions. This edition might also include minor corrections and editorial changes that are not identified.

Summary of Changes
for SG24-7739-02
for IBM PowerHA SystemMirror for AIX Cookbook
as created or updated on January 30, 2024.

January 2024, Third Edition

This revision includes the following new and changed information.

New information

- ▶ Includes information about the recent IBM PowerHA SystemMirror for AIX 7.2.7.
- ▶ Added Chapter 13, “Cross-Site Logical Volume Manager stretched campus cluster” on page 525.
- ▶ Added Chapter 14, “IBM PowerHA and IBM Power Virtual Server” on page 563.
- ▶ Added Chapter 15, “Geographic Logical Volume Manager wizard” on page 575.

Changed information

- ▶ Removed chapter on workload partitions (WPARs) because they are no longer supported.
- ▶ Updates were made to several of the chapters to incorporate the latest improvements to IBM PowerHA SystemMirror for AIX.



Part 1

Introduction

In Part 1, we provide an overview of PowerHA and describe the PowerHA components as part of a successful implementation.

Because PowerHA is a mature product, we consider it important to present some of the recent PowerHA history, which can help in planning future actions, such as migrating existing configurations to the latest version and by using the new features in PowerHA.

We also introduce the basic PowerHA management concepts, with best practices and considerations to ease the system administrator's job.

This part contains the following chapters:

- ▶ Chapter 1, "Introducing IBM PowerHA SystemMirror for AIX" on page 3
- ▶ Chapter 2, "High availability components" on page 27



Introducing IBM PowerHA SystemMirror for AIX

This chapter contains the following topics:

- ▶ What is PowerHA SystemMirror for AIX
- ▶ Availability solutions: An overview
- ▶ History and evolution
- ▶ High availability terminology and concepts
- ▶ Fault-tolerant versus high availability
- ▶ Software planning
- ▶ PowerHA software installation

1.1 What is PowerHA SystemMirror for AIX

The IBM PowerHA SystemMirror software provides a low-cost commercial computing environment that ensures quick recovery of mission-critical applications from hardware and software failures.

With the PowerHA SystemMirror software, critical resources remain available. For example, a PowerHA SystemMirror cluster can run a database server program that services client applications. The clients send queries to the server program that responds to their requests by accessing a database that is stored on a shared external disk.

This high availability (HA) system combines custom software with industry-standard hardware to minimize downtime by quickly restoring services when a system, component, or application fails. Although not instantaneous, the restoration of service is rapid, usually within 30 - 300 seconds.

In a PowerHA SystemMirror cluster, to ensure the availability of these applications, the applications are put under PowerHA SystemMirror control. PowerHA SystemMirror takes measures to ensure that the applications remain available to client processes even if a component in a cluster fails. To ensure availability if there is a component failure, PowerHA SystemMirror moves the application (along with resources that ensure access to the application) to another node in the cluster.

HA is sometimes confused with simple hardware availability. Fault-tolerant, redundant systems (such as redundant array of independent disks (RAID)) and dynamic switching technologies (such as dynamic logical partition (DLPAR)) provide the recovery of certain hardware failures, but do not provide the full scope of error detection and recovery that is required to keep a complex application highly available.

A modern, complex application requires access to all these components:

- ▶ Nodes (CPU and memory)
- ▶ Network interfaces (including external devices in the network topology)
- ▶ Disk or storage devices

Recent surveys of the causes of downtime show that actual hardware failures account for only a small percentage of unplanned outages. Other contributing factors include the following items:

- ▶ Operator errors
- ▶ Environmental problems
- ▶ Application and operating system errors

Reliable and recoverable hardware cannot protect against failures of all these different aspects of the configuration. Keeping these varied elements and the application highly available requires the following aspects:

- ▶ Thorough and complete planning of the physical and logical procedures for access and operation of the resources on which the application depends. These procedures help to avoid failures in the first place.
- ▶ A monitoring and recovery package that automates the detection and recovery from errors.
- ▶ A well-controlled process for maintaining the hardware and software aspects of the cluster configuration while keeping the application available.

PowerHA features

IBM PowerHA technology positions you to deploy an HA solution that addresses storage and HA requirements with one integrated configuration and a simplified user interface (UI).

IBM PowerHA SystemMirror for AIX is available in either Standard Edition or Enterprise Edition. The Standard Edition is used for local (single site), or close proximity cross-site or campus-style clusters. The Enterprise Edition is more synonymous with disaster recovery (DR) by using some form of data replication across diverse sites.

IBM PowerHA SystemMirror has the following important features:

- ▶ Host-based replication

Perform failover operations to private or public cloud configurations with IBM PowerHA with Geographic Logical Volume Manager (GLVM).

- ▶ Automation

Manage your cluster from a single interface. Smart assists allow for ready for use availability setup and application management. Automatic recovery actions occur in the event of a failure detection.

- ▶ Clustering technology

Orchestrate cluster operations for either local shared storage configurations or multi-site configurations.

- ▶ Economic value

PowerHA for on-premises deployments is licensed per processor core, with a one-time charge. The first year of software maintenance is included.

- ▶ Highly autonomous

PowerHA requires minimal administrative involvement. It replaces logical replication for more reliable, efficient, and easy-to-use solutions.

- ▶ Integrated IBM Storage Area Network (SAN) storage

PowerHA Enterprise Edition incorporates IBM DS8000®, IBM XIV®, or IBM Spectrum Virtualized Systems Storage (SAN Volume Controller (SVC), IBM Storwize® V5000, IBM Storwize, V7000, IBM FlashSystem® 5000, IBM FlashSystem 7000, or IBM FlashSystem 9000) into a high availability and disaster recovery (HADR) cluster.

- ▶ Integrated non IBM SAN storage

PowerHA Enterprise Edition incorporates the replication facilities that are available in Dell EMC storage and Hitachi storage.

Benefits of PowerHA SystemMirror

PowerHA SystemMirror has the following benefits:

- ▶ The PowerHA SystemMirror planning process and documentation include tips and advice on the best practices for installing and maintaining a highly available PowerHA SystemMirror cluster.
- ▶ When the cluster is operational, PowerHA SystemMirror provides automated monitoring and recovery for all the resources on which the application depends.
- ▶ PowerHA SystemMirror provides a full set of tools for maintaining the cluster while keeping the application available to clients.
- ▶ Set up a basic two-node cluster by using the typical initial cluster configuration System Management Interface Tool (SMIT) path or the application configuration assistants (Smart Assists).

- ▶ Test your PowerHA SystemMirror configuration by using the Cluster Test Tool. You can evaluate how a cluster behaves under a set of specified circumstances, such as when a node or network become inaccessible.
- ▶ Ensure the HA of applications by eliminating single points of failure (SPOFs) in a PowerHA SystemMirror environment.
- ▶ Leverage HA features that are available in AIX.
- ▶ Manage how a cluster handles component failures.
- ▶ Secure cluster communications.
- ▶ Monitor PowerHA SystemMirror components and diagnose problems that might occur.

1.1.1 High availability

In today's complex environments, providing continuous service for applications is a key component of a successful IT implementation. HA is one of the components that contributes to providing continuous service for the application clients by masking or eliminating both planned and unplanned systems and application downtime. A HA solution ensures that the failure of any component of the solution, either hardware, software, or system management, does not cause the application and its data to become permanently unavailable to the user.

HA solutions should eliminate SPOFs through appropriate design, planning, selection of hardware, configuration of software, control of applications, a carefully controlled environment, and change management discipline.

In short, we can define *HA* as the process of ensuring (by using duplicated or shared hardware resources that are managed by a specialized software component) that an application is available for use.

1.1.2 Cluster multiprocessing

In addition to HA, PowerHA also provides the multiprocessing component. The multiprocessing capability comes from the fact that in a cluster there are multiple hardware and software resources that are managed by PowerHA to provide complex application functions and better resource utilization.

A short definition for cluster multiprocessing might be multiple applications running over several nodes with shared or concurrent access to the data.

The cluster multiprocessing component depends on the application capabilities and system implementation to efficiently use all resources that are available in a multi-node (cluster) environment. These items must be implemented starting with the cluster planning and design phase.

PowerHA is only one of the components of your HA environment. It provides monitoring and automated response to issues that occur in an ecosystem of increasingly reliable operating systems, hot-swappable hardware, and increasingly resilient applications.

A HA solution that is based on PowerHA provides automated failure detection, diagnosis, application recovery, and node reintegration. If your application is capable of parallel processing, PowerHA can provide concurrent access to the data for those applications, which allows excellent horizontal and vertical scalability (with the addition of the DLPAR management capabilities).

PowerHA depends on Reliable Scalable Cluster Technology (RSCT). RSCT is a set of low-level operating system components that support the implementation of clustering technologies such as PowerHA and General Parallel File System (GPFS). RSCT is distributed with AIX and on the current AIX release, AIX 7.3, RSCT is at Version 3.3.0.0. After installing the PowerHA and Cluster Aware AIX (CAA) file sets, The RSCT Topology Services subsystem is deactivated and all its functions are performed by CAA.

PowerHA also provides DR functions such as cross-site mirroring, IBM HyperSwap®, and GLVM. These cross-site clustering methods support PowerHA functions between two geographic sites. Various other methods exist for replicating the data to remote sites on both IBM and non IBM storage. For more information about these options and supported configurations, see [IBM PowerHA SystemMirror for AIX Enterprise Edition Version 7.2: Storage-based high availability and disaster recovery](#).

1.2 Availability solutions: An overview

Many solutions can provide a wide range of HA options. Table 1-1 lists various types of availability solutions and their characteristics.

Table 1-1 Types of availability solutions

Solution	Downtime	Data availability	Observations
Stand-alone	Days	From last backup	Basic hardware and software costs
Enhanced stand-alone	Hours	Until last transaction	Double the basic hardware cost
HA clusters	Seconds	Until last transaction	Double hardware and more services, and more costs
Fault-tolerant computing	Zero downtime	No loss of data	Specialized hardware and software, and expensive

HA solutions, in general, offer the following benefits:

- ▶ Standard hardware and networking components (can be used with the existing hardware).
- ▶ Works with nearly all applications.
- ▶ Works with a wide range of disks and network types.
- ▶ Excellent availability at reasonable cost.

The highly available solution for IBM Power offers distinct benefits:

- ▶ Proven solution (more than 25 years of product development)
- ▶ Using “off the shelf” hardware components
- ▶ Proven commitment for supporting customers
- ▶ IP version 6 (IPv6) support for both internal and external cluster communication
- ▶ Smart Assist technology enabling HA support for all prominent applications
- ▶ Flexibility (virtually any application running on a stand-alone AIX system can be protected with PowerHA)

When you plan to implement a PowerHA solution, consider the following aspects:

- ▶ Thorough HA design and detailed planning from end to end
- ▶ Elimination of SPOFs
- ▶ Selection of appropriate hardware
- ▶ Correct implementation (do *not* take “shortcuts”)

- ▶ Disciplined system administration practices and change control
- ▶ Documented operational procedures
- ▶ Comprehensive test plan and thorough testing

A typical PowerHA environment is shown in Figure 1-1. Both IP heartbeat networks and non-IP network heartbeating are performed through the cluster repository disk.

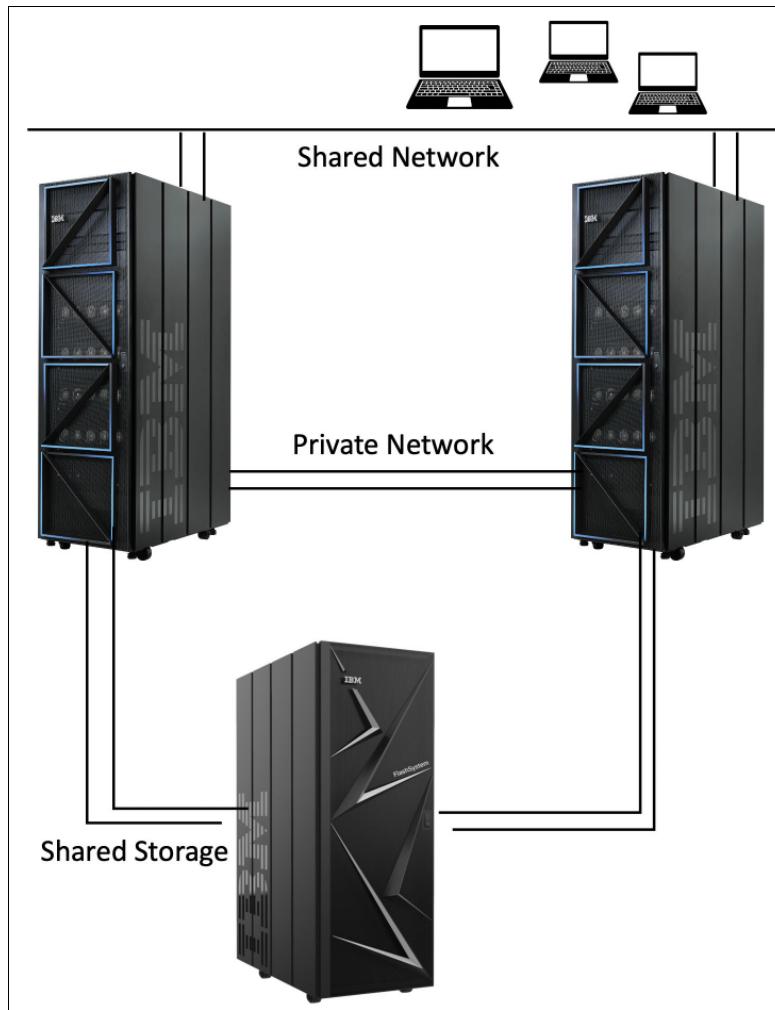


Figure 1-1 PowerHA cluster

1.2.1 Downtime

Downtime is the period when an application or service is unavailable to its clients. Downtime can be classified in two categories:

- ▶ Planned:
 - Hardware upgrades
 - Hardware or software repair or replacement
 - Software updates or upgrades
 - Backups (offline backups)
 - Testing (periodic testing is required for cluster validation)
 - Development

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

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

1.2.2 Single point of failure

A SPOF is any individual component that is integrated in a cluster and that, if there is a failure, renders the application unavailable for users.

Good design can remove SPOFs in the cluster's nodes, storage, and networks. PowerHA manages these components, and also the resources that are required by the application (including the application start and stop scripts).

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

To avoid SPOFs, you need these items:

- ▶ Redundant servers
- ▶ Redundant network paths
- ▶ Redundant storage (data) paths
- ▶ Redundant (mirrored, RAID) storage
- ▶ Monitoring of components
- ▶ Failure detection and diagnosis
- ▶ Automated application failover
- ▶ Automated resource reintegration

A good design can avoid SPOFs, and PowerHA can manage the availability of the application when downtimes occur. Table 1-2 lists cluster objects that can result in loss of availability of the application if they fail. Each cluster object can be a physical or virtual component.

Table 1-2 Single points of failure

Cluster object	Single point of failure is eliminated by
Node (servers)	Multiple nodes
Power supply	Multiple circuits, power supplies, or uninterruptible power supply (UPS)
Network adapter	Redundant network adapters
Network	Multiple networks that are connected to each node, and redundant network paths with independent hardware between each node and the clients
TCP/IP subsystem	Using non-IP networks to connect each node to its neighbor in a ring
I/O adapter	Redundant I/O adapters
Controllers	User-redundant controllers

Cluster object	Single point of failure is eliminated by
Storage	Redundant hardware, enclosures, disk mirroring or RAID technology, or redundant data paths
Application	Configuring application monitoring and backup nodes to acquire the application engine and data
Sites	Using more than one site for DR
Resource groups (RGs)	Using RGs to control all resources that are required by an application

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

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

1.3 History and evolution

IBM High Availability Cluster Multiprocessing (HACMP) development started in 1990 to provide HA solutions for applications running on IBM RS/6000 servers. We do not provide information about the early releases, which are no longer supported or were not in use at the time of writing. Instead, we provide highlights about the most recent versions.

HACMP was originally designed as a stand-alone product (known as HACMP classic). After the IBM HA infrastructure, which is known as RSCT, became available, HACMP adopted the technology and became HACMP Enhanced Scalability (HACMP/ES), which provided performance and functional advantages over the classic version. Later, HACMP terminology was renamed to PowerHA starting at Version 5.5, and then to PowerHA SystemMirror with Version 6.1.

Starting with the PowerHA SystemMirror 7.1, the CAA feature of the operating system is used to configure, verify, and monitor the cluster services. This major change improved the reliability of PowerHA because the cluster service functions were moved to kernel space, rather than running in user space. CAA was introduced in AIX 6.1 Technology Level (TL) 6.

At the time of writing, the current release is PowerHA SystemMirror 7.2.7.

1.3.1 PowerHA SystemMirror 7.2.4

Released in December 2019, PowerHA SystemMirror 7.2.4 introduced the following enhancements:

- ▶ The Cross-Cluster Verification (CCV) utility

The CCV utility checks the cluster configuration on all nodes within the same cluster. In PowerHA SystemMirror 7.2.4, or later, you can use the CCV utility to compare the configuration of two different clusters and view differences in the installed software and other attributes of clusters. For more information, see the **c1mgr compare** command.

- ▶ Cluster snapshot and **c1mgr** command enhancements

All cluster snapshot functions are now available through the **c1mgr** command. The **c1snapshot** command is no longer supported. You can use new cluster snapshot functions to view information within a snapshot and compare snapshot files.

- ▶ Availability metrics enhancements

The availability metrics feature is enhanced with extra reporting capabilities for analysis. For more information, see the **c1_availability** command.

- ▶ Support for up to 256 RGs

The support for the maximum number of RGs increased from 64 to 256. For more information, see [PowerHA SystemMirror maximum limits](#).

- ▶ IBM Db2 multiple database support

You can configure and independently monitor multiple databases for the same Db2 server. More databases can be configured manually or through PowerHA SystemMirror Smart Assist for Db2.

- ▶ WebSphere MQ listener support

PowerHA SystemMirror supports multiple listeners for the same WebSphere MQ application. More listeners can be configured either manually or by using a discovery process.

- ▶ Support for CAA 4-K disk block size for repository disks

The CAA subsystem now supports repository disks that use a 4-K block size. PowerHA SystemMirror checks for the supported repository disk sizes when you configure the repository.

- ▶ Support for internet Small Computer System Interface (iSCSI) disks as cluster repository disks

In PowerHA SystemMirror 7.2.4, or later, you can configure iSCSI disks as cluster repository disks.

- ▶ Enhanced PowerHA SystemMirror role-based access control (RBAC) to include new commands and roles

Federated Security is enhanced to enable PowerHA SystemMirror commands that are not already supported for RBAC.

Note: RBAC enablement is *not* recommended for PowerHA SystemMirror Enterprise Edition.

- ▶ Simplified SVC Peer-to-Peer Remote Copy (PPRC) configuration tasks by using automatic discovery
Configuring SVC Peer-to-Peer Remote Copy (PPRC) mirrors was a manual process in earlier releases. In PowerHA SystemMirror Enterprise Edition 7.2.4, or later, you can automatically configure SVC PPRC mirrors by using a discovery facility.
- ▶ Non-root SVC mirror management
In PowerHA SystemMirror Enterprise Edition 7.2.4 or later, you can now specify alternative credentials instead of the root user credentials, which PowerHA SystemMirror uses to access the SVC storage management functions.
- ▶ PowerHA SystemMirror User Interface (SMUI) enhancements:
 - Enhanced visual representation of availability metrics and the newly available resource-centric views. You can also view the CPU and memory usage statistics. You can troubleshoot performance issues and predict application downtime and application uptime, and optimize your RGs and applications.
 - Create and manage logical volumes (LVs), mirror pools, and RG dependencies.
 - Generate the snapshot report and view the contents of a snapshot before you restore a snapshot.
 - Enhanced cluster reports to display details about repository disks and methods.
 - Non-root support for cloning a cluster from snapshots.
 - Configure a PowerHA SMUI server to be highly available by using the High Availability wizard in the PowerHA SMUI.
 - Import multiple clusters by using the Add Multiple Clusters wizard.
 - Enhanced activity log to display the activity ID, start time, and end time of the activity, and the duration of the activity. You can also view details such as the number of successful login attempts and failed login attempts, the number of new activities, and the number of activities that are not viewed.
 - Enhanced security features with options to disable anonymous login and global access.
 - Automatic download and installation of the remaining files that are required to complete the PowerHA SMUI installation process from the IBM website by using the `smuiinst.ksh` command.

1.3.2 PowerHA SystemMirror 7.2.5

Released in December 2020, PowerHA SystemMirror 7.2.5 introduced the following enhancements:

- ▶ Support for the `CRIT_DAEMON_RESTART_GRACE_PERIOD` tunable

The RSCT subsystem consists of multiple daemons that provide various functions to PowerHA SystemMirror. Some RSCT daemons are marked as critical because PowerHA SystemMirror depends on these daemons to provide HA. When RSCT critical daemons are not available, PowerHA SystemMirror nodes are halted to avoid corruptions.

If there is a system failure, the Resource Monitoring and Control (RMC) subsystem and IBM.ConfigRM daemon restart automatically. To save the downtime that is required to halt and restart the node, you can use the `CRIT_DAEMON_RESTART_GRACE_PERIOD` tunable. RSCT waits for the specified grace period to allow the RMC subsystem and IBM.ConfigRM daemons to restart without halting the node.

You can set the value of the **CRIT_DAEMON_RESTART_GRACE_PERIOD** tunable both at the cluster level and node level of PowerHA SystemMirror to 0 - 240 seconds. For more information, see [Configuring the Critical Daemon Restart Period tunable](#).

- ▶ Skip resource handling for Unmanaged resource groups option

In PowerHA SystemMirror 7.2.5 or later, you can skip cluster resource processing by using the **SKIP_EVENT_PROCESSING_MANAGE_MODE** tunable after restarting the cluster services that are stopped by using the Unmanaged Resource Groups option. During the cluster restart operation, the **SKIP_EVENT_PROCESSING_MANAGE_MODE** tunable restores the state of the cluster RG to the previous state without disturbing its current state. After you enable the **SKIP_EVENT_PROCESSING_MANAGE_MODE** tunable in the hacmp.out log file, the new forced option of the node_up event directs the node_up event to skip cluster resources processing. For more information, see [Starting cluster services on a node with a resource group in the UNMANAGED state](#).

- ▶ Cloud tiebreaker option

PowerHA SystemMirror supports different tiebreaker mechanisms that can be used to determine cluster behavior during split and merge scenarios. The disk and Network File System (NFS) tiebreaker options use storage outside of the cluster domain to read/write files. This approach determines the winning and losing sides of a partition. PowerHA SystemMirror 7.2.5 introduces a new Cloud option that uses cloud-based storage for the same purpose. This feature supports IBM and Amazon Web Services (AWS) cloud services. For more information, see [Configuring split and merge policies](#).

- ▶ Enhancements to the **c1verify** option for *netmon.cf* file content validation

PowerHA SystemMirror includes a robust verification mechanism, which checks multiple aspects of the cluster and AIX configuration for proper settings and consistency. In PowerHA SystemMirror 7.2.5 or later, the cluster verification operation includes checking an optional *netmon.cf* file. This verification process avoids false network events. The cluster verification process now verifies the content and consistency of the *netmon.cf* file across the cluster nodes.

- ▶ Oracle migration support

During Oracle database migration, a customer might change the Oracle home directory. In such cases, PowerHA SystemMirror Smart Assist for Oracle must be configured with the new Oracle home directory. In PowerHA SystemMirror 7.2.5 or later, a new option is introduced that automatically updates PowerHA SystemMirror Smart Assist for Oracle to use the new Oracle home directory.

- ▶ Oracle temporary file removal

During the discovery, start, stop, and monitoring operations of PowerHA SystemMirror Smart Assist for Oracle, temporary files are created in the /tmp directory. However, system failure might occur due to an increase in the size of the /tmp directory and low availability of space. In PowerHA SystemMirror 7.2.5 or later, you can use the PowerHA SystemMirror default log directory for intermediate file operations.

- ▶ SMIT enhancements

After changing the PowerHA SystemMirror configuration, the cluster must be verified and synchronized to implement the updates. If the updates are made by using the SMIT interface, the option to verify and synchronize was not available for some SMIT panels. In PowerHA SystemMirror 7.2.5 or later, the verify and synchronize options are available on most SMIT panels.

- ▶ CCV utility enhancements

The cluster verification utility checks the cluster configuration on all nodes within the same cluster. In PowerHA SystemMirror 7.2.5 or later, you can use the CCV utility to compare CAA tunables between two different clusters.

- ▶ Dell EMC Symmetrix Remote Data Facility (SRDF)/Metro SmartDR configuration

PowerHA SystemMirror Enterprise Edition 7.2.5 Service Pack (SP) 1 added an EMC SRDF/Metro SmartDR configuration, which is a two-region HADR framework that integrates SRDF/Metro and SRDF/Async replicated resources.

- ▶ GLVM Configuration Assistant enhancements

In PowerHA SystemMirror Enterprise Edition 7.2.5, the GLVM Configuration Assistant is enhanced with new features that converts an existing volume group (VG) to GLVM-based VG and updates an existing RG to include GLVM resources. Also, the delete or rollback feature that is used for removing resources and configurations by using the GLVM Configuration Assistant is improved. For more information, see [Geographic Logical Volume Manager](#).

- ▶ PowerHA SMUI enhancements:

- GLVM asynchronous configuration is now supported in the PowerHA SMUI.
- The Operation Center Support (OCS) feature is used to configure the PowerHA SMUI for long-term use, with visual and audio alerts.
- The CCV feature in PowerHA SystemMirror compares two cluster configurations to show differences in the PowerHA SystemMirror configuration, file sets, interim fixes, and more. CCV compares attributes that are collected from one or two clusters by using the current cluster configuration or snapshots.
- All the supported application monitor configuration parameters are now available in the PowerHA SMUI.
- The PowerHA SMUI server's hostname is logged in the `c1mgr` log file for cluster changes that are initiated from the PowerHA SMUI. This feature provides complete GUI-to-cluster and cluster-to-GUI activity auditing.
- PowerHA SystemMirror can create a backup communication method for the PowerHA SMUI server by configuring a Secure Shell (SSH) key while adding a cluster to the PowerHA SMUI.
- The PowerHA SMUI removed the ability of the `hostname` command to determine how to communicate with nodes. The PowerHA SMUI server now collects either a public boot IP address or persistent IP address from each cluster node and uses that IP address to communicate with that node.
- Importing multiple clusters was enhanced to provide a progress indicator.
- Improved the clarity of events that are displayed in the PowerHA SMUI by adding a start and complete indicator for two-phase events.

1.3.3 PowerHA SystemMirror 7.2.6

Released in December 2021, PowerHA SystemMirror 7.2.6 introduced the following enhancements:

- ▶ Support for LV encryption

Starting with PowerHA SystemMirror 7.2.6 and IBM AIX 7.3, the Logical Volume Manager (LVM) enables data encryption for data VGs that are configured in the PowerHA SystemMirror environment.

PowerHA SystemMirror 7.2.6 or later supports platform keystore (PKS) and key server authentication methods to enable the LV encryption. For more information about encrypting LVs, see [Encrypting logical volumes](#).

- ▶ EMC SRDF/Metro SmartDR configuration

PowerHA SystemMirror Enterprise Edition 7.2.6 provides EMC SRDF/Metro SmartDR configuration, which is a two-region HADR framework that integrates SRDF/Metro and SRDF/Async replicated resources.

- ▶ GLVM Configuration wizard enhancements

PowerHA SystemMirror 7.2.6 Enterprise Edition provides the following GLVM Configuration wizard enhancements:

- You can dynamically update the cache size of the LV.
- The GLVM Configuration wizard collects the Remote Physical Volume (RPV) mirroring statistics and stores the information in the JavaScript Object Notation (JSON) format. You can use the RPV statistics by using any tool that can display the JSON format. The RPV statistics data is automatically sent to the PowerHA SMUI, which displays it in a graphical format.
- Added GLVM policies such as compression, io_grp_latency, and no_parallel_ls.

- ▶ Cloud reliability, availability, and serviceability (RAS) enhancements

For Cloud Backup Management (CBM), RAS was enhanced with an improved logging process.

- ▶ Standard to linked cluster conversion

In PowerHA SystemMirror 7.2.6, an existing standard cluster can be dynamically changed to a linked cluster by using the `c1mgr` command. This feature is useful for converting a standard cluster to an IBM Power Virtual Server cloud cluster. For more information, see [Converting a standard cluster to a linked cluster](#).

- ▶ PowerHA SMUI:

- GLVM historical charts

GLVM historical charts provide information about cache utilization data in a graphical format. You can view the historical data about cache utilization, network utilization, and disk utilization for the specified date range and different time intervals (minute, hour, day, week, and month).

- Asynchronous cache size in GLVM

You can view and modify the asynchronous cache size that is set during GLVM configuration.

- GLVM policies

GLVM tunables that are used to configure the mirror pool in the physical volume (PV) at the remote site. In PowerHA SystemMirror 7.2.6 or later, you can set the following GLVM tunable attributes:

- Compression
- I/O group latency
- Number of parallel LV

- Multi-factor authentication

In PowerHA SystemMirror 7.2.6 or later, multi-factor authentication is enabled for non-root GUI users. PowerHA SMUI uses an IBM Security® Verify Access account for multi-factor authentication. Multi-factor authentication can be performed by using either mobile authentication or email authentication.

The mobile authentication method uses login credentials (a username and password). For the email authentication method, you must select either a one-time password (OTP) that is delivered through an email or select the Short Message Service (SMS).

Note: Create an IBM Security Verify application account to use the multi-factor authentication features.

- CBM

The CBM feature allows you to create, view, edit, and delete backup profiles of an RG on the cloud. You back up VG data and store it on cloud services. You can back up VG data at IBM and Amazon cloud services.

- The Multiple CCV feature

The Multiple CCV feature can be used to compare one primary cluster with multiple clusters in a one-step procedure. You can filter the comparison result that displays differences and similarities between different clusters. You can select many attributes for cluster comparison.

1.3.4 PowerHA SystemMirror 7.2.7

Released in December 2022, PowerHA SystemMirror 7.2.7 introduced the following enhancements:

- ▶ Resource Optimized High Availability (ROHA) in IBM Power Virtual Server Cloud

PowerHA 7.2.7 adds support for ROHA by using IBM Power Virtual Server Cloud. Both SMIT and the `c1mgr` CLI, along with cluster verification, are updated to accommodate this support. This support also includes Hardware Management Console (HMC) 10.

- ▶ CBM and Cloud Backup and Restore (CBR)

Automated backup and recovery to and from a public cloud as an alternative to physical media. Hybrid cloud automated backup and recovery through PowerHA for IBM AIX and IBM FlashSystem storage eliminates the need for on-premises backup and recovery to on-premises physical media.

- ▶ Active node halt policy for public cloud

This function prevents partitioned cluster scenarios, which can cause data corruption.

- ▶ GLVM DR Sizing Tool

The DR Sizing Tool (`/usr/es/sbin/cluster/c1_survey`) analyzes the GLVM Async configuration. The tool is used to estimate network bandwidth and cache requirements for the Async GLVM networks that support GLVM traffic. The estimation is achieved by collecting the statistical information of the cluster configuration over a period and analyzing the information to recommend configuration changes for network bandwidth and cache size.

- ▶ Power Hypervisor Watchdog timer support

The Watchdog timer enables the hypervisor to turn off non-responsive VMs. This CAA-provided capability can be modified by using either SMIT or the `c1mgr` CLI.

- ▶ Three-site clustering with IBM DS8000 storage

PowerHA 7.2.7 Enterprise Edition and DS8000 enable HADR operations between three sites.

- ▶ French catalog message support

- ▶ Fix Central support

The PowerHA SMUI menu now shows the latest available SPs on Fix Central.

- ▶ Automatic expiration of on-screen notifications

1.4 High availability terminology and concepts

To understand the functions of PowerHA SystemMirror and to use it effectively, you must understand several important terms and concepts.

1.4.1 Terminology

The terminology that is used to describe the PowerHA configuration and operation continues to evolve. The following terms are used throughout this book:

Cluster	A loosely coupled collection of independent systems (nodes) or logical partitions (LPARs) that is organized into a network for sharing resources and communicating with each other.
	PowerHA defines relationships among cooperating systems where peer cluster nodes provide the services that are offered by a cluster node if that node is unable to do so. These individual nodes are together responsible for maintaining the functions of one or more applications if there is a failure of any cluster component.
Node	An IBM Power server (or LPAR) running AIX and PowerHA that is defined as part of a cluster. Each node has a collection of resources (disks, file systems, IP addresses, and applications) that can be transferred to another node in the cluster in case the node or a component fails.
Clients	A client is a system that can access the application running on the cluster nodes over a local area network (LAN). Clients run a client application that connects to the server (node) where the application runs.

1.4.2 Concepts

The basic concepts of PowerHA can be classified as follows:

Topology	Contains basic cluster components nodes, networks, communication interfaces, and communication adapters.
Resources	Logical components or entities that are being made highly available (for example, file systems, raw devices, service IP labels, and applications) by being moved from one node to another one. All resources that together form a highly available application or service are grouped in an RG.
	PowerHA keeps the RG highly available as a single entity that can be moved from node to node if a component or node fails. RGs can be available from a single node or (in the case of concurrent applications) available simultaneously from multiple nodes. A cluster can host more than one RG, thus allowing for the efficient usage of the cluster nodes.
Service IP label	A label that matches to a service IP address and is used for communication between clients and the node. A service IP label is part of an RG, which means that PowerHA can monitor it and keep it highly available.
IP Address Takeover	The process where an IP address is moved from one adapter to another adapter on the same logical network. This adapter can be on the same node, or another node in the cluster. If aliasing is used as the method of assigning addresses to adapters, then more than one address can be on a single adapter.
Resource takeover	The operation of transferring resources between nodes inside the cluster. If one component or node fails because of a hardware or operating system problem, its RGs are moved to another node.
Failover	The movement of an RG from one active node to another node (backup node) in response to a failure on that active node.
Fallback	The movement of an RG back from the backup node to the previous node when it becomes available. This movement is typically in response to the reintegration of the previously failed node.
Heartbeat packet	A packet that is sent between communication interfaces in the cluster, and is used by the various cluster daemons to monitor the state of the cluster components (nodes, networks, and adapters).
RSCT daemons	Two types of processes (topology and group services) that monitor the state of the cluster and each node. The cluster manager receives event information that is generated by these daemons and takes corresponding (response) actions if there is a failure.
Group leader	The node with the highest IP address as defined in one of the PowerHA networks (the first network that is available), which acts as the central repository for all topology and group data coming from the RSCT daemons concerning the state of the cluster.

Group leader backup	The node with the next highest IP address on the same arbitrarily chosen network, which acts as a backup for the group leader. It takes over the role of group leader if the group leader leaves the cluster.
Mayor	A node that is chosen by the RSCT group leader (the node with the next highest IP address after the group leader backup), if such exists, or it is the group leader backup itself. The mayor is responsible for informing other nodes of any changes in the cluster as determined by the group leader.

1.5 Fault-tolerant versus high availability

Based on the response time and response action to system-detected failures, clusters and systems can belong to one of the following classifications:

- ▶ Fault-tolerant systems
- ▶ HA systems

1.5.1 Fault-tolerant systems

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

All components, CPUs, memory, and disks have a special design and provide continuous service, even if one subcomponent fails. Only special software solutions can run on fault-tolerant hardware.

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

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

1.5.2 High availability systems

The systems that are configured for HA are a combination of hardware and software components that are configured to work together to ensure automated recovery if there is a failure, with a minimal acceptable downtime.

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

Therefore, eliminating all SPOFs in the environment is important. For example, if the machine has only one network interface (connection), provide a second network interface (connection) in the same node to take over in case the primary interface providing the service fails.

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

The PowerHA software provides the framework and a set of tools for integrating applications in a highly available system.

Applications that are integrated into a PowerHA cluster can require a fair amount of customization, possibly both at the application level and at the PowerHA and AIX platform levels. PowerHA is a flexible platform that allows integration of generic applications running on the AIX platform, providing for highly available systems at a reasonable cost.

Remember, PowerHA is not a fault-tolerant solution, and should never be misconstrued as one.

1.6 Software planning

While planning a PowerHA cluster, one of the most important steps is to choose the software levels to be running on the cluster nodes.

The decision factors in node software planning are as follows:

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

1.6.1 AIX level and related requirements

Before you install PowerHA, check the other software level requirements. Table 1-3 on page 21 shows the supported PowerHA and AIX levels at the time of writing.

Table 1-3 AIX level requirements

PowerHA version	AIX level	Recommended Authorized Program Analysis Report (APAR)	Minimum RSCT level
PowerHA v7.2.2	7100-04-02	IV90451, IJ04512, IJ04252, IJ07856, and IJ03871	3.2.1.10
	7100-05-00	IJ02843, IJ04266, IJ05079, IJ06703, IJ04252, IJ06118, and IJ03871	3.2.3.0
	7200-00-02	IV90451, IJ04252, IJ07856, and IJ03871	3.2.1.10
	7200-01-01	IV90485, IV91020, IJ04267, IJ04252 IJ06118, and IJ03871	3.2.2.0
	7200-02-00	IJ02843, IJ04268, IJ05079, IJ06703, IJ04252, IJ06118, and IJ03871	3.2.3.0
	7200-03-01		3.2.4.0
	7200-04-01		3.2.5.0
PowerHA v7.2.3	7100-04-06	IJ04512, IJ04252, IJ07856, and IJ03871	3.2.1.10
	7100-05-02	IJ02843, IJ04266, IJ05079, IJ06703, IJ04252, IJ06118, and IJ03871	3.2.3.0
	7200-00-06	IJ04252, IJ07856, and IJ03871	3.2.1.10
	7200-01-04	IJ04267, IJ04252, IJ06118, and IJ03871	3.2.2.0
	7200-02-02	IJ02843, IJ04268, IJ05079, IJ06703, IJ04252, IJ06118, and IJ03871	3.2.3.0
	7200-03-01		3.2.4.0
	7200-04-01		3.2.5.0
PowerHA v7.2.4	7100-04-08		3.2.1.10
	7100-05-05		3.2.3.0
	7200-01-06		3.2.2.0
	7200-02-04		3.2.3.0
	7200-03-03		3.2.4.0
	7200-04-01		3.2.5.0

PowerHA version	AIX level	Recommended Authorized Program Analysis Report (APAR)	Minimum RSCT level
PowerHA v7.2.5	7100-05-06		3.2.3.0
	7200-04-02		3.2.5.0
	7200-05-00		3.2.6.0
PowerHA v7.2.6	7100-05-09		3.2.3.0
	7200-01-06		3.2.2.0
	7200-02-06		3.2.3.0
	7200-03-07		3.2.4.0
	7200-04-04		3.2.5.0
	7200-05-03		3.2.6.0
	7300-00-00		3.3.0.0
PowerHA v7.2.7	7100-05-10		3.2.3.0
	7200-01-06		3.2.2.0
	7200-02-06		3.2.3.0
	7200-03-06		3.2.4.0
	7200-04-06		3.2.5.0
	7200-05-05		3.2.6.0
	7300-00-02		3.3.0.0
	7300-01-01		3.3.1.0

The current list of recommended SPs for PowerHA are at the [PowerHA AIX code Level Reference Table](#).

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

- ▶ bos.adt.lib
- ▶ bos.adt.libm
- ▶ bos.adt.syscalls
- ▶ bos.ahafs
- ▶ bos.cluster
- ▶ bos.clvm.enh
- ▶ bos.data
- ▶ bos.net.tcp.client
- ▶ bos.net.tcp.server
- ▶ bos.rte.SRC
- ▶ bos.rte.libc
- ▶ bos.rte.libcfg
- ▶ bos.rte.libcurl
- ▶ bos.rte.libpthreads
- ▶ bos.rte.lvm
- ▶ bos.rte.odm
- ▶ devices.common.IBM.storfwk.rte (optional, but required for sancomm)

Requirements for NFSv4

The cluster.es.nfs file set that is included with the PowerHA installation medium installs the NFSv4 support for PowerHA, along with an NFS Configuration Assistant. To install this file set, the following BOS NFS components must also be installed on the system:

- ▶ AIX 7.1:
 - bos.net.nfs.server 7.1.5.0
 - bos.net.nfs.client 7.1.5.0
- ▶ AIX 7.2:
 - bos.net.nfs.server 7.2.1.4
 - bos.net.nfs.client 7.2.1.4
- ▶ AIX 7.3:
 - bos.net.nfs.server 7.3.0.1
 - bos.net.nfs.client 7.3.0.1

Requirements for RSCT

Install the RSCT file sets before installing PowerHA. Ensure that each node has the same version of RSCT.

To determine whether the appropriate file sets are installed and what their levels are, issue the following commands:

```
/usr/bin/lslpp -l rsct.compat.basic.hacmp  
/usr/bin/lslpp -l rsct.compat.clients.hacmp  
/usr/bin/lslpp -l rsct.basic.rte  
/usr/bin/lslpp -l rsct.core.rmc
```

If the file sets are not present, install the appropriate version of RSCT, as shown in Table 1-3 on page 21.

1.6.2 Licensing for other software

Most software vendors require that you have a unique license for each application for each physical machine and on a per core basis. Usually, the license activation code is entered at installation time.

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

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

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

1.6.3 PowerHA licensing

PowerHA is licensed per active core, and its licensing depends on how the PowerHA cluster is configured. If the cluster is a typical active-passive (also known as hot-standby) one, then a minimum of one PowerHA core can be provisioned for a passive node along with all the cores on the active node. This configuration is often referred to as N+1, with N being the total cores on the active node. If a two-node cluster is configured for mutual takeover (also known as dual hot-standby), then you must provision PowerHA licenses for all the activated cores for both nodes to be license-compliant. If you have any questions about licensing, contact your IBM sales representative or IBM Business Partner.

PowerHA licensing considerations in IBM Cloud and IBM Power Virtual Server

Consider the following situation when licensing PowerHA on IBM Cloud:

- ▶ A customer's perpetual PowerHA licenses cannot be transferred to the cloud.
- ▶ Customers cannot bring their own PowerHA license to the cloud.
- ▶ Customers can acquire PowerHA for AIX fixed term licenses and deploy it on a system within their enterprise or in the cloud or a service provider machine.
- ▶ When deploying in public cloud, you should consider licensing as many processors as you want to hold in reserve on the secondary node.
- ▶ A public cloud is multi-tenant, so scaling up capacity on the secondary node on failover cannot be guaranteed.
- ▶ Enterprise Edition is required when replicating data within a PowerHA cluster.
- ▶ Standard Edition is deployed only with a shared-storage configuration.
- ▶ Subcapacity licensing means that only the processor cores that are incorporated into the cluster must be licensed.
- ▶ N+1 licensing means that the second node (system or LPAR) in the cluster requires only one PowerHA license. This situation is not necessarily applicable in a public cloud.

1.7 PowerHA software installation

The PowerHA software provides a series of facilities that you can use to make your applications highly available. Remember, not all system or application components are protected by PowerHA.

For example, if all the data for a critical application is on a single disk, then that disk is a SPOF for the entire cluster. If that specific disk fails, it cannot be protected by PowerHA. AIX LVM or storage subsystem protection must be used in this case. PowerHA provides takeover only for the disk on the backup node to make the data available for use.

This reason is why PowerHA planning is so important because your major goal throughout the planning process is to eliminate SPOFs. A SPOF exists when a critical cluster function is provided by a single component. If that component fails, the cluster has no other way of providing that function, and the application or service that depends on that component becomes unavailable.

A well-planned cluster provides higher application availability, performs as expected, and requires less maintenance than a poorly planned cluster. To help you get started, planning worksheets are provided in Appendix A, “Paper planning worksheets” on page 609.

1.7.1 Checking for prerequisites

After you complete the planning worksheets, verify that your system meets the requirements of PowerHA (see Table 1-3 on page 21). Many potential errors can be eliminated if you make this extra effort.

1.7.2 New installation

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

Installation by using an NIM server

As a best practice, use NIM because it allows you to load the PowerHA software onto other nodes faster from the server than from other media. Furthermore, it is a flexible way of distributing, updating, and administering your nodes. It allows you to install multiple nodes in parallel and provides an environment for maintaining software updates. This approach is useful and a time saver in large environments; for smaller environments, a local repository might be sufficient.

If you choose NIM, you must copy all the PowerHA file sets onto the NIM server and define an `lpp_source` resource before proceeding with the installation.

Installation from CD, DVD, or a hard disk drive

If your environment has only a few nodes, or if the use of NIM is more than you need, you can use CD or DVD installation or make a local repository by copying the PowerHA file sets locally and then use the `exportfs` command. This approach allows other nodes to access the data by using NFS.

1.7.3 Installing PowerHA

Before installing PowerHA SystemMirror for AIX, consult the release notes. For more information about installing and configuring PowerHA SystemMirror for AIX, see Chapter 4, “Installation and configuration” on page 133.

To install the PowerHA software on a server node, complete the following steps:

1. If you are installing directly from the installation media, such as a DVD image or from a local repository, enter the `smitty install_all` fast path command. The SMIT displays the “Install and Update from ALL Available Software” panel.
2. Enter the device name of the installation medium or installation directory in the INPUT device/directory for software field and press Enter.
3. Enter the corresponding field values.

To select the software to install, press F4 for a software listing, or enter `a11` to install all server and client images. Select the packages that you want to install according to your cluster configuration. Some of the packages might require prerequisites that are not available in your environment.

The following file sets are required and must be installed on all servers:

- `cluster.es.server`
- `cluster.es.client`
- `cluster.cspoc`

Read the license agreement and select Yes in the Accept new license agreements field. Choose Yes for this item to proceed with the installation. If you choose No, the installation might stop, and issue a warning that one or more file sets require the software license agreements. You accept the license agreement only once for each node.

4. Press Enter to start the installation process.

Tip: A best practice is to download and install the latest PowerHA SP at the time of installation from [IBM Fix Central](#).

Postinstallation steps

To complete the installation, complete the following steps:

1. Verify the software installation by using the AIX `1ppchk` command, and check the installed directories to see whether the expected files are present.
2. Run the `1ppchk -v` and `1ppchk -c cluster*` commands. No output is produced if the installation is good; if not, use the proper problem determination techniques to fix any problems.
3. A restart might be required if RSCT prerequisites were installed since the last time the system was restarted.

For more information about upgrading PowerHA, see Chapter 5, “Migration” on page 159.



High availability components

PowerHA uses the underlying cluster topology (nodes, networks, and storage) to keep the cluster resources highly available.

This chapter contains the following topics:

- ▶ PowerHA configuration data
- ▶ Software components
- ▶ Cluster topology
- ▶ Resources and resource groups
- ▶ Smart assists
- ▶ Other features
- ▶ Limits
- ▶ Storage characteristics
- ▶ Shared storage configuration
- ▶ PowerHA cluster events

2.1 PowerHA configuration data

The two main components of the cluster configuration are as follows:

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

When the cluster is configured, the cluster topology and resource information is entered on one node. Then, a verification process runs, and the data is synchronized out to the other nodes that are defined in the cluster. PowerHA keeps this data in its own Object Data Manager (ODM) classes on each node in the cluster.

Although PowerHA can be configured or modified from any node in the cluster, a best practice is to perform administrative operations from one node to ensure that PowerHA definitions are kept consistent across the cluster. This best practice prevents a cluster configuration update from multiple nodes, which might result in inconsistent data.

Use the following basic steps for configuring your cluster:

1. Define the cluster and the nodes.
2. Modify the topology as you want.
3. Verify and synchronize to check for errors.
4. Define the resources and resource groups (RGs).
5. Verify and synchronize.

AIX configuration and changes

PowerHA changes the system when PowerHA is installed or started. We describe these changes in the following sections.

Installation changes

The following AIX configuration changes are made:

- ▶ These files are modified:
 - /etc/inittab
 - /etc/rc.net
 - /etc/services
 - /etc/snmpd.conf
 - /etc/snmpd.peers
 - /etc/syslog.conf
 - /etc/trcfmt
 - /var/spool/cron/crontabs/root
- ▶ The hacmp group is added.
- ▶ The /etc/hosts file can be changed by adding or modifying entries by using the cluster configuration and verification auto-correct option.

- ▶ The **routerevalidate** network option is set to 1 (1 = enabled) on startup.
- ▶ The verification utility ensures that the value of each network option is consistent across all cluster nodes for the following settings:
 - **tcp_pmtu_discover**
 - **udp_pmtu_discover**
 - **ipignoreredirects**
 - **nbc_limit**
 - **nbc_pseg_limit**

Tuning operating system parameters

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

2.2 Software components

The following layered model describes the software components of a PowerHA cluster:

Application layer	Any application that is made highly available through the services that are provided by PowerHA.
PowerHA layer	Software that responds to changes within the cluster to ensure that the controlled applications remain highly available.
Reliable Scalable Cluster Technology (RSCT) layer	PowerHA communicates with RSCT Group Services (grpsvcs was replaced by ctags), but PowerHA replaced the topsvcs function with the new CAA function
AIX layer	Includes CAA components that provide network communications through both an IP network and repository disk. AIX also supports PowerHA through the Logical Volume Manager (LVM) layer that manages the storage.
LVM layer	Provides access to storage and status information back to PowerHA.
TCP/IP layer	Provides reliable communication, both node to node and node to client.

The application layer can consist of these items:

- ▶ Application code (programs, daemons, kernel extensions, and others)
- ▶ Application configuration data (files or binary files)
- ▶ Application (customer) data (files or raw devices)

The PowerHA layer consists of these items:

- ▶ PowerHA code (binary files, daemons and executable commands, libraries, and scripts)
- ▶ PowerHA configuration (ODM, and ASCII files)
- ▶ PowerHA log files
- ▶ Services:
 - Cluster manager (**c1strmgrES**)
 - Cluster information daemon (**c1infoES**)
 - Cluster event manager (**c1evmgrdES**)

The RSCT layer consists of these items:

- ▶ RSCT code (binary files, daemons and commands, libraries, and scripts)
- ▶ Configuration files (binary registry and ASCII files)
- ▶ Services:
 - IBM.ConfigRM
 - IBM.HostRM
 - IBM.ServiceRM
 - Group (**cthags**)
 - Resource Monitoring and Control (RMC) (**ctrmc**)

The AIX layer consists of these items:

- ▶ Kernel, daemons, and libraries
- ▶ CAA daemons
 - Cluster Communications daemon (**c1cmd**)
 - Cluster configuration daemon (**c1confd**)
- ▶ Device drivers
- ▶ Networking and TCP/IP layer
- ▶ LVM
- ▶ Configuration files (ODM and ASCII)

2.3 Cluster topology

The cluster topology represents the physical view of the cluster and how hardware cluster components are connected by using networks (IP network and non-IP network). To understand the operation of PowerHA, you must also understand the underlying topology of the cluster, the role that each component is responsible for, and how PowerHA interacts. In this section, we describe the following components:

- ▶ PowerHA cluster
- ▶ Nodes
- ▶ Sites
- ▶ Networks
- ▶ Network interfaces
- ▶ Communication devices
- ▶ Physical and logical networks

Figure 2-1 on page 31 shows a typical cluster topology and has these components:

- ▶ Two nodes
- ▶ Two IP networks (PowerHA logical networks) with redundant interfaces on each node
- ▶ Shared storage
- ▶ Repository disk

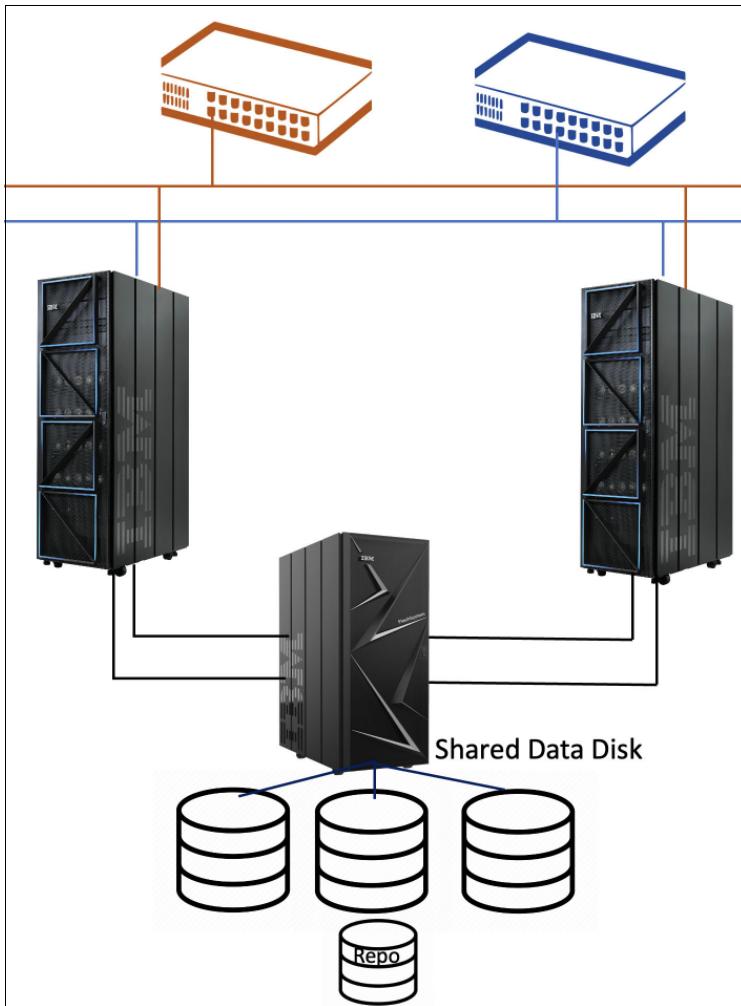


Figure 2-1 Example of cluster topology

2.3.1 PowerHA cluster

A name is assigned to the cluster. The name can be up to 64 characters that are [a-z], [A-Z], [0-9], hyphen (-), or underscore (_). In previous versions of PowerHA, the cluster name was not allowed to start with a number or contain hyphens, but this situation is no longer the case starting with PowerHA 7.1.3. The cluster ID (number) is also associated with the cluster, which automatically generates a unique ID for the cluster. All heartbeat packets contain this ID, so two clusters on the same network should never have the same ID.

2.3.2 Sites

The usage of sites is optional. They are primarily designed for use in cross-site LVM mirroring, PowerHA Enterprise Edition configurations, or both. A site consists of one or more nodes that are grouped at a location. PowerHA supports a cluster that is divided into two sites. Site relationships can also exist as part of an RG definition, but should be set to ignore if sites are not used.

Although using sites outside of PowerHA Enterprise Edition and cross-site LVM mirroring is possible, appropriate methods or customization must be provided to handle site operations. If sites are defined, site-specific events that run during `node_up` and `node_down` events might be unnecessary.

2.3.3 Cluster nodes

Nodes form the core of a PowerHA cluster. A node is a system running an image of the AIX operating system (stand-alone or a partition), PowerHA code, and application software. The maximum number of supported nodes is 16 in PowerHA 7. However, a CAA cluster supports 32 nodes.

When defining the cluster node, a unique name must be assigned and a communication path to that node must be supplied (an IP address or a resolvable IP label that is associated with one of the interfaces on that node). The node name can be the hostname (short), a fully qualified name (host name.domain.name), or any name up to 64 characters that are [a-z], [A-Z], [0-9], hyphen (-), or underscore (_). The name can start with either an alphabetic or numeric character.

The communication path is first used to confirm that the node can be reached, and then it is used to populate the ODM on each node in the cluster after secure communications are established between the nodes. However, after the cluster topology and CAA cluster are configured, any interface can be used to attempt to communicate between nodes in the cluster.

Important: If you want the node name to differ from the system hostname, you *must* explicitly state the hostname IP address for the communication path.

2.3.4 Networks

In PowerHA, the term *network* is used to define a logical entity that groups the communication interfaces that are used for IP communication between the nodes in the cluster, and for client access. The networks in PowerHA can be defined with an attribute of either *public* (which is the default) or *private*. Private networks indicate to CAA that it should *not* be used for heartbeat or communications.

Three network types can be used:

- ether** The most common type of network. It is used in almost all local clusters.
- XD_ip** Primarily used across sites for primary client communications.
- XD_data** This network type is unique to and used only in Geographic Logical Volume Manager (GLVM) configurations. This type GLVM informs which networks should be used specifically for data replication traffic.

PowerHA network interfaces

A *network interface* refers to the network adapter (either physical or virtual) that supports the TCP/IP protocol and is represented by an IP address. The network interfaces that are connected to a common physical network are combined into logical networks that are used by PowerHA.

Each interface can host several IP addresses. When configuring a cluster, you define the IP addresses that PowerHA monitors by using CAA and the IP addresses that PowerHA itself keeps highly available (the service IP addresses and persistent aliases).

The following terms describe PowerHA network interfaces:

IP address	The dotted decimal IP address.
IP label	The label that is associated with a particular IP address as defined by the name resolution method (domain name server (DNS). If the IP address is static, use /etc/hosts).
Base IP label or address	The default IP label or address that is set on the interface by AIX on startup. It is the base address of the interface.
Service IP label or address	An IP label or address over which a service is provided. It can be bound to a single node or shared by multiple nodes. Although not part of the topology, these addresses are the ones that PowerHA keeps highly available because they are defined as a resource within an RG.
Boot interface	Previous versions of PowerHA used the terms <i>boot adapter</i> and <i>standby adapter</i> , depending on the function. These terms are collapsed into one term (boot interface) to describe any IP network interface that can be used by PowerHA to host a service IP label or address.
IP aliases	An IP address that is added to an interface, rather than replacing its base IP address. This function is an AIX one that is supported by PowerHA. However, PowerHA assigns to the IP alias the same subnet mask of the base IP address over which it is configured.
Logical network interface	The name to which AIX resolves a port (for example, en0) of a physical network adapter.

Important: A best practice is to have all those IP addresses defined in the /etc/hosts file on all nodes in the cluster. There is certainly no requirement to use fully qualified names. While PowerHA is processing network changes, the *NSORDER* variable is set to *local* (for example, pointing to /etc/hosts). However, another best practice is to set the variable in the /etc/netsvc.conf file.

PowerHA communication devices

PowerHA also uses special communication devices that are provided by CAA. They are the repository disk and optional storage area network (SAN) heartbeat (*sfwcomm*) device. However, no System Management Interface Tool (SMIT) options are explicitly called *communication devices*, as in previous PowerHA versions.

Physical and logical networks

A *physical network* connects two or more physical network interfaces. PowerHA, like AIX, has the concept of logical networks. Two or more network interfaces on one physical network can be grouped to form a logical network. These logical networks are known by a unique name (for example, *net_ether_01* if assigned by PowerHA) and can consist of one or more subnets. A logical network can be viewed as the group of interfaces that are used by PowerHA to host one or more service IP labels or addresses.

Network definitions can be added by using the SMIT panels. However, during the initial cluster configuration, a discovery process is run that automatically defines the networks and assigns the interfaces to them.

The discovery process harvests information from the /etc/hosts file, defined interfaces, defined adapters, and existing enhanced concurrent mode (ECM) disks. Then, the process creates the following files in the /usr/es/sbin/cluster/etc/config directory:

clip_config	Contains details about the discovered interfaces. It is used in the F4 SMIT lists.
cvg_config	Contains details of each physical volume (PV) (physical volume identifier (PVID), volume group (VG) name, status, major number, and others), and a list of available major numbers.

Running discovery can also reveal any inconsistencies in the network at your site.

2.3.5 Cluster Aware AIX

The PowerHA cluster manager uses various sources to get information about possible failures:

- ▶ CAA and RSCT monitor the state of the network interfaces, and devices.
- ▶ AIX LVM monitors the state of the disks, logical volumes (LVs), and VGs.
- ▶ The PowerHA application monitors the state of the applications.

PowerHA SystemMirror 7.1 and later uses CAA services to configure, verify, and monitor the cluster topology. This change is a major reliability improvement because the core functions of the cluster services, such as topology-related services, now run in the kernel space. This setup makes the cluster services less susceptible to interference by the workloads running in the user space.

Communication paths

Cluster communication is achieved by communicating over multiple redundant paths. The following redundant paths provide a robust clustering foundation that is less prone to cluster partitioning:

- ▶ TCP/IP
 - PowerHA SystemMirror and CAA, either by using multicast or unicast, use all network interfaces that are available for cluster communication. All these interfaces are discovered by default and used for health management and other cluster communication. You can use the PowerHA SystemMirror management interfaces to remove any interfaces that you do not want to be used by specifying these interfaces in a *private* network.
- ▶ SAN-based (sfwcomm)
 - A redundant, high-speed path of communication is established between the hosts by using the SAN fabric that exists in any data center between the hosts. Discovery-based configuration reduces the burden for you to configure these links.
- ▶ Repository disk
 - Health and other cluster communication is also achieved through the central repository disk.

Of these three paths, only SAN-based is optional for any cluster.

Repository disk

CAA maintains cluster-related configuration information like a node list, and various cluster tunables. All this configuration information is also maintained in memory by CAA. In a live cluster, CAA can re-create the configuration information when a new replacement disk for the repository disk is provided.

Configuration management

CAA identifies the repository disk by using a unique, 128-bit universally unique identifier (UUID). The UUID is generated in the AIX storage device drivers by using the characteristics of the disk concerned. CAA stores the repository disk-related identity information in the AIX ODM Customized Attribute (CuAT) as part of the cluster information. Example 2-1 shows a sample output from a PowerHA 7.1.3 cluster.

Example 2-1 Sample output for a cluster configuration

CuAt:

```
name = "cluster0"
attribute = "node_uuid"
value = "d12204fe-a9a6-11e4-9ab9-96d758730003"
type = "R"
generic = "DU"
rep = "s"
nls_index = 3
```

CuAt:

```
name = "cluster0"
attribute = "clvdisk"
value = "2fb6d8b9-1147-45f9-185b-4e8e67716d4d"
type = "R"
generic = "DU"
rep = "s"
nls_index = 2
```

When another node tries to join a cluster during AIX boot time, CAA uses the ODM information to find the repository disk. The repository disk must be reachable to retrieve the necessary information to join and synchronize with all other nodes in the cluster. If CAA cannot reach the repository disk, then CAA does not proceed with starting the cluster services, and logs an error about the repository disk in the AIX error log. In this case, the administrator fixes the repository disk-related issues and then starts CAA manually.

If a node fails to join a cluster because the ODM entry is missing, the ODM entry can be repopulated and the node can be forced to join the cluster by using **clusterconf**, if the administrator knows the hard disk name for the repository disk (to discover the name, run **clusterconf -r hdisk#**).

Health management

The repository disk plays a key role in bringing up and maintaining the health of the cluster. Here are some of the ways that the repository disk is used for heartbeats, cluster messages, and node-to-node synchronization.

There are two key ways that the repository disk is used for health management across the cluster:

1. Continuous health monitoring
2. Distress time cluster communication

For *continuous health monitoring*, CAA and disk device drivers maintain health counters per node. These health counters are updated and read at least once every 2 seconds by the storage framework device driver. The health counters of the other nodes are compared every 6 seconds to determine whether the other nodes are still functional. These time settings may be changed in the future if necessary.

When all the network interfaces on a node have failed, then the node is in a *distress* condition. In this distress environment, CAA and the storage framework use the repository disk to do all the necessary communication between the distressed node and other nodes. This type of communication requires certain areas of the disk to be set aside per node for writing the messages that are meant to be delivered to other nodes. This disk space is automatically allocated at cluster creation time. No action from the customer is needed. When operating in this mode, each node must scan the message areas of all other nodes several times per second to receive any messages that are meant for them.

This second method of communication is not the most efficient form of communication. It requires more polling of the disk, and it is expected that this form of communication is used only when the cluster is in distress mode. A failover of selective failover network loss occurs automatically without any user intervention.

Failures of any of these reads/writes result in repository failure-related events toward CAA and PowerHA. The administrator must provide a new disk to use as a replacement disk for the original, failed repository disk.

Repository disk replacement

PowerHA 7.2.7 has a feature, which was introduced in PowerHA 7.2.0, to automate the replacement of a failed repository disk. This capability is called Automatic Repository Update (ARU). The purpose of ARU is to automate the replacement of a PowerHA repository disk if an active repository disk fails without intervention from a system administrator and without affecting the active cluster services. All that is required is to configure an additional repository disk to use if an active repository disk fails.

If a repository disk fails, PowerHA detects the failure of the active repository disk. Then, it verifies that the active repository disk is not usable. If the disk is not usable, PowerHA attempts to switch to the backup repository disk. If this switch is successful, then the backup repository disk becomes the active repository disk. For more information about the process for replacing the repository disk in PowerHA 7.2.7, see 6.6, “Repository disk replacement” on page 222.

Repository disks across sites

In PowerHA SystemMirror Enterprise Edition, defining two repository disks, one for each site, is required when configuring a linked cluster. The repositories between sites are kept in sync internally by CAA.

New quorum rule

Although the cluster continues operating if one or more nodes lose access to the repository disk, the affected nodes are considered to be in *degraded mode*. If the heartbeat communication is also affected, the nodes can potentially form an independent cluster (partition) by seeing other nodes register an abnormal failure.

Therefore, starting with PowerHA SystemMirror Enterprise Edition 7.1.2, PowerHA does not allow a node to operate if it no longer has access to the repository disk *and* also registers an abnormal node down event. This approach allows a double failure scenario to be tolerated.

Split and merge policies

When sites split and then merge, CAA provides a mechanism to reconcile the two repositories by defining split and merge policies. These policies apply only when you use linked clusters and PowerHA SystemMirror Enterprise Edition. The options and their definitions are described in Table 2-1.

PowerHA SystemMirror Enterprise Edition 7.1.3 introduced the manual split and merge policies. The policies can and should be applied globally across the cluster. However, there is also an option to specify whether they should apply to storage replication recovery.

Demonstration: For more information about the manual split-merge option, see the demonstration at [YouTube](#).

Table 2-1 Configuring the cluster split and merge policies fields

Policies	Options and description
Split handling policy	<ul style="list-style-type: none">▶ None: The default setting. Select it if you want the partitions to operate independently of each other after the split occurs.▶ Tie breaker:<ul style="list-style-type: none">Disk: Select this option to use the disk that is specified in the Select tiebreaker field after a split occurs. When the split occurs, one site wins the SCSI reservation on the tie breaker disk. The site that loses the SCSI reservation uses the recovery action that is specified in the policy setting. The disk that is used must support SCSI3-persistent or SCSI2 reserve to be a suitable candidate disk. Note: If you select TieBreaker-Disk in the Merge handling policy field, you must select TieBreaker-Disk for this field.NFS: Select this option to specify a Network File System (NFS) file as the tie breaker. During the cluster split, a predefined NFS file is used to decide the winning partition. The partition that loses the NFS file reservation uses the recovery action that is specified in the policy setting. Note: If you select TieBreaker-NFS in the Merge handling policy field, you must select TieBreaker-NFS for this field.Manual: Select this option to wait for manual intervention when a split occurs. PowerHA SystemMirror does not perform any actions on the cluster until you specify how to recover from the split. Note: If you select Manual in the Merge handling policy field, you must select Manual for this field.Cloud: Select this option to use a bucket from either IBM Cloud or Amazon Web Services (AWS) in a tiebreaker type fashion. Note: If you select Cloud in the Merge handling policy field, you must select Cloud for this field.

Policies	Options and description
Merge handling policy	<ul style="list-style-type: none"> ▶ Majority: Select this option to choose the partition with the highest number of nodes as the primary partition. ▶ Tie breaker: <ul style="list-style-type: none"> – Disk: Select this option to use the disk that is specified in the Select tiebreaker field after a split occurs. When the split occurs, one site wins the SCSI reservation on the tie breaker disk. The site that loses the SCSI reservation uses the recovery action that is specified in the policy setting. Note: If you select TieBreaker-Disk in the Split handling policy field, you must select TieBreaker-Disk for this field. – NFS: Select this option to specify an NFS file as the tie breaker. During the cluster split, a predefined NFS file is used to decide the winning partition. The partition that loses the NFS file reservation uses the recovery action that is specified in the policy setting. Note: If you select TieBreaker-NFS in the Split handling policy field, you must select TieBreaker-NFS for this field. ▶ Manual: Select this option to wait for manual intervention when a split occurs. PowerHA SystemMirror does not perform any actions on the cluster until you specify how to recover from the split. Note: If you select Manual in the Split handling policy field, you must select Manual for this field. ▶ Cloud: Select this option to use a bucket from either IBM Cloud or AWS in a tiebreaker type fashion. Note: If you select Cloud in the Split handling policy field, you must select Cloud for this field.
Split and merge action plan	<ul style="list-style-type: none"> ▶ Reboot: Restarts all nodes in the site that do not win the tie breaker or do not respond when you use the manual choice option. ▶ Disable Applications Auto-Start and Reboot: Select this option to restart nodes on the losing partition when a cluster split event occurs. If you select this option, the RGs are not brought online automatically after the system restarts. Note: This option is available only if your environment is running AIX 7.2.1 or later. ▶ Disable Cluster Services Auto-Start and Reboot: Select this option to restart nodes on the losing partition when a cluster split event occurs. If you select this option, CAA is not started. The RGs are not brought online automatically. After the cluster split event is resolved, go to SMIT, and select Problem DeterminationTools → Start CAA on Merged Node to restore the cluster.
Select tie breaker	Select an internet Small Computer System Interface (iSCSI) disk or a SCSI disk that you want to use as the tie breaker disk. The disk must support either SCSI-2 or SCSI-3 reserves.
NFS export server	This field is available if you specify Tie Breaker - NFS in both the Split Handling Policy and the Merge Handling Policy fields. Specify the fully qualified domain name of the NFS server that is used for the NFS tie-breaker. The NFS server must be accessible from each node in the cluster by using the NFS server IP address.
Local mount directory	This field is available if you specify Tie Breaker - NFS in both the Split Handling Policy and the Merge Handling Policy fields. Specify the absolute path of the NFS mount point that is used for the NFS tie-breaker. The NFS mount point must be mounted on all nodes in the cluster.

Policies	Options and description
NFS Export Directory	<ul style="list-style-type: none"> ▶ This field is available if you specify Tie Breaker - NFS in both the Split Handling Policy and the Merge Handling Policy fields. ▶ Specify the absolute path of the NFSv4 exported directory that is used for the NFS tie-breaker. The NFS exported directory must be accessible from all nodes in the cluster that use NFSv4. <p>Verify that the following services are active in the NFS server:</p> <ul style="list-style-type: none"> - biod - nfsd - nfsgryd - portmap - rpc.lockd - rpc.mountd - rpc.statd - TCP <p>Verify that the following services are active in the NFS client on all cluster nodes:</p> <ul style="list-style-type: none"> - biod - nfsd - rpc.mountd - rpc.statd - TCP
Manual Choice Option	<ul style="list-style-type: none"> ▶ <i>Notify Method</i>: This option is invoked in addition to a message to /dev/console to inform the operator that they must choose which site will continue after a split or merge. The method is specified as a path name, followed by optional parameters. When invoked, the last parameter is either split or merged to indicate the event. ▶ <i>Notify Interval</i>: This option is the frequency of the notification time, in seconds, between the message to inform the operator that they must choose which site will continue after a split or merge. The supported values are 10 - 3600. ▶ <i>Maximum Notifications</i>: This option is the maximum number of times that PowerHA SystemMirror prompts the operator to choose which site will continue after a split or merge. The default, blank, is infinite. Otherwise, the supported values are 3 - 1000. This value <i>cannot</i> be blank when a surviving site is specified. ▶ <i>Default Surviving Site</i>: This option is the site that continues if the operator does not respond to a request for a manual choice of a surviving site on a split or a merge. The other site takes the action that is selected under “action plan.” The time that the operator has to respond is this value Notify Interval x Maximum Notifications +1. ▶ <i>Apply to Storage Replication Recovery</i>: Determines whether the manual response on a split also applies to those storage replication recovery mechanisms that provide an option for manual recovery. If Yes is selected, the partition that was selected to continue on a split proceeds with the takeover of the storage replication recovery. This option <i>cannot</i> be used if IBM DS8000 or IBM XIV replication is used.

2.3.6 Reliable Scalable Cluster Technology

RSCT is a set of low-level operating system components that you may use to implement cluster technologies such as PowerHA SystemMirror, GPFS, and others.

All the RSCT functions are based on the following components:

- ▶ RMC subsystem

Considered the backbone of RSCT. The RMC runs on each single server and provides a common abstraction layer of server resources (hardware or software components).

- ▶ RSCT core resource manager

A software layer between a resource and RMC. Resource Manager maps the abstraction that is defined by RMC to real calls and commands for each resource.

- ▶ RSCT security services

Provides the security infrastructure that is required by RSCT components to authenticate the identity of other parties.

- ▶ Topology service subsystem

Provides the infrastructure and mechanism for the node and network monitoring and failure detection.

Important: Starting with PowerHA 7.1.0 and later, the RSCT topology service subsystem is deactivated, and all its functions are performed by CAA topology services.

- ▶ Group services subsystem

Coordinates cross-node operations in the cluster environment. This subsystem is responsible for spanning changes across all cluster nodes and ensures all of them finish properly with all modifications performed.

Figure 2-2 shows CAA, RSCT daemons, and how they interact with each other and the PowerHA daemons and with other applications.

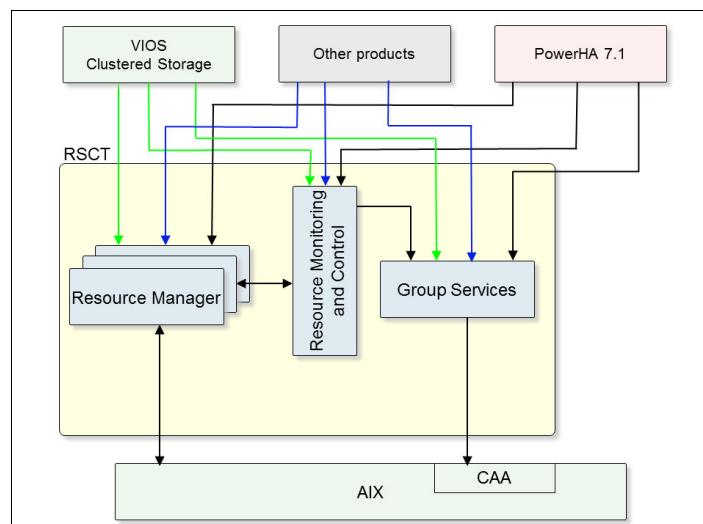


Figure 2-2 RSCT, CAA, and PowerHA interaction

2.3.7 Internet Protocol networks

Internet Protocol networks that use only Ethernet can be classified as public or private:

Public	Logical networks that are designed for client communication to the nodes. Each one is built from a collection of the IP adapters, so each network can contain multiple subnets. Because these networks are designed for client access, IP Address Takeover (IPAT) is supported.
Private	These networks were historically for use by applications such as Oracle Real Application Cluster (RAC). This setting is an indicator to CAA to exclude the interfaces that are part of this network from heartbeating.

2.3.8 The IP Address Takeover mechanism

A key role of PowerHA is to maintain the service IP labels and addresses so that they are highly available. PowerHA does this task by starting and stopping each service IP address as required on the appropriate interface. PowerHA 7.x supports IPAT only through aliasing.

IPAT through aliasing

When using the Disable FirstAlias service distribution policy, the service IP label, or address, is aliased onto the interface without removing the underlying boot IP address by using the **ifconfig** command, as shown in Figure 2-3. IPAT through aliasing also makes the concept of standby interfaces obsolete (all network interfaces are labeled as boot interfaces).

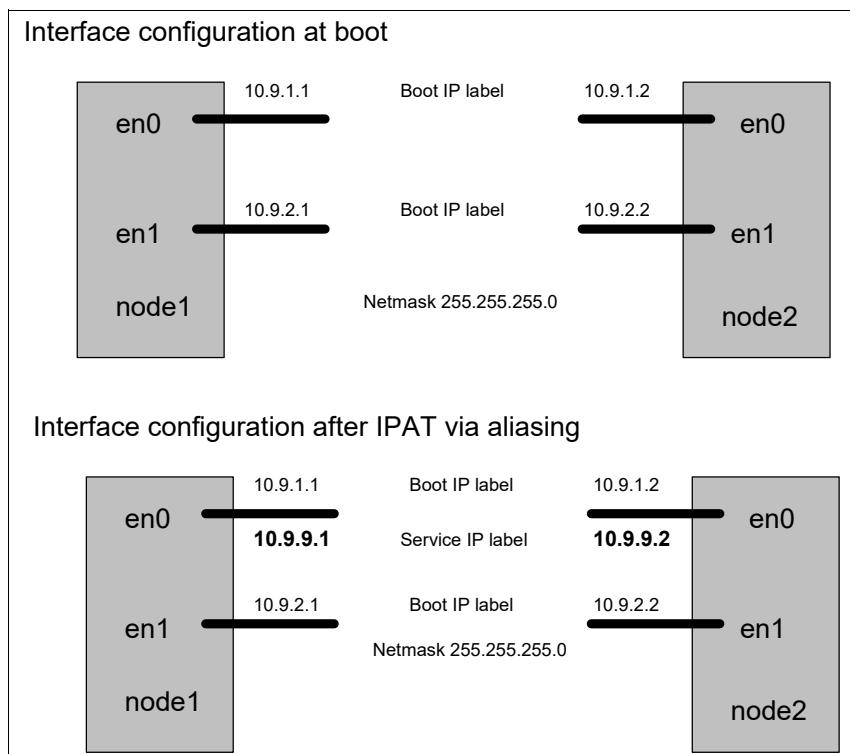


Figure 2-3 IPAT through IP aliases

As IP addresses are added to the interface through aliasing, more than one service IP label can coexist on one interface. By removing the need for one interface per service IP address that the node can host, IPAT through aliasing is the more flexible option, and sometimes can require less hardware. IPAT through aliasing also reduces failover time, because adding an alias to an interface is faster than removing the base IP address and then applying the service IP address.

IPAT through aliasing is supported only on networks that support the gratuitous Address Resolution Protocol (ARP) function of AIX. Gratuitous ARP is when a host sends out an ARP packet before using an IP address and the ARP packet contains a request for this IP address. In addition to confirming that no other host is configured with this address, gratuitous ARP ensures that the ARP cache on each machine on the subnet is updated with this new address.

If multiple service IP alias labels or addresses are active on one node, PowerHA by default equally distributes them among all available interfaces on the logical network. This placement can be controlled by using distribution policies, which is explained in more detail in 12.4, “Site-specific service IP labels” on page 509.

For IPAT through aliasing, each boot interface on a node must be on a different subnet, though interfaces on different nodes can obviously be on the same subnet. The service IP labels can be on the same subnet as the boot adapter *only* if it is a single adapter configuration. Otherwise, they must be on separate subnets also.

Important: For IPAT through aliasing networks, PowerHA briefly has the service IP addresses active on both the failed Interface and the takeover interface so that it can preserve routing. This approach might cause a DUPLICATE IP ADDRESS error log entry, which can be ignored.

2.3.9 Persistent IP label or address

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

- ▶ Always stays on the same node (is node-bound).
- ▶ Coexists with other IP labels that are present on the same interface.
- ▶ Does not require the installation of another physical interface on that node.
- ▶ Is not part of any RG.

By assigning a persistent node IP label for a network on a node, you may have a highly available node-bound address on a cluster network. This address can be used for administrative purposes because it always points to a specific node regardless of whether PowerHA is running.

Note: There can be one persistent IP label per network per node. For example, if a node is connected to two networks that are defined in PowerHA, that node can be identified through two persistent IP labels (addresses), one for each network.

The persistent IP labels are defined in the PowerHA configuration, and they become available when the cluster definition is synchronized. A persistent IP label remains available on the interface on which it was configured, even if PowerHA is stopped on the node or the node is restarted. If the interface on which the persistent IP label is assigned fails while PowerHA is running, the persistent IP label is moved to another interface in the same logical network on the same node.

The persistent IP alias must be on a different subnet from each of the boot interface subnets and can be either in the same subnet or in a different subnet of the service IP address. If the node fails or all interfaces on the logical network on the node fail, then the persistent IP label is no longer available.

2.3.10 Cluster heartbeat settings

PowerHA, through CAA, can modify the time that is involved to discover a network and a node (among other things) by using cluster tunables. For more information about these options, see 12.3.1, “Changing cluster-wide tunables” on page 506.

2.3.11 Network security considerations

PowerHA security is important to limit both unauthorized access to the nodes and unauthorized interception of inter-node communication. Earlier versions of PowerHA used **rsh** to run commands on other nodes. This approach was difficult to secure, and IP addresses could be spoofed. PowerHA uses the CAA Cluster Communications daemon (**c1cmd**) to control communication between the nodes.

PowerHA provides cluster security by using these methods:

- ▶ Controlling user access to PowerHA
- ▶ Providing security for inter-node communications

For more information about CoD, see 8.1, “Cluster security” on page 348.

The Cluster Communications daemon

The Cluster Communications daemon, **c1cmd**, runs remote commands based on the principle of least privilege, which ensures that no arbitrary command can run on a remote node with root privilege. Only a small set of PowerHA commands are *trusted* and allowed to run as root. These commands are in /usr/es/sbin/cluster. The remaining commands do not have to run as root.

The Cluster Communications daemon is started by **inittab**, with the entry being created by the installation of PowerHA. The daemon is controlled by the system resource controller, so **startsrc**, **stopsrc**, and **refresh** work. In particular, **refresh** is used to reread /etc/cluster/rhosts and move the log files.

You use the /etc/cluster/rhosts file before the cluster is first synchronized in an insecure environment. After the CAA cluster is created, the only time that the file is needed is when you add more nodes to the cluster. After the cluster is synchronized and the CAA cluster is created, the contents within the file can be deleted. However, do not remove the file.

The Cluster Communications daemon provides the transport medium for PowerHA cluster verification, global ODM changes, and remote command execution. The following commands use **c1cmd** (they cannot be run by a standard user):

c1exec	Run specific and potentially dangerous commands.
c1_rcp	Copy AIX configuration files.
c1_rsh	Used by the cluster to run commands in a remote shell.
c1cmd	Takes an AIX command and distributes it to a set of nodes that are members of a cluster.

The Cluster Communications daemon is also used for these tasks:

- ▶ File collections
- ▶ Auto synchronization and automated cluster verification
- ▶ User and password administration
- ▶ Cluster Single Point of Control (C-SPOC)

The logging for the clcomd daemon is turned on by default, and the log files `clcomd.log` and `clcomddiag.log` can be found within the `/var/hacmp/clcomd` directory.

2.4 Resources and resource groups

This section describes the PowerHA resource concepts of definitions, resources, and RGs.

2.4.1 Definitions

PowerHA uses the underlying topology to ensure that the applications that are under its control and the resources they require are kept highly available:

- ▶ Service IP labels or addresses
- ▶ Physical disks
- ▶ VGs
- ▶ LVs
- ▶ File systems
- ▶ NFS
- ▶ Application controller scripts
- ▶ Application monitors
- ▶ Tape resources

The applications and the resources that are required are configured into RGs. The RGs are controlled by PowerHA as single entities whose behavior can be tuned to meet the requirements of clients and users.

Figure 2-4 on page 45 shows resources that PowerHA makes highly available, superimposed on the underlying cluster topology.

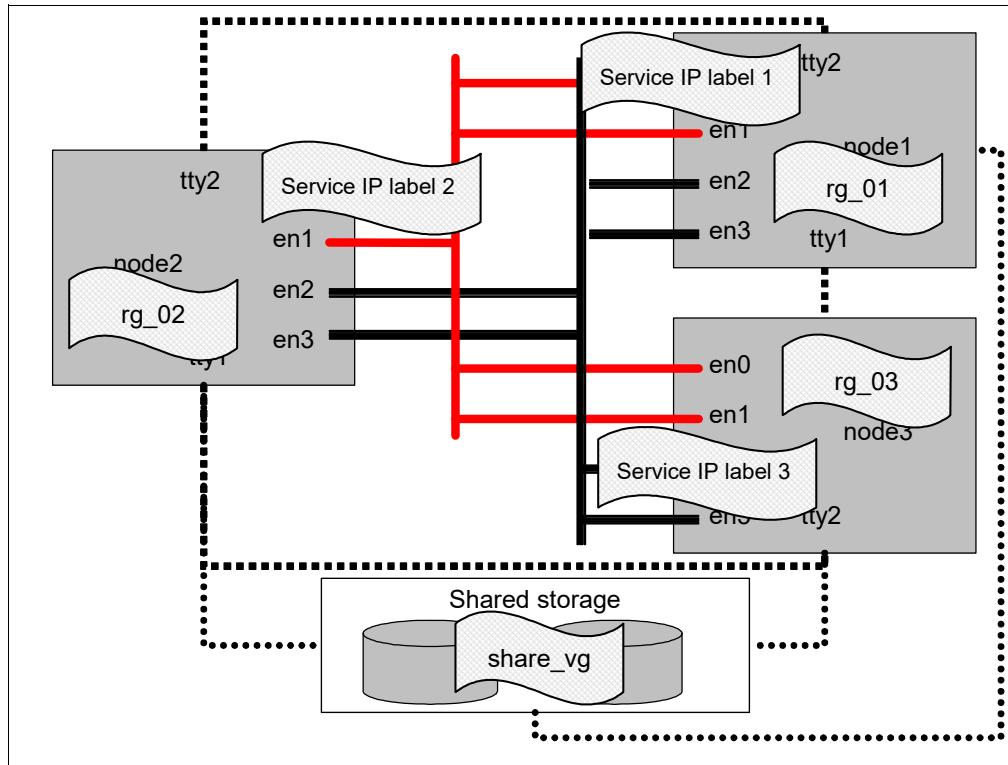


Figure 2-4 Highly available resources superimposed on the cluster topology

The following common resources that are shown in Figure 2-4 are made highly available:

- ▶ Service IP labels
- ▶ Applications shared between nodes
- ▶ Storage is shared between nodes

2.4.2 Resources

The items in this section are considered resources in a PowerHA cluster.

Service IP address or label

The service IP address is an IP address that is used by clients to access the applications or nodes. This service IP address, and its associated label, is monitored by PowerHA, and is a resource that is assigned to an RG. The two types of service IP addresses or labels are as follows:

- ▶ Shared service IP address or label

An IP address that can be configured on multiple nodes and is part of an RG that can be active on only one node at a time.
- ▶ Node-bound service IP address or label

An IP address that can be configured on only one node (is not shared by multiple nodes). Typically, this service IP address is associated with concurrent RGs.

The service IP addresses become available when PowerHA brings the associated RG into an ONLINE status.

The placement of the service IP labels is determined by the specified Service IP label distribution preference. The IP label distribution preference can also be changed dynamically, but is used only in subsequent cluster events to avoid any extra interruptions in service. For more information about the available options, see 12.2, “Distribution preference for service IP aliases” on page 501.

Storage

The following storage types can be configured as resources:

- ▶ VGs (AIX and Veritas virtual machines (VMs)).
- ▶ LVs (all LVs in a defined VG).
- ▶ File systems (JFS and JFS2): Either for all the defined VGs, or each can be specified individually.
- ▶ Raw disks: Defined by PVID.

If storage will be shared by some or all the nodes in the cluster, then all the components must be on external storage and configured in such a way that failure of one node does not affect the access by the other nodes.

The storage can be accessed in two ways:

- ▶ Non-concurrent configurations where one node owns the disks, allowing clients to access them along with other resources that are required by the application. If this node fails, PowerHA determines the next node to take ownership of the disks, restart applications, and provide access to the clients. ECM disks are often used in non-concurrent configurations. ECM refers to the method of locking access to the disks, not to the access itself being concurrent or not.
- ▶ Concurrent configurations where one or more nodes may access the data concurrently with locking that is controlled by the application. The disks must be in a concurrent VG.

For a list of devices that are supported by PowerHA, see the [PowerHA Hardware Support Matrix](#).

Important: Even if a third-party storage product is not listed in the matrix, it still might be supported. If Virtual I/O Server (VIOS) supports third-party storage, and PowerHA supports virtual devices through VIOS, then the storage should also be supported by PowerHA. However, always verify support with the storage vendor.

Choosing redundant array of independent disks data protection

Storage protection (data or otherwise) is independent of PowerHA. For high availability (HA) of storage, you must use storage that has proper redundancy and fault tolerance levels. PowerHA does not have any control on storage availability.

For data protection, you can use either redundant array of independent disks (RAID) technology (at the storage or adapter level) or AIX LVM mirroring (RAID 1).

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

The RAID levels are as follows:

► RAID 0

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

- Data transfer rates are higher for sequential operations because of the overlapping of multiple I/O streams.
- Random access throughput is higher because access pattern skew is eliminated as a result of the distribution of the data. With data distributed evenly across several disks, random accesses most likely find the required information spread across multiple disks, and thus benefit from the increased throughput of more than one drive.

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

► RAID 1

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

► RAID 2 and RAID 3

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

► RAID 4

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

► RAID 5

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

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

► RAID 6

Identical to RAID 5, except it uses one more parity block than RAID 5. You can have two disks fail and still have data integrity. Often referred to as *double parity*.

► RAID 0+1 (RAID 10)

RAID 0+1, also known as IBM RAID-1 Enhanced, or RAID 10, is a combination of RAID 0 (data striping) and RAID 1 (data mirroring). RAID 10 provides the performance advantages of RAID 0 while maintaining the data availability of RAID 1. In a RAID 10 configuration, both the data and its mirror are striped across all the disks in the array. The first stripe is the data stripe, and the second stripe is the mirror, with the mirror placed on a different physical drive than the data. RAID 10 implementations provide excellent write performance because they do not have to calculate or write parity data. RAID 10 can be implemented by using software (AIX LVM), hardware (storage subsystem level), or a combination of hardware and software. The appropriate solution for an implementation depends on the overall requirements. RAID 10 has the same cost characteristics as RAID 1.

Some newer storage subsystems have any more specialized RAID type methods that do not fit exactly into any of these categories, for example, IBM XIV.

Important: Although all RAID levels (other than RAID 0) have data redundancy, data must be regularly backed up, which is the only way to recover data if a file or directory is corrupted or deleted.

LVM quorum issues

A quorum must be enabled for concurrent VGs because each node might be accessing a different disk, and without proper locking that access might result in data divergence.

Leaving the quorum on, which is the default, causes RG failover if the quorum is lost. The VG is forced to vary on the next available node if a forced varyon of VGs attribute is enabled. When forced varyon of VG is enabled, PowerHA checks to determine the following conditions:

- At least one copy of each mirrored set is in the VG.
- Each disk is readable.
- At least one accessible copy of each logical partition (LPAR) is in every LV.

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

Note: An automatic failover of a VG through quorum loss is also referred to as *selective failover on VG loss*, which is enabled by default and can be disabled if needed. However, this setting affects all VGs that are assigned as a resource to PowerHA.

Using enhanced concurrent mode volume groups

PowerHA 7.x requires that the shared data VGs be an enhanced concurrent VG (ECVG). With ECVGs, you can vary on the VG in two modes:

Active state	The VG behaves the same way as the traditional varyon. Operations can be performed on the VG, and LVs and file systems can be mounted.
Passive state	The passive state allows limited read only access to the Volume Group Descriptor Area (VGDA) and the logical volume control block (LVCB).

When a node is integrated into the cluster, PowerHA builds a list of all ECVGs that are a resource in any RG containing the node. Then, these VGs are activated in passive mode.

When the RG comes online on the node, then the ECVGs are varied on in active mode. When the RG goes offline on the node, the VG is varied off to passive mode.

Important: PowerHA also uses the Journaled File System 2 (JFS2) **mountguard** option. This option prevents a file system from being mounted on more than one system at a time. PowerHA 7.1.1 and later automatically enable this feature if it is not already enabled.

Shared physical volumes

For applications that access raw disks, the PVID can be added as a resource in an RG.

Shared logical volumes

Although not explicitly configured as part of an RG, each LV in a shared VG is available on a node when the RG is online. These shared LVs can be configured to be accessible by one node at a time or concurrently by several nodes if the VG is part of a concurrent RG. If the ownership of the LV must be modified, reset it each time after the parent VG is imported.

Some applications that use raw LVs can start writing from the beginning of the device, therefore overwriting the LVCB. In this case, the application should be configured to skip at least the first 512 bytes of the LV where the LVCB is stored.

Custom disk methods

The extended resource SMIT menus allow the creation of custom methods to handle disks, volumes, and file systems. To create a custom method, you must define to PowerHA the appropriate scripts to manage the item in a highly available environment, as in these examples:

- For custom disks:** PowerHA provides scripts to identify ghost disks, determine whether a reserve is held, break a reserve, and make the disk available.
- For VGs:** PowerHA provides scripts to list VG names, list the disks in the VG, and bring the VG online and offline.
- For file systems:** PowerHA provides scripts to mount, unmount, list, and verify a status.

Custom methods are provided for Veritas Volume Manager (VxVM) starting with the Veritas Foundation Suite (VFS) 4.0. For a later version, you might need to create a custom user-defined resource to handle the storage. For more information about this option, see 2.4.7, “User-defined resources and types” on page 52.

File systems (jfs and jfs2) recovery by using fsck and logredo

AIX native file systems use database journaling techniques to maintain their structural integrity. After a failure, AIX uses the Journaled File System log (JFSlog) **logredo** utility to restore the file system to its last consistent state. This process is faster than using the **fsck** utility. If the process of replaying the JFSlog fails, an error occurs and the file system is not mounted.

The **fsck** utility performs a verification of the consistency of the file system by checking the inodes, directory structure, and files. Although this approach is more likely to recover damaged file systems, it does take longer. Both options are available to be chosen within an RG, with **fsck** as the default setting.

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

2.4.3 NFS

PowerHA works with the AIX NFS to provide a highly available NFS server, which allows the backup NFS server to recover the current NFS activity if the primary NFS server fails. This feature is available only for two-node clusters when using NFSv2 or NFSv3, and more than two nodes when using NFSv4 because PowerHA preserves locks for the NFS file systems and handles the duplicate request cache correctly. The attached clients experience the same hang if the NFS RG is acquired by another node as they would if the NFS server restarts.

When configuring NFS through PowerHA, you can control these items:

- ▶ The network that PowerHA uses for NFS mounting.
- ▶ NFS exports and mounts at the directory level.
- ▶ Export options for NFS exported directories and file systems. This information is kept in `/usr/es/sbin/cluster/etc/exports`, which has the same format as the AIX exports file (`/etc/exports`).

Note: Using the alternative `/usr/es/sbin/cluster/etc/exports` file is optional. PowerHA SystemMirror checks this file, and if there is an entry for the file system or directory in this file, PowerHA SystemMirror uses the options that are listed. If the NFS file system or directory is not listed or if the alternative exports file does not exist, the file system or directory is exported from NFS with the default option of root access for all cluster nodes.

NFS and PowerHA limitations

The following limitations apply:

- ▶ Only two nodes are allowed in the cluster if the cluster is using NFSv2 or NFSv3. More than two nodes are allowed if the cluster is using NFSv4.
- ▶ Shared VGs that contain file systems to be exported by NFS must have the same major number on all nodes, or the client applications will not recover on a failover.
- ▶ If NFS exports are defined on the node through PowerHA, all NFS exports must be controlled by PowerHA. AIX and PowerHA NFS exports cannot be mixed.

- ▶ If an RG has NFS exports defined, the field “Filesystems mounted before IP configured” must be set to true.
- ▶ By default, an RG that contains NFS exported file systems will automatically be cross-mounted. This situation also implies that each node in the RG acts as an NFS client, so you must have an IP label on the same subnet as the service IP label for the NFS server.
- ▶ For PowerHA SystemMirror and NFS to work together, the IP address for the NFS server must be configured in an RG for HA.
- ▶ To ensure the best performance, NFS file systems that are used by PowerHA SystemMirror should include the entry `vers=<version number>` in the options field in the `/etc/filesystems` file.

NFS cross-mounts

NFS cross-mounts work as follows:

- ▶ The node that is hosting the RG mounts the file systems locally. NFS exports and mounts them, so the node becomes both an NFS server and an NFS client.
- ▶ All other participating nodes of the RG NFS-mount the file systems, and the nodes become NFS clients.
- ▶ If the RG is acquired by another node, that node mounts the file system locally and NFS exports them, and the node becomes the new NFS server.

Consider the following example:

- ▶ Node1 with service IP label `svc1` locally mounts `/fs1` and NFS exports it.
- ▶ Node2 NFS-mounts `svc1:/fs1` on `/mntfs1`.
- ▶ Node1 also NFS-mounts `svc1:/fs1` on `/mntfs1`.

2.4.4 Application controller scripts

Virtually any application that can run on a stand-alone AIX server can run in a clustered environment that is protected by PowerHA. The application must be able to be started and stopped by scripts, and to be recovered by running a script after an unexpected shutdown. All these actions can be performed without manual intervention. Applications are defined to PowerHA as application controllers with the following attributes:

Start script	This script must be able to start the application from both clean and unexpected shutdowns. Output from the script is logged in the <code>hacmp.out</code> log file if <code>set -x</code> is defined within the script. The exit code from the script is monitored by PowerHA.
Stop script	This script must be able to successfully stop the application. Output is also logged, and the exit code monitored.
Application monitors	To keep applications highly available, PowerHA can monitor the application and the required resources.
Application startup mode	Introduced in PowerHA 7.1.1 this mode specifies how the application controller startup script is called. Select <code>background</code> , which is the default value, if you want the start script to be called as a background process. Event processing continues even if the start script does not complete. Select <code>foreground</code> if you need the event processing to wait until the start script exits.

The full path name of the script must be the same on all nodes, but the contents of the script itself can be different from node to node. If the contents do differ on each node, this situation inhibits your ability to use the file collections feature, which is why we suggest that you have an intelligent script that can determine on which node the script is running and then start the script.

While the exit codes from the application scripts are monitored, PowerHA assumes that a nonzero return code (RC) from the script means that the script failed and that starting or stopping the application was not successful. If so, the RG goes into the ERROR state, and a config_too_long message is recorded in the hacmp.out log.

Consider the following factors when configuring the application for PowerHA:

- ▶ The application is compatible with the AIX version.
- ▶ The storage environment is compatible with a highly available cluster.
- ▶ The application and platform interdependencies must be understood. The location of the application code, data, temporary files, sockets, pipes, and other components of the system, such as printers, must be replicated across all nodes that host the application.
- ▶ The application must be able to start and stop without any operator intervention, particularly after a node unexpectedly halts. The application start and stop scripts must be thoroughly tested before implementation and with every change in the environment.
- ▶ The RG that contains the application must contain all the resources that are required by the application, or be the child of an RG that does.
- ▶ Application licensing must be accounted for. Many applications have licenses that depend on the CPU ID; careful planning must be done to ensure that the application can start on any node in the RG node list. Also, be careful with the numbers of CPUs and other items on each node because some licensing is sensitive to these amounts.

2.4.5 Application monitors

By default, PowerHA is application-unaware. However, with PowerHA, application monitors can be used to ensure that applications are kept highly available. On a failure, PowerHA can respond as you want to the failure. For more information, see 7.7.9, “Application monitoring” on page 333.

Application availability

PowerHA also offers an application availability analysis tool, which is useful for auditing the overall application availability and assessing the cluster environment. For more information, see 7.7.10, “Measuring application availability” on page 344.

2.4.6 Tape resources

Some SCSI and Fibre Channel (FC) connected tape drives can be configured as highly available resources as part of any non-concurrent RG. This feature is rarely used.

2.4.7 User-defined resources and types

With PowerHA SystemMirror, you can add your own resource type and specify management scripts to configure how and where PowerHA SystemMirror processes the resource type. Then, these scripts can be added to a user-defined resource instance for use in an RG.

A user-defined resource type is one where you define a customized resource that you can add to an RG. A user-defined resource type contains several attributes that describe the properties of the instances of the resource type.

Ensure that the user-defined resource type management scripts exist on all nodes that are possible owners of the RG where the user-defined resource is. For more information and configuration options for user-defined resources and types, see 11.3, “User-defined resources and types” on page 481.

2.4.8 Resource groups

Each resource must be included in an RG to be made highly available by PowerHA. RGs allow PowerHA to manage a related group of resources as a single entity. For example, an application can consist of start and stop scripts, a database, and an IP address. These resources are included in an RG for PowerHA to control as a single entity.

PowerHA ensures that RGs remain highly available by moving them from node to node as conditions within the cluster change. The main states of the cluster and the associated RG actions are as follows:

Cluster startup	The nodes in the cluster are up, and then the RGs are distributed according to their startup policy.
Resource failure or recovery	When a particular resource that is part of an RG becomes unavailable, the RG can be moved to another node. Similarly, the RG can be moved back when the resource becomes available.
PowerHA shutdown	There are several ways to stop PowerHA on a node. One method causes the node's RGs to fall over to other nodes. Another method takes the RGs offline. Sometimes, stopping the cluster services on the node while leaving the resources active is possible.
Node failure or recovery	If a node fails, the RGs that were active on that node are distributed among the other nodes in the cluster, depending on their failover distribution policies. When a node recovers and is reintegrated into the cluster, RGs can be reacquired depending on their fallback policies.
Cluster shutdown	When the cluster is shut down, all RGs are taken offline. However, for some configurations, the resources can be left active, but the cluster services are stopped.

Before learning about the types of behavior and attributes that can be configured for RGs, you must understand the following terms:

Node list	The list of nodes that can host a particular RG. Each node must be able to access the resources that make up the RG.
Default node priority	The order in which the nodes are defined in the RG. An RG with default attributes moves from node to node in this order as each node fails.
Home node	The highest priority node in the default node list. By default, this node is the one on which an RG will initially be activated. The home node does not specify the node on which the RG is active.
Startup	The process of bringing an RG into an online state.

Fallover	The process of moving an RG that is online on one node to another node in the cluster in response to an event.
Fallback	The process of moving an RG that is online on a node that is not its home node to a re-integrating node.

Resource group behavior: Policies and attributes

The behavior of RGs is defined by configuring the RG policies and behavior. The key attributes of the RGs are the startup, fallover, and fallback options. The following sections describe the custom RG behavior options:

- ▶ Startup options
- ▶ Fallover options
- ▶ Fallback options

Startup options

These options control the behavior of the RG on initial startup:

- ▶ Online on Home Node Only (OHNO)

The RG is brought online when its home node joins the cluster. If the home node is not available, it stays in an offline state, as shown in Figure 2-5.

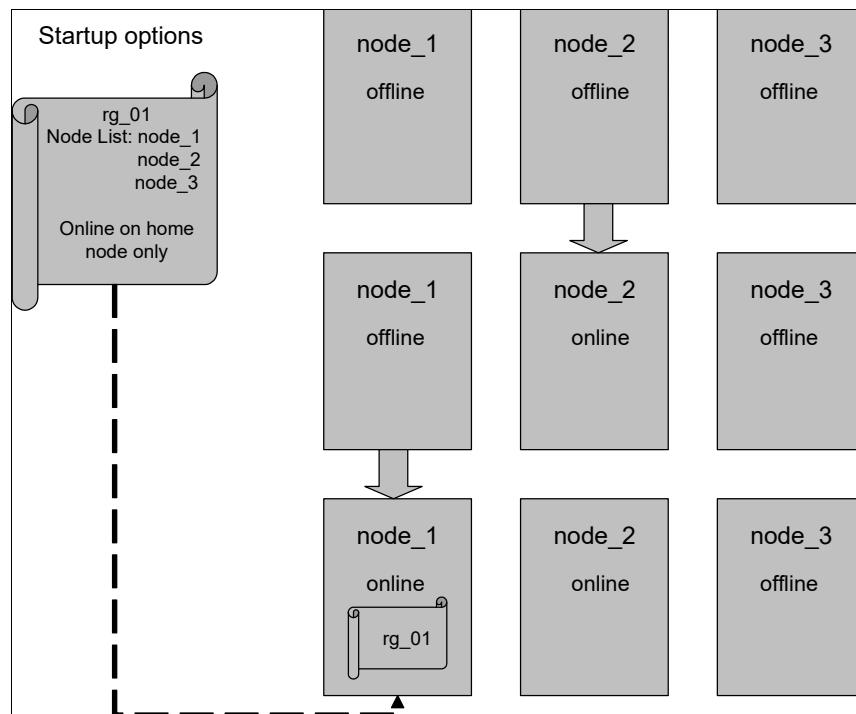


Figure 2-5 Online on Home Node Only

- ▶ Online on First Available Node (OFAN)

The RG is brought online when the first node in its node list joins the cluster, as shown in Figure 2-6 on page 55.

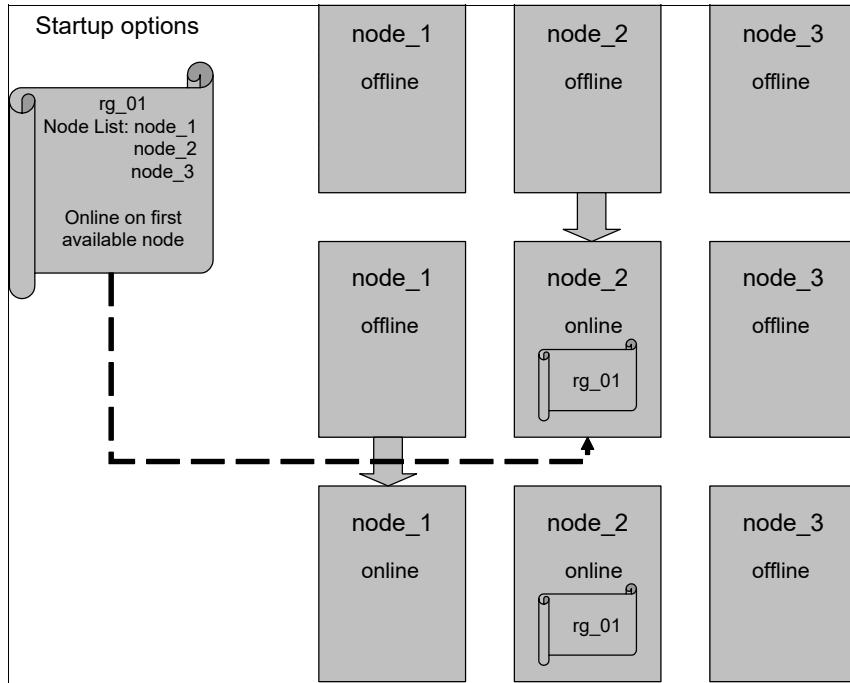


Figure 2-6 Online on First Available Node

► Online on all available nodes

The RG is brought online on all nodes in its node list as they join the cluster, as shown in Figure 2-7.

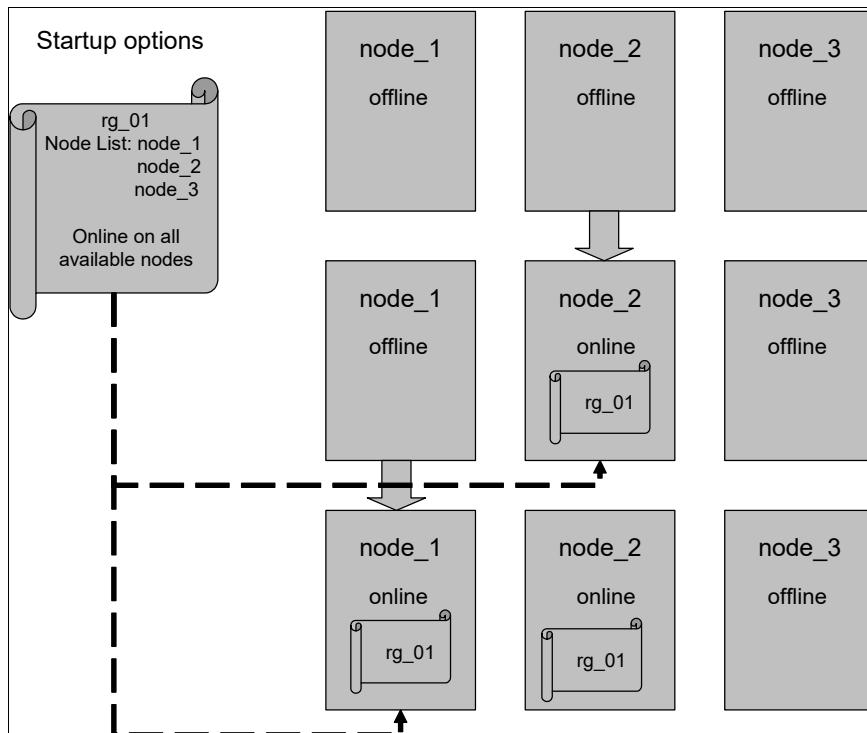


Figure 2-7 Online on all available nodes

- ▶ Online using distribution policy

The RG is brought online only if the node has no other RG of this type already online, as shown in Figure 2-8. If more than one RG of this type exists when a node joins the cluster, PowerHA selects the RG with fewer nodes in its node list. However, if one node has a dependent RG (that is, it is a parent in a dependency relationship), it is given preference.

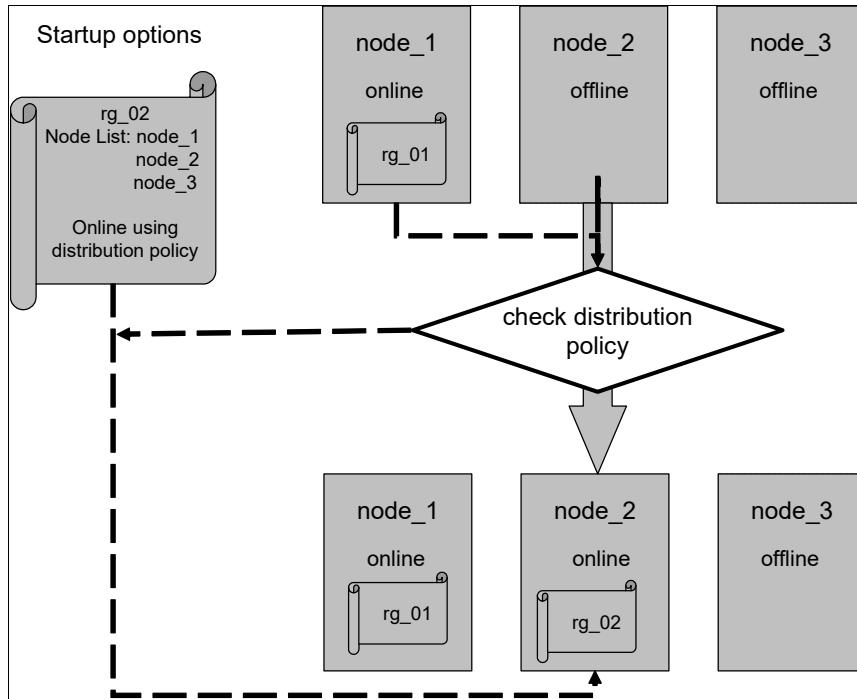


Figure 2-8 *Online using distribution policy*

Failover options

These options control the behavior of the RG if PowerHA must move it to another node in the response to an event:

- ▶ Fall over to next priority node in list

The RG falls over to the next node in the RG node list, as shown in Figure 2-9 on page 57.

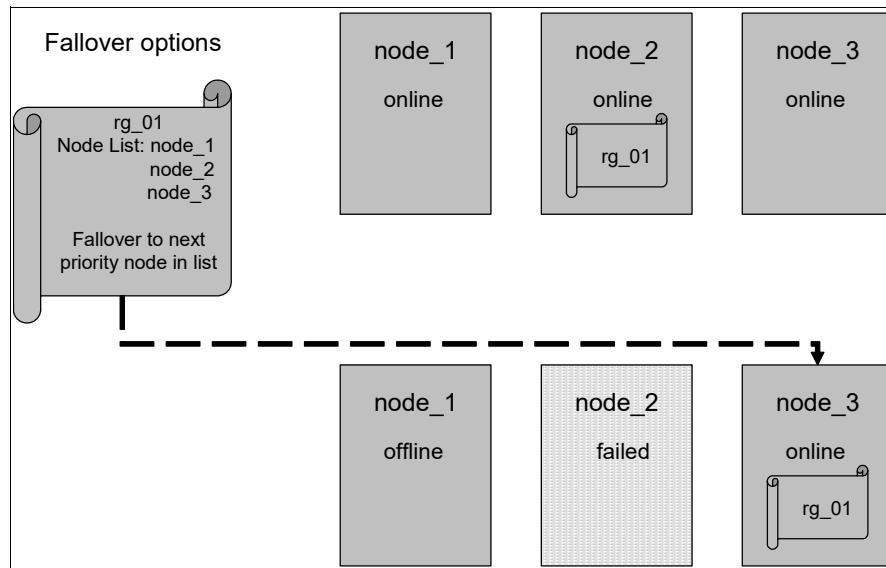


Figure 2-9 Fallback to next priority node in list

► Fallback using dynamic node priority

Dynamic node priority (DNP), as shown in Figure 2-10, entails the selection of a node that acquires the RG based on values of system attributes that are calculated at run time. These values are obtained by querying the RMC subsystem.

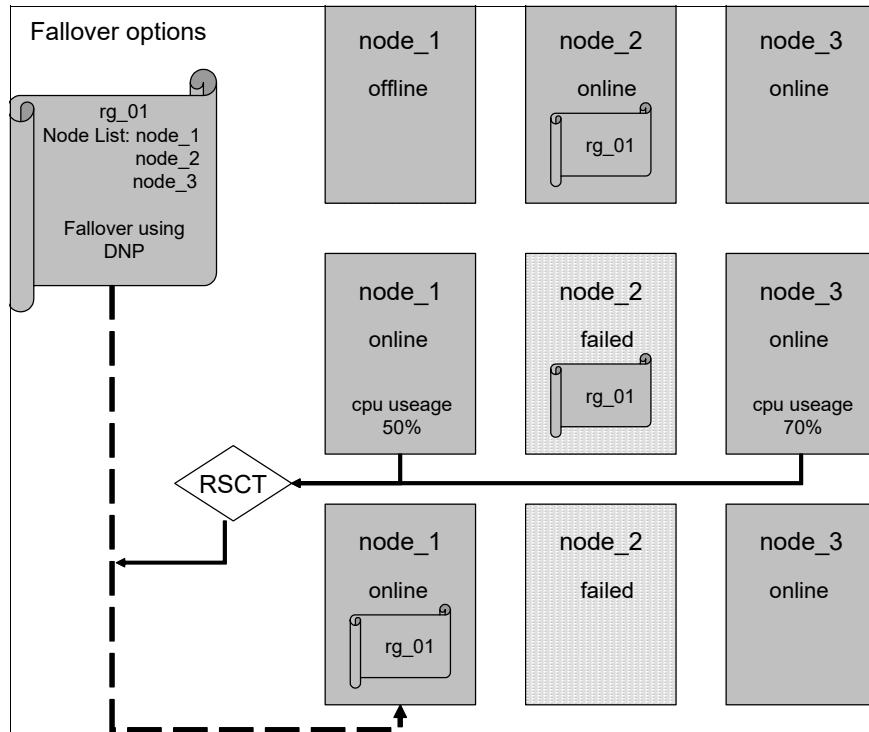


Figure 2-10 Fallback using dynamic node priority

In particular, select one of the following attributes for DNP:

- `cl_highest_free_mem` (Select the node with the highest percentage of free memory.)
- `cl_highest_idle_cpu` (Select the node with the most available processor time.)
- `cl_lowest_disk_busy` (Select the disk that is least busy.)

The PowerHA SystemMirror cluster manager queries the RMC subsystem every 3 minutes to obtain the current value of these attributes on each node and distributes them cluster-wide. The interval at which the queries of the RMC subsystem are performed (3 minutes) is not user-configurable. During a failover event of an RG with DNP configured, the most recently collected values are used in the determination of the best node to acquire the RG.

This policy applies only to RGs with three or more nodes. Introduced in PowerHA 7.1, you can choose the DNP based on the user-defined property by selecting one of the following attributes:

- `cl_highest_udscript_rc`
- `cl_lowest_nonzero_udscript_rc`

When you select one of these attributes, you must also provide values for the DNP script path and DNP timeout attributes for an RG. When the DNP script path attribute is specified, that script is invoked on all nodes and return values are collected from all nodes. The failover node decision is made by using these values and the specified criteria. If you select the `cl_highest_udscript_rc` attribute, collected values are sorted, and the node that returned the highest value is selected as a candidate node to fall over. If you select the `cl_lowest_nonzero_udscript_rc` attribute, collected values are sorted, and the node that returned lowest nonzero positive value is selected as a candidate node to failover. If the return values of the script from all nodes are the same or zero, the default node priority is considered. PowerHA verifies the script existence and the execution permissions during verification.

Demonstration: For a demonstration of user-defined node priority, see [YouTube](#).

► Bring offline, on error node only

The RG is brought offline if there is an error. This option is designed for RGs that are online on all available nodes. Figure 2-11 on page 59 show the process.

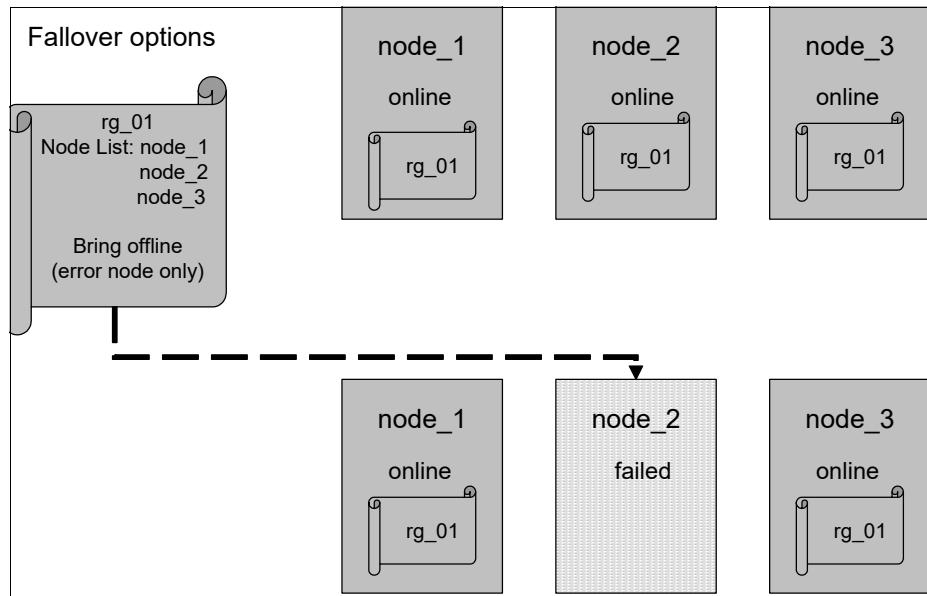


Figure 2-11 Bring offline on error node only

Fallback options

These options control the behavior of an online RG when a node joins the cluster:

- ▶ Fall back to higher priority node in list

The RG falls back to a higher priority node when it joins the cluster, as shown in Figure 2-12.

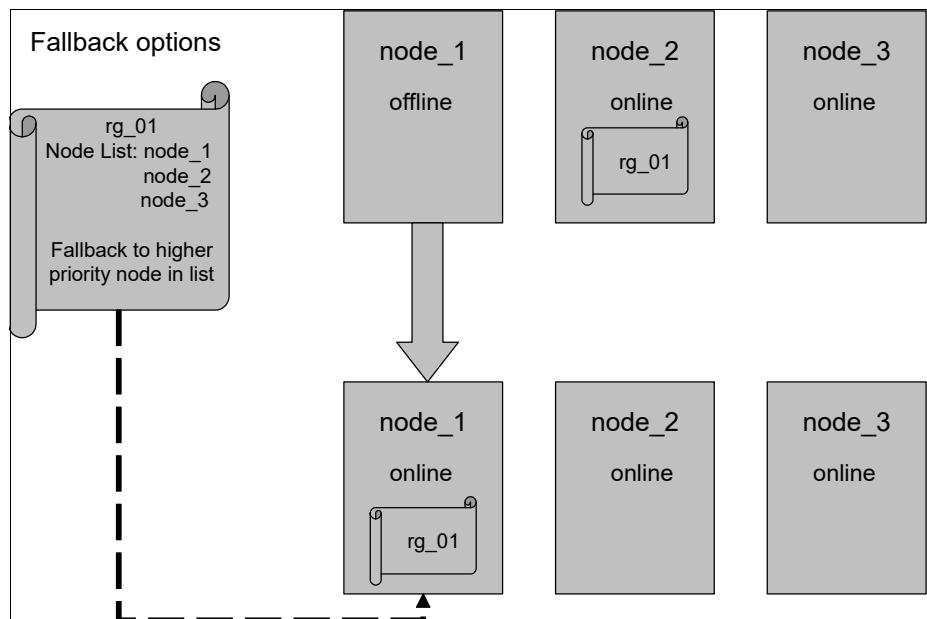


Figure 2-12 Fall back to higher priority node in list

- Never fall back

The RG does not move if a high priority node joins the cluster. RGs with the “Online on all available nodes” startup policy must be configured with this option, as shown in Figure 2-13.

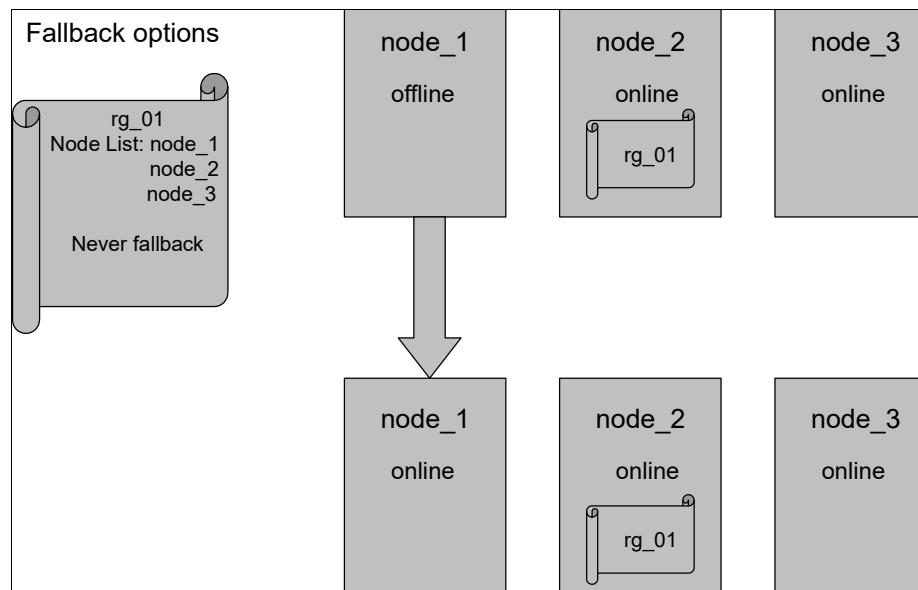


Figure 2-13 Never fall back

Resource group attributes

RG behavior can now be further tuned by setting RG attributes:

- Settling time
- Delayed fallback timers
- Distribution policy
- Dynamic node priorities
- RG processing order
- RG dependencies: parent/child
- RG dependencies: location
- RG dependencies: start after/stop after

Full more information, including how to configure RG attributes, and scenarios, see Chapter 10, “Extending resource group capabilities” on page 441.

Resource group manipulation

RGs can be manipulated in the following ways:

- Brought online

An RG can be brought online on a node in the RG node list. The RG would be offline unless it uses the “Online on all available nodes” startup policy.

- Brought offline

An RG can be taken offline from a particular node.

- Moved to another node while online

An RG that is online on one node can be taken offline and then brought online on another node in the RG node list. This action can include moving the RG to another site.

Certain changes are not allowed:

- ▶ A parent RG cannot be taken offline or moved if a child RG is in an online state.
- ▶ A child RG cannot be started until the parent RG is online.

Resource group states

PowerHA RG failures are handled in such a way that manual intervention is rarely required.

If a node fails to bring online an RG when it joins the cluster, the RG is left in the ERROR state. If the RG is not configured as “Online on all available nodes”, PowerHA attempts to bring online the RG on the other active nodes in the RG node list.

Each node that joins the cluster automatically attempts to bring online any of the RGs that are in the ERROR state.

If a node fails to acquire an RG during failover, the RG is marked as “recoverable” and PowerHA attempts to bring online the RG in all the nodes in the RGs node list. If this action fails for all nodes, the RG is in the ERROR state.

If there is a failure of a network on a particular node, PowerHA determines what RGs are affected (the ones that had service IP labels in the network) and then attempt to bring them online on another node. If no other nodes have the required network resources, the RGs remain in the ERROR state. If any interfaces become available, PowerHA works out what ERROR-state RGs can be brought online, and then attempts to do so.

Tip: If you want to override the automatic behavior of bringing an RG in the ERROR state back online, specify that it must remain offline on a node.

Selective failovers

The following failures are categorized as selective failover events, but they are all enabled by default:

- ▶ Interface failure:
 - PowerHA swaps interfaces if possible.
 - If not possible, the RG is moved to the highest priority node with an available interface, and if not successful, the RG is set to the ERROR state.
- ▶ Network failure:
 - If the failure is local, affected RGs are moved to another node.
 - If the failure is global, the result is `node_down` for all nodes.
- ▶ Application failure:
 - If an application monitor indicates that an application failed, depending on the configuration, PowerHA first attempts to restart the application on the same node (usually three times).
 - If a restart is not possible, PowerHA moves the RG to another node, and if this action fails also, the RG is set to the ERROR state.
- ▶ VG failure:
 - PowerHA attempts to move the RG to another node.
 - When an LVM_SA_QUORCLOSE error is encountered on a shared VG, PowerHA attempts to move the affected RGs to another node.

2.5 Smart assists

The PowerHA Smart Assist software contains sample scripts to help you configure the following applications as part of a highly available cluster. Table 2-2 shows the applications and version levels that are supported in PowerHA 7.1.3. To get the most recent information about the supported versions, check with your local IBM Support.

Table 2-2 Smart Assist applications

Middleware application	AIX 7.1 with Technology Level 5 or later	AIX 7.2 with Technology Level 2 or later	AIX 7.3 with Technology Level 0 or later
AIX print subsystem	7.1	7.2	7.3
Oracle Database Server	18C	19C	19C
SAP NetWeaver	7.52	7.52	7.52
Db2	11.1	11.5	11.5
IBM WebSphere MQ	9	9.2	9.5
IBM Tivoli® Directory Server	6.3	6.4	6.4
IBM Lotus Domino Server	9.0.1	9.0.1	9.0.1
MaxDB	7.9.08	7.9.08	7.9.10
IBM Spectrum Protect (IBM Tivoli Storage Manager)	NA	8.1.8	8.1.12

In addition to the list of provided Smart Assists in Table 2-2, you can build a customized Smart Assist program to manage other applications that are not in Table 2-2. The General Application Smart Assist (GASA) is a preinstalled Smart Assist that comes with PowerHA SystemMirror. Its intended purpose is to configure applications that do not already have target Smart Assists, but can be easily managed by using start and stop scripts. For more information about the process, see [Smart Assist development concepts](#).

2.6 Other features

This section lists several other available features in PowerHA.

2.6.1 Notifications

This section shows notification options in your PowerHA SystemMirror cluster. Notifications can be customized to meet your business requirements.

Error notification

This option uses the AIX error notification facility so that you can trap any specific error that is logged in the error report and run a custom notification method that a user provides.

Custom remote notification of events

You can define a notification method through the SMIT interface to issue a customized page in response to a cluster event. You can send a text message notification to any number of devices, including a cell phone, or you can send a notification to an email address.

You can use the verification automatic monitoring `cluster_notify` event to configure a PowerHA SystemMirror remote notification method to send a message in case of detected errors in a cluster configuration. The output of this event is logged in the `hacmp.out` file throughout the cluster on each node that is running cluster services.

You can configure any number of notification methods for different events and with different text or numeric messages and telephone numbers to dial. The same notification method can be used for several different events if the associated text message conveys enough information to respond to all the possible events that trigger the notification. This method also includes SMS message support.

After configuring the notification method, you can send a test message to be sure that the configurations are correct and that the expected message is sent for an event.

2.6.2 Rootvg system event

PowerHA SystemMirror 7.1 introduced system events. These events are handled by a subsystem that is named `clevmgrdES`. The rootvg system event allows for the monitoring of loss of access to the rootvg VG. By default, in the case of a loss of access, the event logs an entry in the system error log and restarts the system. If required, you can change this option in the SMIT menu to log only an event entry and not to restart the system.

Event monitoring is now at the kernel level. The following kernel extension, which is loaded by the `clevmgrdES` subsystem, monitors these events for the loss of rootvg:

```
/usr/lib/drivers/phakernmgr
```

For more information about how to check and change this option and its behavior, see 11.2, “System events” on page 480.

2.6.3 Capacity on Demand and dynamic logical partition support on failover

Capacity on Demand (CoD) is a facility of DLPAR. With CoD, you can activate preinstalled processors that are inactive and not paid for as resource requirements change.

The extra processors and memory, although physically present, are not used until you decide that the extra capacity that you need is worth the cost. This approach provides a fast upgrade in capacity to meet peak or unexpected loads.

PowerHA SystemMirror integrates with the dynamic logical partition (DLPAR) and CoD functions. You can configure cluster resources in a way where the LPAR with allocated resources serves as a standby node, and the application is on another LPAR node that has more resources than the standby node.

When it is necessary to run the application on the standby node, PowerHA SystemMirror ensures that the node has sufficient resources to successfully run the application and allocates the necessary resources.

For more information about using this feature, see 9.3, “Resource Optimized High Availability” on page 380.

2.6.4 File collections

Certain AIX and PowerHA SystemMirror configuration files, which are on each cluster node, must be kept in sync (be identical) for PowerHA SystemMirror to behave correctly. Such files include event scripts, application scripts, and some system and node configuration files.

By using the PowerHA SystemMirror file collection function, you can request that a list of files be automatically kept in sync across the cluster. You no longer have to manually copy an updated file to every cluster node, confirm that the file is properly copied, and confirm that each node has the same version of it. With PowerHA SystemMirror file collections enabled, PowerHA SystemMirror can detect and warn you if one or more files in a collection is deleted or has a zero value on one or more cluster nodes.

For more information, see 7.2, “File collections” on page 254.

2.6.5 PowerHA SystemMirror Enterprise Edition

PowerHA SystemMirror Enterprise Edition is a separate offering that can automate disaster recovery (DR) across sites. The key difference is the usage and management of data replication. In PowerHA Enterprise Edition 7.2.7, the following data replication methods are supported:

- ▶ GLVM:
 - Synchronous
 - Asynchronous
- ▶ DS8000 Copy Services:
 - Metro Mirror
 - Global Mirror
 - HyperSwap
- ▶ SAN Volume Controller (SVC) and IBM Spectrum Virtualized Storage:
 - Metro Mirror
 - Global Mirror
- ▶ XIV:
 - Synchronous replication
 - Asynchronous replication
- ▶ Dell EMC Symmetrix Remote Data Facility (SRDF):
 - Synchronous replication
 - Asynchronous replication
- ▶ Hitachi:
 - TrueCopy for synchronous replication
 - Hitachi Universal Replicator (HUR) for Asynchronous replication

2.7 Limits

This section lists several common PowerHA limits at the time of writing. These limits are presented in Table 2-3.

Table 2-3 PowerHA limits

Component	Maximum number or other limits
Nodes	16
Backup repository disks	6
RGs	256
Resources in an RG	512
VGs in an RG	512 (minus any other resources in the RG)
File systems in an RG	512 (minus any other resources in the RG)
Networks	48
Cluster IP addresses	256
Service IP labels in an RG	256 (minus the rest of the total IP addresses in the cluster)
Parent-child dependencies	Three levels maximum
Sites	2
Interfaces per node per network	7
Application controllers in an RG	512 (minus any other resources in the RG)
Application monitors	512
Persistent IP alias	1 per node per network
XD_data networks	4 per cluster
GLVM Modes	Synchronous, asynchronous, or nonconcurrent
GLVM Devices	All disks that are supported by AIX. They can be different types of disks.

Subnet requirements

The AIX kernel routing table supports multiple routes for the same destination. If multiple matching routes have the same weight, each subnet route is used alternately. The problem that this approach poses for PowerHA is that if one node has multiple interfaces that share the route, PowerHA has no means to determine its health. Therefore, as a best practice, each interface on a node should belong to a unique subnet so that each interface can be monitored.

2.8 Storage characteristics

This section presents information about storage characteristics, and PowerHA storage handling capabilities.

2.8.1 Shared LVM

For a PowerHA cluster, the key element is the data that is used by the highly available applications. This data is stored on AIX LVM entities. PowerHA clusters use the capabilities of the LVM to make this data accessible to multiple nodes. AIX LVM provides shared data access from multiple nodes.

Here are the components of the shared LVM:

- ▶ Shared VG: A VG that is entirely on the external disks that are shared by cluster nodes.
- ▶ Shared PV: A disk that is in a shared VG.
- ▶ Shared LV: An LV that is entirely in a shared VG.
- ▶ Shared file system: A file system that is entirely in a shared LV.

A system administrator of a PowerHA cluster might be asked to perform any of the following LVM-related maintenance tasks:

- ▶ Create a shared VG.
- ▶ Extend, reduce, change, or remove an existing VG.
- ▶ Create a shared LV.
- ▶ Extend, reduce, change, or remove an existing LV.
- ▶ Create a shared file system.
- ▶ Extend, change, or remove an existing file system.
- ▶ Add and remove PVs.

When performing any of these maintenance tasks on shared LVM components, make sure that ownership and permissions are reset when a VG is exported and then reimported. For more information about performing these tasks, see 7.4, “Shared storage management” on page 275.

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

Note: Applications, such as some database servers, that use raw LVs might be affected by this change if they change the ownership of the raw LV device. Restore the ownership and permissions back to what is needed after this sequence.

2.9 Shared storage configuration

Most PowerHA configurations require shared storage, which is those disk subsystems that support access from multiple hosts.

There are also third-party (OEM) storage devices and subsystems that can be used, although most of them are not directly certified by IBM for PowerHA usage. For these devices, check the manufacturer’s respective websites.

You can configure OEM VGs in AIX and use PowerHA SystemMirror to manage such VGs, and their corresponding file systems and application controllers. In particular, PowerHA SystemMirror automatically detects and provides the methods for VGs that are created with the VxVM by using VFS 4.0. For other OEM file systems, depending on the type of OEM volume, you can use custom methods in PowerHA SystemMirror (or an OEM vendor) to tell PowerHA SystemMirror that a file system that is unknown to AIX LVM should be treated the same way as a known and supported file system, or to specify the custom methods that provide the file systems processing functions that are supported by PowerHA SystemMirror.

PowerHA also supports shared tape drives (SCSI or FC). The shared tapes can be connected by using SCSI or FC. Concurrent mode tape access is *not* supported.

Storage configuration is one of the most important tasks that you must perform before starting the PowerHA cluster configuration. Storage configuration can be considered a part of PowerHA configuration.

Depending on the application needs, and on the type of storage, you decide how many nodes in a cluster have shared storage access, and which RGs use which disks.

2.9.1 Shared LVM requirements

Planning a shared LVM for a PowerHA cluster depends on the method of shared disk access and the type of shared disk device. Consider the following methods for a shared LVM:

- ▶ Data protection method
- ▶ Storage access method
- ▶ Storage hardware redundancy

Note: PowerHA does not provide data storage protection. Storage protection is provided by using these items:

- ▶ AIX (LVM mirroring)
- ▶ GLVM
- ▶ Hardware RAID

In this section, we provide information about data protection methods at the storage level, and also talk about the LVM shared disk access modes:

- ▶ Non-concurrent
- ▶ ECM

Both access methods use ECVGs. In a non-concurrent access configuration, only one cluster node can access the shared data concurrently. If the RG containing the shared disk space moves to another node, the new node activates the disks and checks the current state of the VGs, LVs, and file systems.

In non-concurrent configurations, the disks can be shared as these items:

- ▶ Raw PVs
- ▶ Raw LVs
- ▶ File systems

In a concurrent access configuration, data on the disks is available to all nodes concurrently. This access mode does not support file systems (either Journaled File System (JFS) or JFS2).

LVM requirements

The LVM component of AIX manages storage by coordinating data mapping between physical and logical storage. Logical storage can be expanded and replicated, and it can span multiple physical disks and enclosures.

The main LVM components are as follows:

- ▶ Physical volume

A PV represents a single physical disk as it is seen by AIX (hdisk). The PV is partitioned into physical partitions (PPs), which represent the physical allocation units that are used by LVM.

- ▶ Volume group

A VG is a set of PVs that AIX treats as a contiguous, addressable disk region. In PowerHA, the VG and all its LVs can be part of a shared RG. A VG cannot be part of multiple RGs.

- ▶ Physical partition

The PP is the allocation unit in a VG. The PVs are divided into PPs (when the PV is added to a VG), and the PPs are used for LVs (one, two, or three PPs per LPAR).

- ▶ Volume Group Descriptor Area

The VGDA is an area on the disk that contains information about the storage allocation in that VG.

For a single disk VG, there are two copies of the VGDA. For a two-disk VG, there are three copies of the VGDA: two on one disk and one on the other disk. For a VG consisting of three or more PVs, there is one VGDA copy on each disk in the VG.

- ▶ Quorum

For an active VG to be maintained as active, a quorum of VGDAs must be available (50% plus 1). Also, if a VG has the quorum option set to off, it cannot be activated (without the **force** option) if one VGDA copy is missing. If the quorum is turned off, the system administrator must know the mapping of that VG to ensure data integrity.

- ▶ Logical volume

An LV is a set of LPARs that AIX makes available as a single storage entity. The LVs can be used as raw storage space or as file system storage. In PowerHA, a LV that is part of a VG is already part of an RG, and cannot be part of another RG.

- ▶ Logical partition

The LPAR is the space allocation unit for LVs, and is a logical view of a PP. With AIX LVM, the LPARs can be mapped to one, two, or three PPs to implement LV mirroring.

- ▶ File system

The file system is a simple database for storing files and directories. A file system in AIX is stored on a single LV. The main components of the file system (JFS or JFS2) are the LV that holds the data, the file system log, and the file system device driver. PowerHA supports both JFS and JFS2 as shared file systems.

Forced varyon of volume groups

PowerHA provides a facility to use the forced varyon of VG option on a node. If during the takeover process the normal **varyon** command fails on that VG (due to lack of quorum), PowerHA ensures that at least one valid copy of each LPAR for every LV in that VG is available before varying on that VG on the takeover node.

By forcing a VG to vary on, you can bring and keep a VG online (as part of an RG) with one copy of the data that is available. Use a forced varyon of VG option only for VGs that have mirrored LVs. However, be cautious when using this facility to avoid creating a partitioned cluster.

Note: Also specify the superstrict allocation policy for all LVs in VGs that are used with the forced varyon of VG option. This way, the LVM ensures that the copies of a LV are always on separate disks, and increases the chances that forced varyon of VG is successful after a failure of one or more disks.

This option is useful in a takeover situation if a VG that is part of that RG loses one or more disks (VGDA). If this option is not used, the RG is not activated on the takeover node, thus rendering the application unavailable.

When using the forced varyon of VG option in a takeover situation, PowerHA first tries a normal **varyonvg** command. If this attempt fails because of lack of quorum, PowerHA checks the integrity of the data to ensure that at least one available copy of all data is in the VG before trying to force the volume online. If there is, it runs the **varyonvg -f** command. If not, the VG remains offline and the RG action results in an error state.

Note: The forced varyon of VG feature is specific to cross-site LVM and GLVM configurations.

2.10 PowerHA cluster events

Considering all the involved components, the PowerHA solution provides ways to monitor almost any part of the cluster structure. Also, according to the output of these monitoring methods, the PowerHA cluster takes an automatic action that can be a notification or even an RG failover.

With PowerHA, you can customize predefined cluster events and create events. When you create events, an important step is to check whether any standard event exists that covers the relevant action or situation.

All standard cluster events have their own meaning and functioning behavior. Some of the most common examples of cluster events are listed in Table 2-4.

Table 2-4 Examples of standard cluster events

Event name	Event type	Description summary
node_up	Nodes joining or leaving a cluster	The node_up event starts when a node joins or rejoins the cluster.
node_down	Nodes joining or leaving a cluster	The node_down event starts when a cluster is not receiving heartbeats from a node. and it considers the node gone and starts a node_down event.
network_up	Nodes joining or leaving a cluster	The network_up event starts when a cluster detects that a network is available and ready for cluster usage (for example, for a service IP address activation).

Event name	Event type	Description summary
start_server	RG startup, and application monitoring restart	The start_server event runs either when an RG with a specified application controller is brought online, or when application monitoring detects that a start or restart is required.
stop_server	RG startup, application monitoring restart	The stop_server event runs either when an RG with a specified application controller is brought offline, or when application monitoring detects that a start or restart is required.
network_down	Network-related events	The network_down event starts when a specific network becomes unreachable. The event can be network_down_local, when only a specific node lost its connectivity for a network, or network_down_global, when all nodes lost connectivity.
swap_adapter	Network-related events	The swap_adapter event starts when the interface that hosts one service IP address experiences a failure. If other boot networks are available on the same node, then the swap_adapter event moves the service IP address to another boot interface and refreshes the network routing table.
fail_interface	Interface related issues	The fail_interface event starts when any node interface experiences a failure. If the interface has no service IP address that is defined, only the fail_interface event runs. If the failing interface hosts a service IP address and there is no other boot interface that is available to host it, then an rg_move event runs.
join_interface	Interface related issues	The join_interface event starts when a boot interface becomes available or when it recovers itself from a failure.
fail_standby	Interface related issues	The fail_standby event starts when a boot interface, hosting no service IP address, faces a failure.
join_standby	Interface related issues	The join_standby event starts when a boot interface becomes available or when it recovers itself from a failure.
rg_move	RG changes	The rg_move event starts when an RG operation from one node to another one starts.
rg_up	RG changes	The rg_up event starts when an RG is successfully brought online at a node.
rg_down	RG changes	The rg_down event starts when an RG is brought offline.

Note: All events have a detailed usage description in the script file. All standard events are in the /usr/es/sbin/cluster/events directory.



Part 2

Planning, installation, and migration

In Part 2, we provide information about PowerHA cluster and environment planning, and explain how to install a sample cluster. We also present examples for migrating a cluster from an earlier PowerHA version to the latest PowerHA version. Our scenarios provide step-by-step instructions and comments, and also some problem determination for migration.

This part contains the following chapters:

- ▶ Chapter 3, “Planning” on page 73
- ▶ Chapter 4, “Installation and configuration” on page 133
- ▶ Chapter 5, “Migration” on page 159



Planning

In this chapter, we describe the planning aspects for a PowerHA 7.2.7 cluster. Planning and preparation are necessary to successfully install and maintain a PowerHA cluster. Time that is spent planning your cluster configuration and preparing your environment results in a cluster that is easier to install and maintain and provides higher application availability.

Before you begin planning the cluster, you must understand your current environment, your application, and your expected behavior for PowerHA. Building on this information, you can develop an implementation plan that helps you to more easily integrate PowerHA into your environment, and have PowerHA manage your application availability to your expectations.

This chapter contains the following topics:

- ▶ High availability planning
- ▶ Planning for PowerHA
- ▶ Planning cluster hardware
- ▶ Planning cluster software
- ▶ Operating system considerations
- ▶ Planning security
- ▶ Planning cluster networks
- ▶ Planning storage requirements
- ▶ Application planning
- ▶ Planning for resource groups
- ▶ Detailed cluster design
- ▶ Developing a cluster test plan
- ▶ Developing a PowerHA 7.2.7 installation plan
- ▶ Backing up the cluster configuration
- ▶ Documenting the cluster
- ▶ Change and problem management
- ▶ Planning tools

3.1 High availability planning

The primary goal of planning a high availability (HA) cluster solution is to eliminate or minimize service interruptions for any specific applications. To achieve that goal, single points of failure (SPOFs), in both hardware and software, must be addressed and eliminated where possible. You can eliminate SPOFs by using redundant hardware, such as power supplies, network interfaces, storage area network (SAN) adapters, and mirrored or redundant array of independent disks (RAID) disks. All these components come with added expense and might not protect the application if a server or operating system fails.

PowerHA can be configured to monitor server hardware, the operating system, and application components. In a failure, PowerHA can take corrective actions, such as moving specified resources (service IP addresses, storage, and applications) to surviving cluster components to restore application availability as quickly as possible.

Because PowerHA is a flexible product, designing a cluster to fit your organization requires thorough planning. Knowing your application requirements and behavior provides important inputs to your PowerHA plan, and they are primary factors in determining the cluster design. Ask yourself the following questions while developing your cluster design:

- ▶ Which application services must be highly available?
- ▶ What are the service-level requirements for these application services (always available?), and how quickly must service be restored if a failure occurs?
- ▶ What are the potential points of failure in the environment, and how can they be addressed?
- ▶ Which points of failure can be automatically detected by PowerHA, and which require custom code to be written to trigger an event?
- ▶ What is the skill level within the group implementing and maintaining the cluster?

Although AIX system administrators are responsible for the implementation of PowerHA, they cannot do it alone. A team consisting of the following representatives should be assembled to assist with the PowerHA planning. Each member has a role in the success of the cluster:

- ▶ Network administrator
- ▶ AIX system administrator
- ▶ Database administrator
- ▶ Application programmer
- ▶ Support personnel
- ▶ Application users

3.2 Planning for PowerHA

The major steps to a successful PowerHA implementation are shown in Figure 3-1 on page 75. A cluster implementation does not end with the successful configuration of a cluster. Cluster testing, backup, documentation, and system and change management procedures are important to ensure the ongoing integrity of the cluster.

Using the concepts that were described in Chapter 1, “Introducing IBM PowerHA SystemMirror for AIX” on page 3, begin the PowerHA implementation by developing a detailed PowerHA cluster configuration and implementation plan.

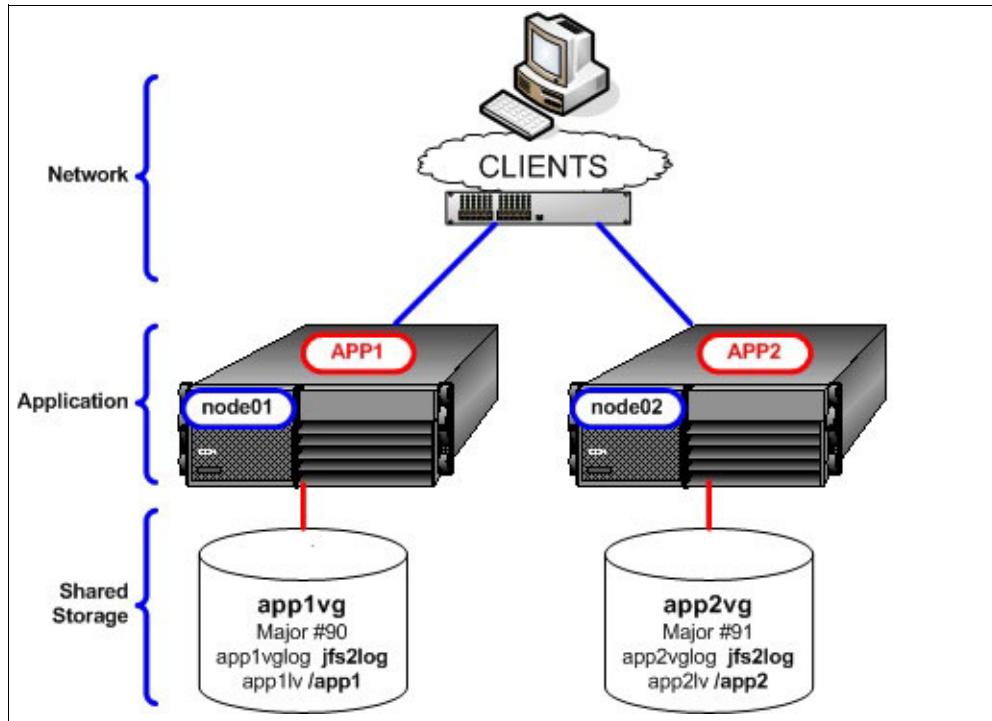


Figure 3-1 Initial environment

3.2.1 Planning strategy and example

Planning provides the foundation on which the implementation is built. Proper planning should touch on all aspects of cluster implementation and include these factors:

- ▶ The cluster design and behavior
- ▶ A detailed cluster configuration
- ▶ Installation considerations and plans
- ▶ A plan to test the integrity of the cluster
- ▶ A backup strategy for the cluster
- ▶ A procedure for documenting the cluster
- ▶ A plan to manage problems and changes in the cluster

For simplicity, we use the planning of a simple 2-node mutual takeover cluster as an example. Sample planning worksheets are included as we work through this chapter so that you can see how the cluster planning is developed.

3.2.2 Planning tools

The following tools are available to help with the planning of a PowerHA 7.2.7 cluster:

- ▶ Cluster diagram
- ▶ Paper planning worksheets
- ▶ Cluster report of an existing cluster

Both the cluster diagram and the paper planning worksheets provide a manual method of recording your cluster information. A set of planning worksheets is in Appendix A, “Paper planning worksheets” on page 609.

3.2.3 Getting started

Begin cluster planning by assessing the environment and your expectations for PowerHA. Here are some questions that you might ask:

- ▶ Which applications must be highly available?
- ▶ How many nodes are required to support the applications?
- ▶ Are the existing nodes adequate in size (CPU and memory) to run multiple applications, or is it a new installation?
- ▶ How do the clients connect to the application, and what is the network configuration?
- ▶ What type of shared disk is used?
- ▶ What are the expectations for PowerHA?

3.2.4 Current environment

Figure 3-1 on page 75 illustrates a simple starting configuration that we use as our example. It focuses on two applications to be made highly available, which might be an existing pair of servers or two new servers. A server is a representation of an AIX image running on IBM Power hardware. A best practice is for these “servers” to be logical partitions (LPARs).

The starting configuration shows this information:

- ▶ Each application is on a separate node (server).
- ▶ Clients access each application over a dedicated Ethernet connection on each server.
- ▶ Each node is relatively the same size in terms of CPU and memory, each with extra spare capacity.
- ▶ Each node has redundant power supplies and mirrored internal disks.
- ▶ The applications are on an external SAN disk.
- ▶ The applications each have their own start and stop scripts.
- ▶ There is a monitoring tool to verify the health of each application.
- ▶ AIX 7.x is installed.

Important: Each application that is integrated into the cluster must run in stand-alone mode. You must also be able to fully control the application (start, stop, and validation test).

The intention is to use the two nodes in a mutual takeover configuration where app1 normally is on Node01, and app2 normally is on Node02. In a failure, we want both applications to run on the surviving server. As you can see from the diagram, we must prepare the environment so that each node can run both applications.

Note: Each application that is integrated into the cluster must be able to run in stand-alone mode on any node that it might have to run on (under both normal and failover situations).

After analyzing the PowerHA cluster requirements, we discover three key focus areas, as shown in Figure 3-1 on page 75: network, application, and storage. All planning activities support one of these three items to some extent:

Network	How clients connect to the application (the service address). The service address floats between all designated cluster nodes.
Application	What resources are required by the application. The application must have all that it needs to run on a failover node, including CPU and memory resources, licensing, runtime binary files, and configuration data. It should have start and stop scripts and a tool to monitor its status.
Storage	What type of shared disks are used. The application data must be on shared disks that are available to all cluster nodes.

3.2.5 Addressing single points of failure

Table 3-1 summarizes the various SPOFs that are found in the cluster infrastructure and how to protect against them. Consider these items during the development of the detailed cluster design.

Table 3-1 Single points of failure

Cluster objects	How to eliminate the object as a single point of failure	PowerHA 7.2.7 and AIX supports
Nodes	Use multiple nodes.	Up to 16
Power sources	Use multiple circuits or uninterruptible power supplies (UPS).	As many as needed
Networks	Use multiple networks to connect nodes.	Up to 48
Network interfaces, devices, and IP addresses.	Use redundant network adapters.	Up to 256
TCP/IP subsystem	Use point-to-point networks to connect adjoining nodes.	As many as needed
Disk adapters	Use redundant disk adapters.	As many as needed
Storage controllers	Use redundant disk controllers.	As many as needed (hardware-limited)
Disks	Use redundant hardware and disk mirroring, striping, or both.	As many as needed
Applications	Assign a node for application takeover, configure application monitors, and configure clusters with nodes at more than one site.	As many as needed
Sites	Use more than one site for disaster recovery (DR).	2

3.2.6 Initial cluster design

Now that you understand the current environment, PowerHA concepts, and your expectations for the cluster, you can begin the cluster design.

As a best practice, create a diagram of the PowerHA cluster. Start simply, and gradually increase the level of detail as you go through the planning process. The diagram can help identify SPOFs, application requirements, and guide you through the planning process.

Also, use the paper or online planning worksheets to record the configuration and cluster details as you go.

Figure 3-2 illustrates the initial cluster diagram that is used in our example. Now, the focus is on high-level cluster functions. Cluster details are developed as we move through the planning phase.

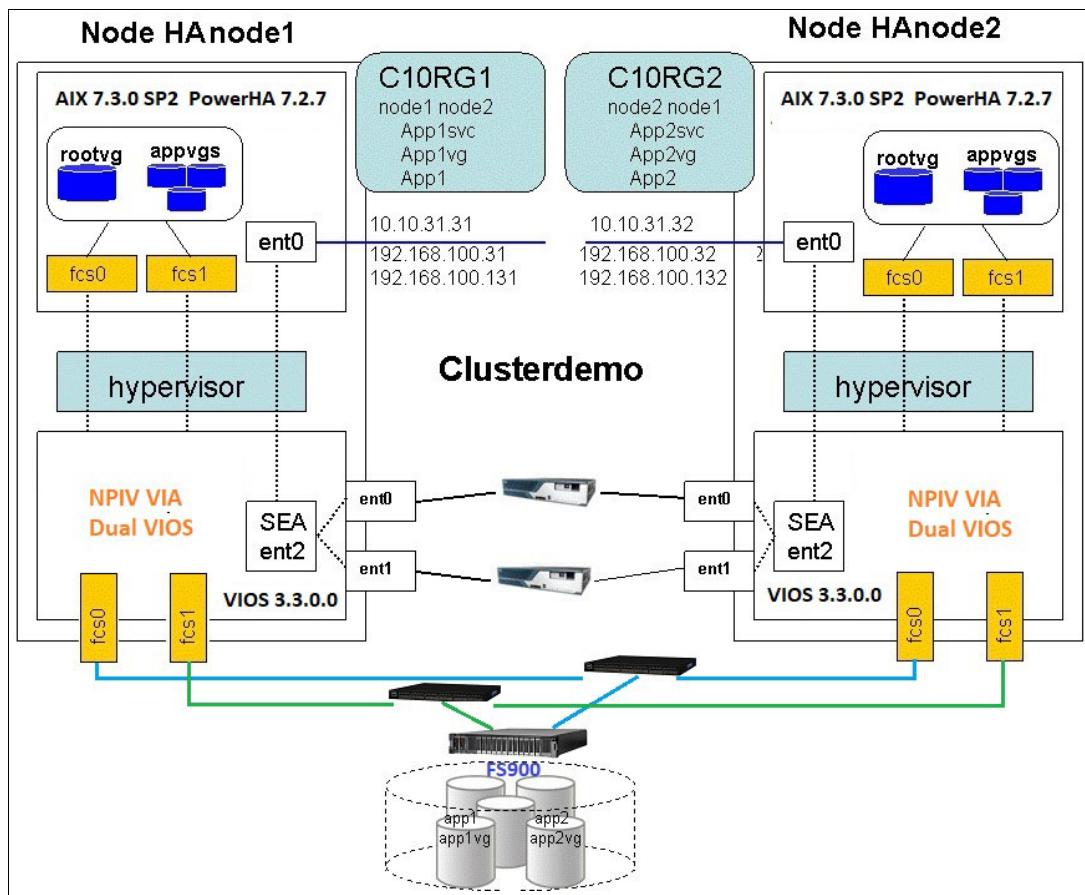


Figure 3-2 Initial cluster design

We make design decisions for the cluster topology and behavior based on our requirements. For example, based on our requirements, the initial cluster design for our example includes the following considerations:

- ▶ The cluster is a two-node mutual takeover cluster.
- ▶ Although hostnames can be used as cluster node names, we choose to specify cluster node names instead.

Note: A key configuration requirement is that the LPAR partition name, the cluster node name, and AIX hostname must match. PowerHA makes these assumptions.

- ▶ Each node contains one application but can run both (if there is sufficient network, storage, memory, CPU, and software resources).
- ▶ Each node has one logical Ethernet interface that is protected by using Shared Ethernet Adapter (SEA) in a Virtual I/O Server (VIOS).
- ▶ IP Address Takeover (IPAT) through aliasing is used.
- ▶ Each node has a persistent IP address (an IP alias that is always available while the node is up) and one service IP (aliased to one of the adapters under PowerHA control). The base Ethernet adapter addresses are on separate subnets.
- ▶ Shared disks are virtual SCSI devices that are provided by a VIOS. The disks are on a SAN and available on both nodes.
- ▶ All volume groups (VGs) on the shared disks are created in enhanced concurrent mode (ECM), as required in PowerHA.
- ▶ Each node has enough CPU and memory resources to run both applications.
- ▶ Each node has redundant hardware and mirrored internal disks.
- ▶ AIX 7.2 Technology Level (TL) 3 Service Pack (SP) 3 is installed.
- ▶ PowerHA 7.2.7 is used.

This list captures the basic components of the cluster design. Each item is investigated in further detail as we progress through the planning stage.

3.2.7 Naming conventions

When you plan a cluster, many components of the cluster might have customized naming. However, there are certain reserved words that should *not* be used by themselves, but they can be combined with a prefix or suffix.

Reserved words

The list of reserved words can be found in /usr/es/sbin/cluster/etc/reserved_words and are shown below:

- ▶ adapter
- ▶ cluster
- ▶ command
- ▶ custom
- ▶ daemon
- ▶ event
- ▶ group
- ▶ network
- ▶ node
- ▶ resource
- ▶ name
- ▶ grep
- ▶ subnet
- ▶ nim
- ▶ ip
- ▶ IP
- ▶ ether

- ▶ token
- ▶ rs232
- ▶ socc
- ▶ fddi
- ▶ slip
- ▶ tmscsi
- ▶ fcs
- ▶ hps
- ▶ atm
- ▶ tmssa
- ▶ serial
- ▶ public
- ▶ private
- ▶ diskhb
- ▶ diskhbmulti
- ▶ alias
- ▶ disk
- ▶ volume
- ▶ vpath
- ▶ tty
- ▶ scsi
- ▶ fscsi
- ▶ vscsi
- ▶ nodename
- ▶ OHN
- ▶ OFAN
- ▶ OUDP
- ▶ OAAN
- ▶ FNPN
- ▶ FUDNP
- ▶ BO
- ▶ FBHP
- ▶ NFB
- ▶ ipv6
- ▶ IPv6
- ▶ IPV6
- ▶ IW
- ▶ ALL
- ▶ all

3.2.8 Completing the cluster overview planning worksheet

Complete the initial worksheet. This chapter has 11 worksheets, each covering separate aspects of the cluster planning. Table 3-2 shows the first worksheet, which lists the basic cluster elements.

Table 3-2 Cluster overview

PowerHA 7.2.7 Cluster Worksheet - Part 1 Of 11 Cluster Overview		Date: November 2022
Cluster Name	clusterdemo	
Organization	IBM ITSO	
Node1 Hostname	HAnode1	

PowerHA 7.2.7 Cluster Worksheet - Part 1 Of 11 Cluster Overview		Date: November 2022
Node2 Hostname	HAnode2	
Node1 PowerHA Node Name	Node01	
Node2 PowerHA Node Name	Node02	
Comments	A set of planning tables for a simple two-node PowerHA mutual takeover cluster that uses IPAT through aliasing.	

3.3 Planning cluster hardware

Cluster design starts by determining how many and what type of nodes are required, which depends largely on a couple of factors:

- ▶ The amount of resources that is required by each application
- ▶ The failover behavior of the cluster

Note: The number of nodes in a cluster can be 2 - 16. CAA supports 32 nodes, but PowerHA supports only 16 nodes.

3.3.1 Overview of cluster hardware

A primary consideration when choosing nodes is that in a failover situation, the surviving node or nodes must be capable of running the failing node's applications. For example, if you have a two-node cluster and one node fails, the surviving node must have all the resources that are required to run the failing node's applications (in addition to its own applications). If this situation is not possible, you might consider implementing an extra node as a standby node or using the dynamic logical partition (DLPAR) feature. PowerHA allows for a wide range of cluster configurations depending on your requirements.

PowerHA supports virtually any AIX supported node, from desktop systems to high-end servers. When choosing a type of node, consider this information:

- ▶ Ensure that sufficient CPU and memory resources are available on all nodes to allow the system to behave as you want it to in a failover situation. The CPU and memory resources must be capable of sustaining the selected applications during failover; otherwise, clients might experience performance problems. If you are using LPARs, you might want to use the DLPAR capabilities to increase resources during failover. If you are using stand-alone servers, you do not have this option, so you might have to look at using a standby server.
- ▶ Use highly available hardware and redundant components where possible in each server. For example, use redundant power supplies and connect them to separate power sources.
- ▶ Protect each node's rootvg (local operating system copy) by using mirroring or RAID.
- ▶ Allocate at least two Ethernet adapters per node and connect them to separate switches to avoid a single adapter or switch failure. This task is accomplished by using a single or dual VIOS.
- ▶ Allocate two SAN adapters per node to protect from a single SAN adapter failure. This task is accomplished by using a single or dual VIOS.

As a best practice, use cluster nodes with similar hardware configurations so that you can more easily distribute the resources and perform administrative operations. Do not try to fall over from a high-end enterprise class server to a scale-out model and expect everything to work.

Tip: For a list of supported devices by PowerHA, see the [Hardware Support Matrix](#).

3.3.2 Completing the cluster hardware planning worksheet

The following worksheet (Table 3-3) contains the hardware specifications for our example. Where possible, we use redundant hardware (extra Ethernet and SAN switches), and we ensure that we have enough resources to sustain both applications simultaneously on any node.

Table 3-3 Cluster hardware

PowerHA 7.2.7 Cluster Worksheet - Part 2 Of 11 Cluster Hardware		Date: November 2022
Hardware Component	Specifications	Comments
IBM Power9® technology-based 9009-22A - S922s	IBM Power server with 16 CPUs and 256 GB of memory	Quantity: 2 Latest firmware
Ethernet Adapters	Feature Code EN0H 2-port 10 Gb	Virtual network interface card (vNIC)
Network Switches		All ports are configured in the same virtual local area network (VLAN). The switches support gratuitous Address Resolution Protocol (ARP). The Spanning Tree Protocol is disabled. The switch port speed is set to auto as needed for Gigabit adapters.
SAN Adapters	Feature Code EN0A PCIe3 2-Port 16 Gb FC Adapter	N_Port ID Virtualization (NPIV)
SAN Switches	(2) IBM 2498 SAN24B-5	Zoned for NPIV client worldwide port names (WWPNs)
SAN Storage	IBM 2076-624 V7000	Switch attached (but not shown in the diagram)
Comments	All hardware compatibility is verified.	

3.4 Planning cluster software

Review all software components that you will use in the cluster to ensure compatibility. Items to consider are AIX, Reliable Scalable Cluster Technology (RSCT), PowerHA, VIOS, application, and storage software. This section describes the various software levels and compatibilities.

3.4.1 AIX and RSCT levels

Specific combinations of AIX and RSCT levels are required for installing PowerHA 7.2.7, as listed in Table 3-4.

Table 3-4 AIX and RSCT levels

AIX version	RSCT version
AIX 7.1 TL5 SP10	RSCT 3.2.3.0
AIX 7.2 TL1 SP6	RSCT 3.2.3.0
AIX 7.3 SP2	RSCT 3.3.0.0
AIX 7.3 TL1 SP1	RSCT 3.3.1.0

3.4.2 Virtual Ethernet and vSCSI support

PowerHA supports virtualization that is provided by PowerVM. For more information about VIOS support, see Chapter 9, “IBM PowerHA and IBM PowerVM” on page 373.

3.4.3 Required AIX file sets

The following file sets are required for PowerHA 7.2.7. Install them with the latest version of the fixes for the appropriate AIX level *before* PowerHA 7.2.7 is installed.

- ▶ bos.adt.lib
- ▶ bos.adt.libm
- ▶ bos.ahafs
- ▶ bos.cluster
- ▶ bos.clvm.enh
- ▶ bos.adt.syscalls
- ▶ bos.net.tcp.client
- ▶ bos.net.tcp.server
- ▶ bos.rte.SRC
- ▶ bos.rte.libc
- ▶ bos.rte.libcfg
- ▶ bos.rte.libcurl
- ▶ bos.rte.libpthread
- ▶ bos.rte.odm
- ▶ bos.rte.lvm.
- ▶ clic.rte (for the secure encryption communication option of `clicomd`)
- ▶ devices.common.IBM.storfwk (for the SAN heartbeat)

Requirements for Network File System v4

The `cluster.es.nfs` file set that is included with the PowerHA installation medium installs the Network File System (NFS) v4 support for PowerHA, including an NFS Configuration Assistant. To install this file set, the following base operating system (BOS) NFS components must also be installed on the system:

- ▶ `bos.net.nfs.server`
- ▶ `bos.net.nfs.client`

3.4.4 PowerHA 7.2.7 file sets

The following PowerHA 7.2.7 Standard Edition file sets can be installed from the installation media (excluding additional language file sets):

- ▶ cluster.adt.es
 - cluster.adt.es.client.include
 - cluster.adt.es.client.samples.clinfo
 - cluster.adt.es.client.samples.clstat
 - cluster.adt.es.client.samples.libcl
- ▶ cluster.doc.en_US.assist
 - cluster.doc.en_US.assist.smartassists.pdf
- ▶ cluster.doc.en_US.es.
 - cluster.doc.en_US.es.pdf
- ▶ cluster.doc.en_US.glvm
 - cluster.doc.en_US.glvm.pdf
- ▶ cluster.doc.en_US.pprc
 - cluster.doc.en_US.pprc.pdf
- ▶ cluster.es.assist (by separate Smart Assist LPP)
 - cluster.es.assist.common
 - cluster.es.assist.db2
 - cluster.es.assist.dhcp
 - cluster.es.assist.dns
 - cluster.es.assist.domino
 - cluster.es.assist.filenet
 - cluster.es.assist.ihc
 - cluster.es.assist.maxdb
 - cluster.es.assist.oraappsrv
 - cluster.es.assist.oracle
 - cluster.es.assist.printServer
 - cluster.es.assist.sap
 - cluster.es.assist.tds
 - cluster.es.assist.tsmaadmin
 - cluster.es.assist.tsmaclient
 - cluster.es.assist.tsmaserver
 - cluster.es.assist.websphere
 - cluster.es.assist.wmq
- ▶ cluster.es.client
 - cluster.es.client.clcomd
 - cluster.es.client.lib
 - cluster.es.client.rte
 - cluster.es.client.utils
- ▶ cluster.es.cspoc
 - cluster.es.cspoc.cmds
 - cluster.es.cspoc.rte
- ▶ cluster.es.migcheck
- ▶ cluster.es.nfs
 - cluster.es.nfs.rte (NFSv4 support)

- ▶ cluster.es.server
 - cluster.es.server.diag
 - cluster.es.server.events
 - cluster.es.server.rte
 - cluster.es.server.testtool
 - cluster.es.server.utils
- ▶ cluster.es.smui
 - cluster.es.smui.agent
 - cluster.es.smui.common
- ▶ cluster.es.smui.server
- ▶ cluster.license
- ▶ cluster.man.en_US.es
 - cluster.man.en_US.es.data
- ▶ cluster.msg.Fr_FR.assist
- ▶ cluster.msg.Fr_FR.es
 - cluster.msg.Fr_FR.es.client
 - cluster.msg.Fr_FR.es.server
- ▶ cluster.msg.Ja_JP.assist
- ▶ cluster.msg.Ja_JP.es
 - cluster.msg.Ja_JP.es.client
 - cluster.msg.Ja_JP.es.server
- ▶ cluster.msg.en_US.assist
- ▶ cluster.msg.en_US.es
 - cluster.msg.en_US.es.client
 - cluster.msg.en_US.es.server
- ▶ cluster.msg.fr_FR.assist
- ▶ cluster.msg.fr_FR.es
 - cluster.msg.fr_FR.es.client
 - cluster.msg.fr_FR.es.server
- ▶ cluster.msg.ja_JP.assist
- ▶ cluster.msg.ja_JP.es
 - cluster.msg.ja_JP.es.client
 - cluster.msg.ja_JP.es.server

If you use the installation media of PowerHA 7.2.7 Enterprise Edition, the following additional file sets are available:

- ▶ cluster.es.cgpprc
 - cluster.es.cgpprc.cmds
 - cluster.es.cgpprc.rte
- ▶ cluster.es.genxd
 - cluster.es.genxd.cmds
 - cluster.es.genxd.rte

- ▶ cluster.es.pprc
 - cluster.es.pprc.cmds
 - cluster.es.pprc.rte
- ▶ cluster.es.spprc
 - cluster.es.spprc.cmds
 - cluster.es.spprc.rte
- ▶ cluster.es.sr
 - cluster.es.sr.cmds
 - cluster.es.sr.rte
- ▶ cluster.es.svcpprc
 - cluster.es.svcpprc.cmds
 - cluster.es.svcpprc.rte
- ▶ cluster.es.tc
 - cluster.es.tc.cmds
 - cluster.es.tc.rte
- ▶ cluster.msg.En_US.cgpprc
- ▶ cluster.msg.En_US.genxd
- ▶ cluster.msg.En_US.pprc
- ▶ cluster.msg.En_US.sr
- ▶ cluster.msg.En_US.svcpprc
- ▶ cluster.msg.En_US.tc
- ▶ cluster.msg.Fr_FR.assist
- ▶ cluster.msg.Fr_FR.cgpprc
- ▶ cluster.msg.Fr_FR.genxd
- ▶ cluster.msg.Fr_FR.glvm
- ▶ cluster.msg.Fr_FR.pprc
- ▶ cluster.msg.Fr_FR.sr
- ▶ cluster.msg.Fr_FR.svcpprc
- ▶ cluster.msg.Fr_FR.tc
- ▶ cluster.msg.Ja_JP.cgpprc
- ▶ cluster.msg.Ja_JP.genxd
- ▶ cluster.msg.Ja_JP.glvm
- ▶ cluster.msg.Ja_JP.pprc
- ▶ cluster.msg.Ja_JP.sr
- ▶ cluster.msg.Ja_JP.svcpprc
- ▶ cluster.msg.Ja_JP.tc
- ▶ cluster.msg.en_US.cgpprc
- ▶ cluster.msg.en_US.genxd
- ▶ cluster.msg.en_US.glvm
- ▶ cluster.msg.en_US.pprc
- ▶ cluster.msg.en_US.sr

- ▶ cluster.msg.en_US.svcpprc
- ▶ cluster.msg.en_US.tc
- ▶ cluster.msg.fr_FR.cgpprc
- ▶ cluster.msg.fr_FR.genxd
- ▶ cluster.msg.fr_FR.glvm
- ▶ cluster.msg.fr_FR.pprc
- ▶ cluster.msg.fr_FR.sr
- ▶ cluster.msg.fr_FR.svcpprc
- ▶ cluster.msg.ja_JP.cgpprc
- ▶ cluster.msg.ja_JP.genxd
- ▶ cluster.msg.ja_JP.glvm
- ▶ cluster.msg.ja_JP.pprc
- ▶ cluster.msg.ja_JP.svcpprc
- ▶ cluster.msg.ja_JP.sr
- ▶ cluster.msg.ja_JP.tc
- ▶ cluster.xd.base
- ▶ cluster.xd.glvm
- ▶ cluster.xd.license

3.4.5 AIX files that can be altered by PowerHA

The following system files can be altered by PowerHA during the cluster packages installation, verification, and synchronization process.

/etc/hosts

The cluster event scripts use the /etc/hosts file for name resolution. All cluster node IP interfaces must be added to this file on each node. PowerHA can modify this file to ensure that all nodes have the necessary information in their /etc/hosts file for proper PowerHA operations.

If you delete service IP labels from the cluster configuration by using System Management Interface Tool (SMIT), also remove them from /etc/hosts.

/etc/inittab

The /etc/inittab file is modified in each of the following cases:

- ▶ PowerHA is installed:

The following line is added when you initially install PowerHA. It starts the clcomdES and clstrmgrES subsystems if they are not already running.

```
clcomd:23456789:once:/usr/bin/startsrc -s clcomd
hacmp:2:once:/usr/es/sbin/cluster/etc/rc.init >/dev/console 2>&1
```

Important: This PowerHA entry is used to start the following daemons with the `startsrc` command if they are not already running:

- ▶ `startsrc -s syslogd`
- ▶ `startsrc -s snmpd`
- ▶ `startsrc -s clstrmgrES`

- ▶ If PowerHA is set to start at system restart, add the following line to the `/etc/inittab` file:
`hacmp6000:2:wait:/usr/es/sbin/cluster/etc/rc.cluster -boot -b -A # Bring up Cluster`

Notes:

- ▶ Although starting cluster services from the `inittab` file is possible, as a best practice, do *not* use this option. The better approach is to manually control the startup of PowerHA. For example, in a node failure, investigate the cause of the failure before restarting PowerHA on the node.
- ▶ `ha_star` is also found as an entry in the `inittab` file. This file set is delivered with the `bos.rte.control` file set and not PowerHA.

/etc/rc.net

The `/etc/rc.net` file is called by `cfgmgr`, which is the AIX utility that configures devices and optionally installs device software into the system to configure and start TCP/IP during the boot process. It sets the hostname, default gateway, and static routes.

/etc/services

PowerHA uses the following network ports for communication between cluster nodes. They are listed in the `/etc/services` file, as shown in Example 3-1.

Example 3-1 /etc/services file entries

<code>drmsfsd</code>	<code>4098/tcp/udp</code>
<code>caa_cfg</code>	<code>6181/tcp</code> - Technically added by CAA but PowerHA is codependent.
<code>clinfo_deadman</code>	<code>6176/tcp</code>
<code>clinfo_client</code>	<code>6174/tcp</code>
<code>clsmuxpd</code>	<code>6270/tcp</code>
<code>clm_lkm</code>	<code>6150/tcp</code>
<code>clm_smux</code>	<code>6175/tcp</code>
<code>clcomd_caa</code>	<code>16191/tcp</code>
<code>emsvcs</code>	<code>6180/udp</code>
<code>http-alt</code>	<code>8080/tcp/dup</code> - Used by SystemMirror User Interface (SMUI) server
<code>http-alt</code>	<code>8081/tcp/dup</code> - Used by SMUI agent/client nodes
<code>cthags</code>	<code>12348/udp</code>

Note: If you install PowerHA Enterprise Edition for Geographic Logical Volume Manager (GLVM), the following entry for the port number and connection protocol is automatically added to the `/etc/services` file in each node on the local and remote sites on which you installed the software:

`rpv 6192/tcp`

In addition to PowerHA 7.2.7, Resource Monitoring and Control (RMC) uses the following ports:

rmc	657/tcp
rmc	657/udp

/etc/snmpd.conf

The default version of the file for AIX 5.1 and later is snmpdv3.conf.

The SNMP daemon reads the /etc/snmpd.conf configuration file when it starts and when a refresh or kill -1 signal is issued. This file specifies the community names and associated access privileges and views, hosts for trap notification, logging attributes, snmpd-specific parameter configurations, and SNMP multiplexing (SMUX) configurations for the snmpd. The PowerHA installation process adds a **clsmuxpd** password to this file.

The following entry is added to the end of the file to include the PowerHA Management Information Base (MIB), which is supervised by the Cluster Manager:

```
smux      1.3.6.1.4.1.2.3.1.2.1.5      clsmuxpd_password # PowerHA SystemMirror clsmuxpd
```

/etc/snmpd.peers

The /etc/snmpd.peers file configures snmpd SMUX peers. During installation, PowerHA adds the following entry to include the **clsmuxpd** password to this file:

```
clsmuxpd 1.3.6.1.4.1.2.3.1.2.1.5 "clsmuxpd_password" # PowerHA SystemMirror clsmuxpd
```

/etc/syslog.conf

The /etc/syslog.conf configuration file controls the output of the syslogd daemon, which logs system messages. During installation, PowerHA adds entries to this file that direct the output from problems that are related to PowerHA to certain files.

CAA also adds a line as shown at the beginning of Example 3-2.

Example 3-2 /etc/syslog.conf file entries

```
caa.debug /var/adm/ras/syslog.caa rotate size 10m files 10 compress  
local0.crit /dev/console  
local0.info;user.notice;daemon.notice /var/hacmp/adm/cluster.log rotate size 1m  
files 8
```

The /etc/syslog.conf file must be identical on all cluster nodes.

/etc/trcfmt

The /etc/trcfmt file is the template file for the system trace logging and report utility (**trcrpt**). The installation process adds PowerHA tracing to the trace format file. PowerHA tracing is performed for the **clstrmgrES** and **cldinfo** daemons.

/var/spool/cron/crontab/root

The PowerHA installation process adds PowerHA log file rotation to the /var/spool/cron/crontabs/root file, as shown in Example 3-3.

Example 3-3 /var/spool/cron/crontabs/root file additions

```
0 0 * * * /usr/es/sbin/cluster/utilities/clcycle 1>/dev/null 2>/dev/null # >  
PowerHA SystemMirror Logfile rotation
```

3.4.6 Application software

Typically, applications are not dependent on PowerHA versions because they are not aware of the underlying PowerHA functions, that is, PowerHA only starts and stops them. PowerHA can also monitor applications, but by using an application-dependent method.

Check with the application vendor to ensure that no issues, such as licensing, exist with the usage of PowerHA.

3.4.7 Licensing

The two aspects of licensing are as follows:

- ▶ PowerHA 7.2.7 (features) licensing
- ▶ Application licensing

PowerHA 7.2.7

PowerHA licensing is core-based, which means that PowerHA must be licensed for each core that is used by the cluster nodes. The licensing is enforced by proper entitlement of the LPARs. Because they are core-based licenses for both PowerHA Standard and Enterprise Editions, the licenses depend on the IBM Power servers on which the cluster nodes run. The IBM Power servers can be divided into the following categories:

Small Tier	IBM Power S914 and IBM Power S1014, IBM Power S922 and IBM Power S1022, IBM Power S924 and IBM Power S1024, and IBM Power S950 and IBM Power S1050
Medium tier	IBM Power E980 and IBM Power E1080

Therefore, consider the following information:

- ▶ If you have an IBM Power server with four CPUs running in full system partition mode, you must have a license for four CPUs for that one server.
- ▶ If you have a Power server with four CPUs running LPARs and you run PowerHA 7.2.7 only in a two-CPU partition, you must have a license for two CPUs for that one LPAR.
- ▶ You must have a license for each server or LPAR on which you plan to run PowerHA 7.2.7.

In environments that are considered hot-standby, the total licensing is often $N+1$. N being the total number of CPUs in the production environment, and the +1 for the running standby node. So, in the first two bullets listed previously, the licensing would be five and three, respectively.

Now, if the cluster is mutual takeover and active-active configuration, then all CPUs in the cluster node LPARs must be licensed. Assuming that the LPARs are equally sized, then for the first two bullets above, the licensing would be eight and four, respectively.

Licensing is for entire cores (subcore licensing is not available). Always add up all the cores that are involved, and if there is a partial core that is left over, you must round up. For example, if your cluster ends up with 9.4 cores, you should license 10 cores.

If you have questions about licensing your environment, contact your IBM sales representative or IBM Business Partner for assistance.

Applications

Some applications have specific licensing requirements, such as a unique license for each processor that runs an application, which means that you must be sure that the application is licensed to allow it to run on more than one system. To license-protect an application, you might incorporate processor-specific information into the application when it is installed. As a result, even though the PowerHA 7.2.7 software processes a node failure correctly, it might be unable to restart the application on the failover node because of a restriction on the number of licenses for that application available within the cluster.

Important: To avoid this problem, be sure that you have a license for each system unit in the cluster that might potentially run an application. Check with your application vendor for any license issues for when you use PowerHA 7.2.7.

3.4.8 Completing the software planning worksheet

The worksheet in Table 3-5 lists all the software that is installed in our example.

Table 3-5 Cluster software

PowerHA 7.2.7 Cluster Worksheet - Part 3 Of 11 Cluster Software		Date: November 2022
Software Component	Version	Comments
AIX	7.3 TL0 SP2	Latest AIX version
RSCT	3.3.0.1	Latest RSCT version
PowerHA	7.2.7	GA level
Application	Test Application Version 1.	Add your application versions.
Comments	All software compatibility is verified. No issues with running applications with PowerHA 7.2.7. Application licensing is verified, and licenses are purchased for both servers.	

3.5 Operating system considerations

In addition to the AIX operating system levels and file sets, you must consider several other operating system aspects during the planning stage.

Disk space requirements

PowerHA 7.2.7 requires the following available space in the rootvg VG for installation:

- ▶ /usr requires 82 MB of available space for a full installation of PowerHA 7.2.7.
- ▶ / (root) requires 710 KB of available space.

A best practice is to allow approximately 100 MB available space in /var and /tmp for PowerHA 7.2.7 logs. This action depends on the number of nodes in the cluster, which dictates the size of the messages that are stored in the various PowerHA 7.2.7 logs.

Time synchronization

Time synchronization is important between cluster nodes for both application and PowerHA log issues. Time synchronization is a standard system administration practice, and as a best practice, use an NTP server or other procedure to keep the cluster nodes in sync.

Note: Maintaining time synchronization between the nodes is especially useful for auditing and debugging cluster problems.

Operating system settings

No additional operating system settings are required for PowerHA 7.2.7. Follow normal AIX tuning as required by the application workload. However, PowerHA checks many settings to see whether they are consistent across nodes and advise as needed.

3.6 Planning security

Protecting your cluster nodes (and applications) from unauthorized access is an important factor of the overall system availability. This section emphasizes certain general security considerations, and also PowerHA related aspects.

3.6.1 Cluster security

PowerHA needs a way to authenticate to all nodes in the cluster for running remote commands that are related to cluster verification, synchronization, and certain administrative operations (Cluster Single Point of Control (C-SPOC)). To prevent unauthorized access to cluster nodes, cluster security is required.

PowerHA inter-node communication relies on a cluster daemon (`c1cmd`), which eliminates the need for AIX “classic” remote commands.

For more information, see Chapter 8, “Cluster security” on page 347.

3.6.2 User administration

Most applications require user information to be consistent across the cluster nodes (user ID, group membership, and group ID) so that users can log in to surviving nodes without experiencing problems, which are particularly important in a failover (takeover) situation. Application users must be able to access the shared files from any required node in the cluster, which usually means that the application-related user and group identifiers (UID and GID) must be the same on all nodes. In preparation for a cluster configuration, be sure to consider and correct inconsistent user information, or you might experience service problems during a failover.

After PowerHA is installed, it contains facilities that you use to manage AIX user and group accounts across the cluster. It also provides a utility to authorize specified users to change their own password across nodes in the cluster.

Attention: If you manage user accounts with a utility such as Network Information Service (NIS), PSSP user management, or Distributed Computing Environment (DCE) Manager, do *not* use PowerHA user management. Using PowerHA user management in this environment might cause serious system inconsistencies in the user authentication databases.

For more information about user administration, see 7.3.1, “C-SPOC user and group administration” on page 261.

3.6.3 High Availability Cluster Multiprocessing group

During the installation of PowerHA, the hacmp group is created if it does not exist. During creation, PowerHA picks the next available GID for the hacmp group.

Before installation: If you prefer to control the GID of the hacmp group, create the hacmp group before installing the PowerHA file sets.

For more information about user administration, see 7.3.1, “C-SPOC user and group administration” on page 261.

In addition to the ports identified in the /etc/services file, the following services also require ports. However, these ports are selected randomly when the processes start. Currently, there is no way to indicate specific ports, so be aware of their presence. Typical ports are shown for illustration, but these ports can be altered if you need to do so:

- ▶ #clstrmgr 870/udp
- ▶ #clstrmgr 871/udp
- ▶ #linfo 32790/udp

3.6.4 Planning for PoweHA file collections

PowerHA requires certain files to be identical on all cluster nodes. These files include event scripts, application scripts, certain AIX configuration files, and PowerHA configuration files. With the PoweHA File Collections facility, you can auto-synchronize these files among cluster nodes to warn you of any unexpected results (for example, if a file in a collection was deleted or has a length of zero on one or more cluster nodes).

These file collections can be managed through SMIT menus. You can add, delete, and modify file collections to meet your needs.

Default PowerHA 7.2.7 file collections

When you install PowerHA 7.2.7, it sets up the following default file collections:

- ▶ Configuration_Files
- ▶ HACMP_Files

Configuration_Files

The Configuration_Files collection is a container for the following essential system files:

- ▶ /etc/hosts
- ▶ /etc/services
- ▶ /etc/snmpd.conf
- ▶ /etc/snmpdv3.conf
- ▶ /etc/rc.net
- ▶ /etc/inetd.conf
- ▶ /usr/es/sbin/cluster/netmon.cf
- ▶ /usr/es/sbin/cluster/etc/clhosts
- ▶ /usr/es/sbin/cluster/etc/rhosts
- ▶ /usr/es/sbin/cluster/etc/clinfo.rc

You can alter the propagation options for this file collection. You can also add and remove files to and from this file collection.

HACMP_Files

The HACMP_Files collection is a container that typically holds user-configurable files of the PowerHA configuration such as application start and stop scripts, customized events, and others. This file collection cannot be removed or modified, and you cannot add files to or delete files from it.

Example: For example, when you define an application server to PowerHA (start, stop, and optionally, monitoring scripts), PowerHA automatically includes these files into the HACMP_Files collection.

Unlike the Configuration_Files file collection, you *cannot* directly modify the files in this collection. For more information, see 7.2, “File collections” on page 254.

3.7 Planning cluster networks

Network configuration is a key component in the cluster design. In this section, we look at each network type and decide on the appropriate network connections and addresses.

In a typical clustering environment, clients access the applications through a Internet Protocol network (usually Ethernet) by using a service address. This service address is made highly available by PowerHA and moves between communication interfaces on the same network as required. PowerHA sends heartbeat packets between all communication interfaces (adapters) in the network to determine the status of the adapters and nodes and takes remedial actions as required.

To eliminate the TCP/IP protocol as a SPOF and prevent cluster partitioning, PowerHA also uses non-IP networks for heartbeating, which assists PowerHA with identifying the failure boundary, such as a TCP/IP failure or a node failure.

Figure 3-3 on page 95 provides an overview of the networks that are used in a cluster.

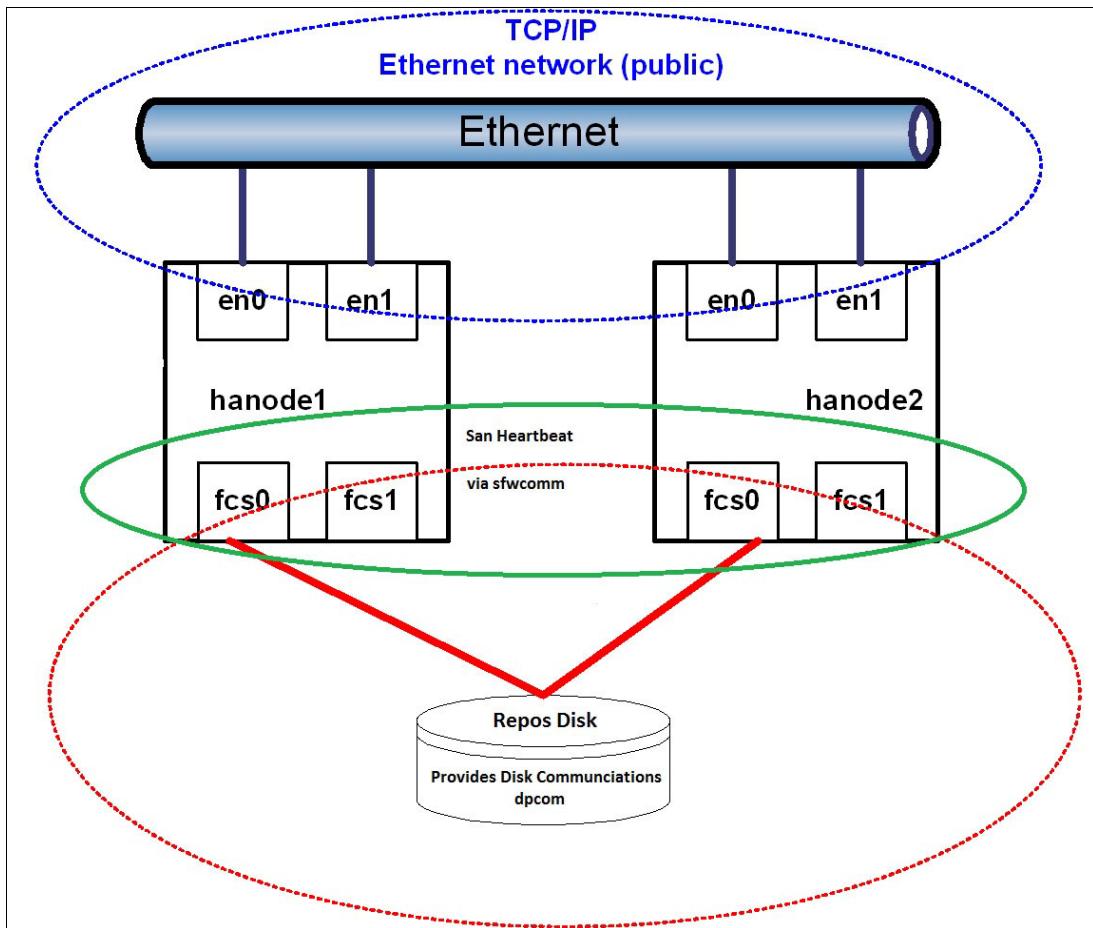


Figure 3-3 PowerHA cluster networks

An Ethernet network is used for public access and has multiple adapters that are connected from each node. This network holds the base IP addresses, the persistent IP addresses, and the service IP addresses. You can have more than one network; however, for simplicity, we use only one.

The cluster repository is also shown. It provides another path of communication across the disk. Multipath devices can be configured whenever there are multiple disk adapters in a node, multiple storage adapters, or both.

PowerHA, through CAA, also can use the SAN HBAs for communication, which is often referred to as SAN heartbeating, or *sancomm*. The device that enables it is *sfwcomm*.

All network connections are used by PowerHA to monitor the status of the network, adapters, and nodes in the cluster by default. In our example, we plan for an Ethernet and repository disk, but not a sancomm network.

3.7.1 Terminology

This section presents a quick summary of the terminology that is used in describing PowerHA 7.2.7 networking:

IP label	A name that is associated with an IP address and is resolvable by the system (/etc/hosts, BIND, and others).
Service IP label or address	An IP label or IP address over which a service is provided. Typically, it is the address that is used by clients to access an application. It can be bound to a node or shared by nodes, and it is kept highly available by PowerHA.
Persistent IP label or address	A node-bound IP alias that is managed by PowerHA 7.2.7 (the persistent alias never moves to another node).
Communication interface	A physical interface that supports the TCP/IP protocol (for example, an Ethernet adapter).
Network interface card (NIC)	A physical adapter that is used to provide access to a network (for example, an Ethernet adapter is referred to as a NIC).

3.7.2 General network considerations

Consider several factors when you design your network configuration. For more information, see Chapter 12, “Network considerations” on page 497.

Supported network types

PowerHA 7.2.7 allows inter-node communication with the following TCP/IP-based networks (Ethernet is the most common network in use):

- ▶ Ethernet (using physical, virtual, or vNIC)
- ▶ Etherchannel (802.3ad Link Aggregation)
- ▶ IP version 6 (IPv6)

The following TCP/IP-based networks are *not* supported:

- ▶ ATM
- ▶ Token Ring
- ▶ Fiber Distributed Data Interface (FDDI)
- ▶ InfiniBand
- ▶ The Virtual IP address (VIPA) facility of IBM AIX 5L
- ▶ Serial Optical Channel Converter (SOCC)
- ▶ Serial Line Internet Protocol (SLIP)
- ▶ Fibre Channel Switch (FCS)
- ▶ IBM High Performance Switch (HPS)
- ▶ 802_ether

Network connections

PowerHA 7.2.7 requires that each node in the cluster have at least one direct, non-routed network connection with every other node. These network connections pass heartbeat messages among the cluster nodes to determine the state of all cluster nodes, networks, and network interfaces.

PowerHA 7.2.7 also requires that all communication interfaces for a cluster network are defined on the same physical network, route packets, and receive responses from each other without interference by any network equipment.

Do not use intelligent switches, routers, or other network equipment that do not transparently pass UDP broadcasts and other packets between all cluster nodes.

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

Figure 3-4 illustrates a physical Ethernet configuration that shows dual-Ethernet adapters on each node that are connected across two switches but all configured in the same physical network (VLAN). This configuration is sometimes referred to as being in the same MAC *collision domain*.

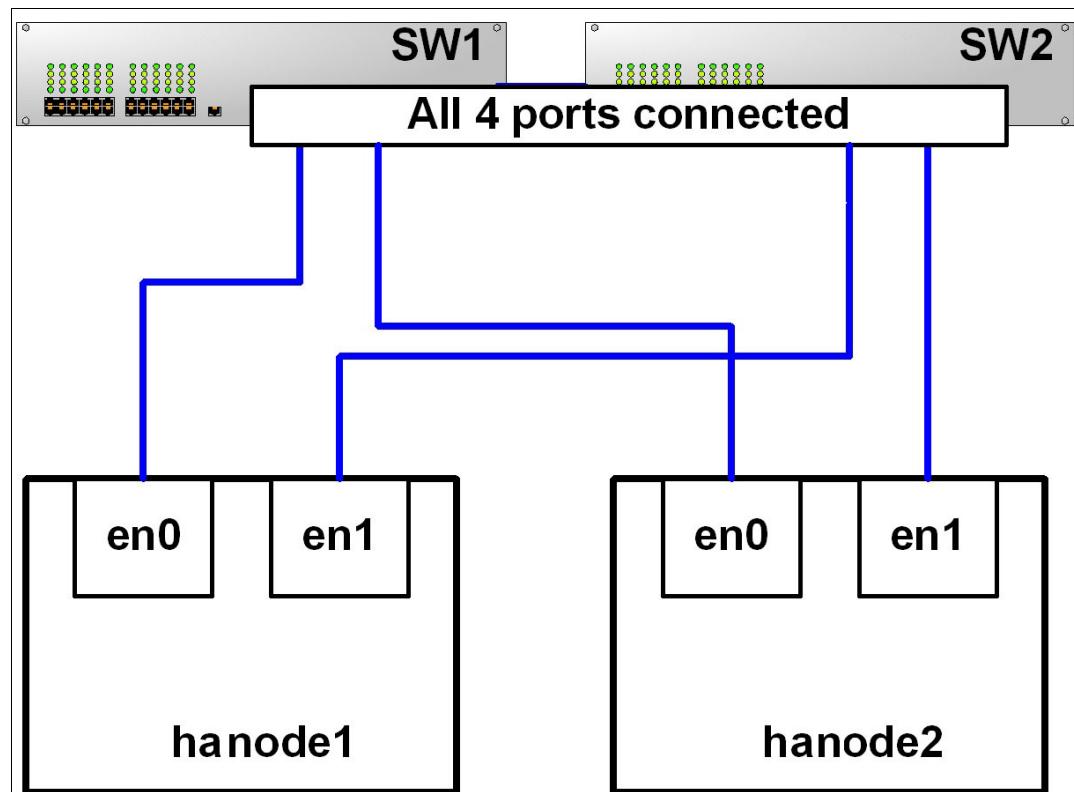


Figure 3-4 Ethernet switch connections

Etherchannel

PowerHA supports the usage of Etherchannel (or link aggregation) for the connection to an Ethernet network. Etherchannel can be useful if you want to use several Ethernet adapters for both extra network bandwidth and failover, but you also want to keep the PowerHA configuration simple. With Etherchannel, you can specify the Etherchannel interface as the communication interface. Any Ethernet failures, except for the Ethernet network itself, can be handled without PowerHA being aware or involved.

Shared Ethernet Adapters

Similar to Etherchannel, SEA allows multiple physical adapters across a VIOS to present themselves as one logical interface. This configuration also provides full redundancy if any individual adapter fails. However, the usage of vNIC (by using single root I/O virtualization (SR-IOV) capable adapters) is far more common today.

Hostnames and node names

Unlike earlier versions, PowerHA SystemMirror 7.1 had strict rules for which interface can be the hostname because of the CAA layer requirements.

Important:

- ▶ The hostname cannot be an alias in the /etc/hosts file.
- ▶ The name resolution for the hostname must work for both ways, so a limited set of characters can be used.
- ▶ The IP address that belongs to the hostname must be reachable on the server, even when PowerHA is down.
- ▶ The hostname cannot be a service address.
- ▶ The hostname cannot be an address on a network that is defined as private in PowerHA.
- ▶ The hostname, the CAA node name, and the “communication path to a node” must be the same.
- ▶ By default, the PowerHA, node name, the CAA nodename, and the “communication path to a node” are set to the same name.
- ▶ The hostname and the PowerHA nodename can differ.

The rules leave the base address and the persistent address as candidates for the hostname. You can use the persistent address as the host name only if you set up the persistent alias manually before you configure the cluster topology.

Starting with PowerHA 7.1.3, PowerHA (through CAA) now offers the ability to change the cluster node hostname dynamically as needed. For more information about this capability, see Chapter 11, “PowerHA cluster monitoring”, in *Guide to IBM PowerHA SystemMirror for AIX Version 7.1.3*, SG24-8167.

/etc/hosts

An IP address and its associated label (name) must be present in the /etc/hosts file. As a best practice, choose one of the cluster nodes to perform all changes to this file, and then use SCP or file collections to propagate the /etc/hosts file to the other nodes. However, in an inactive cluster, the auto-corrective actions during cluster verification can at least keep the IP addresses that are associated with the cluster in sync.

Note: Be sure that you test the direct and reverse name resolution on all nodes in the cluster and the associated Hardware Management Consoles (HMCs). All these nodes must resolve names identically, or you might run into security issues and other problems that are related to name resolution.

IP aliases

An IP alias is an IP address that is configured on a NIC in addition to the base IP address of the NIC. The usage of IP aliases is an AIX function that is supported by PowerHA 7.2.7. AIX supports multiple IP aliases on a NIC, each on the same or different subnets.

Note: While AIX allows IP aliases with *different subnet masks* to be configured for an interface, PowerHA 7.2.7 uses the subnet mask of the base IP address for all IP aliases that are configured on this network interface.

Persistent IP addresses (aliases)

A primary reason for using a persistent alias is to provide access to the node with PowerHA 7.2.7 services down. This alias is a routable address that is available while the node is up. Configure this alias through PowerHA. When PowerHA starts, it checks whether the alias is available. If it is not, PowerHA configures it on an available adapter on the designated network. If the alias is available, PowerHA leaves it alone.

Important: If the persistent IP address exists on the node, it *must* be an alias, *not* the base address of an adapter.

Consider the following information about a persistent alias:

- ▶ Always stays on the same node (node-bound).
- ▶ Coexists with other IP labels that are present on an interface.
- ▶ Does not require installing an extra physical interface on that node.
- ▶ Is not part of any resource group (RG).

Note: The persistent IP address is assigned by PowerHA to one communication interface, which is part of a PowerHA defined network.

Figure 3-5 illustrates the concept of the persistent address, which is simply another IP address that is configured on one of the base interfaces. The **netstat** command shows it as an extra IP address on an adapter.

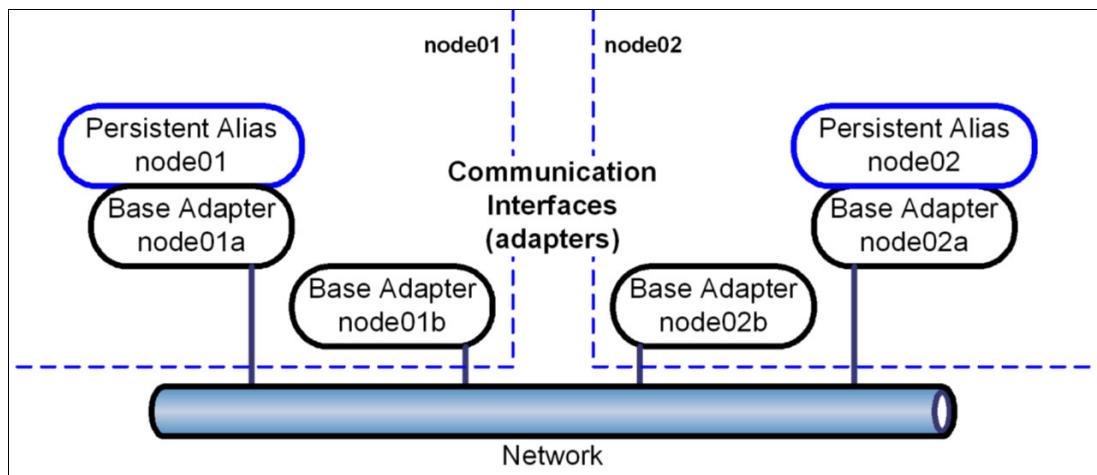


Figure 3-5 Persistent aliases

Subnetting

All the communication interfaces that are configured in the same PowerHA network must have the same subnet mask. Interfaces that belong to a different PowerHA network can have either the same or a different network mask.

For IPAT through aliases, consider this information:

- ▶ All base IP addresses on a node must be on separate subnets (if heartbeat monitoring over IP aliases is not used).
- ▶ All service IP addresses must be on a separate subnet from any of the base subnets.

Note: When using a single adapter per network configuration, the base (or boot) IP and the service IP can be on the same subnet. This approach is common today, and when it is used, it eliminates the need for a persistent IP alias.

- ▶ The service IP addresses can all be in the same or different subnets.
- ▶ The persistent IP address can be in the same or different subnet from the service IP address.

Default gateway (route) considerations

If you link your default route to one of the base address subnets, and that subnet is different from your service IP addresses and the adapter fails, your default route can be lost.

To prevent this situation, if you are *not* using a single adapter configuration with a boot IP on the routable subnet, then be sure to use a persistent address and link the default route to this subnet. The persistent address is active while the node is active, and so will the default route. If you choose not to use a persistent address, then you can create a custom event or post-event script to reestablish the default route if there is an issue.

This issue typically applies to a multiple interface or multiple boot adapters per node configuration, which overall are rare because it is common that the physical adapter redundancy is provided at a layer that is outside the OS, and PowerHA is unaware of it. Hence, most configurations appear to be a single adapter configuration but are still redundant.

PowerHA in a switched network

If VLANs are used, all interfaces that are defined to PowerHA on a network must be on the same VLAN, that is, all adapters in the same network are connected to the same physical network and can communicate between each other.

Note: Not all adapters must contain addresses that are routable outside the VLAN. Only the service and persistent addresses must be routable. The base adapter addresses and any aliases that are used for heartbeating do not need to be routed outside the VLAN because they are not known to the client side.

Ensure that the switch provides a timely response to ARP requests. For many brands of switches, this means turning *off* the following functions:

- ▶ The spanning tree algorithm
- ▶ **portfast**
- ▶ **uplinkfast**
- ▶ **backbonefast**

If you must have the spanning tree algorithm turned on, then the **portfast** function should also be turned on.

Ethernet media speed settings

Wherever possible, do *not* use autonegotiation because the media speed negotiation might cause problems in certain adapter-switch combinations. Instead, set the media to run at the values that you want for speed and duplex.

Multicast

Although multicast is rarely used in PowerHA, it still is a valid option. To use multicast, see 12.1, “Multicast considerations” on page 498.

IPv6 address planning

This section explains the IPv6 concepts and provides details for planning a PowerHA cluster by using IPv6 addresses. IPv6 support is available for PowerHA 7.1.2 and later versions.

IPv6 address format

IPv6 increases the IP address size from 32 bits to 128 bits, which supports more levels of addressing hierarchy, more addressable nodes, and simpler auto-configuration of addresses.

Figure 3-6 shows the basic format for global unicast IPv6 addresses.

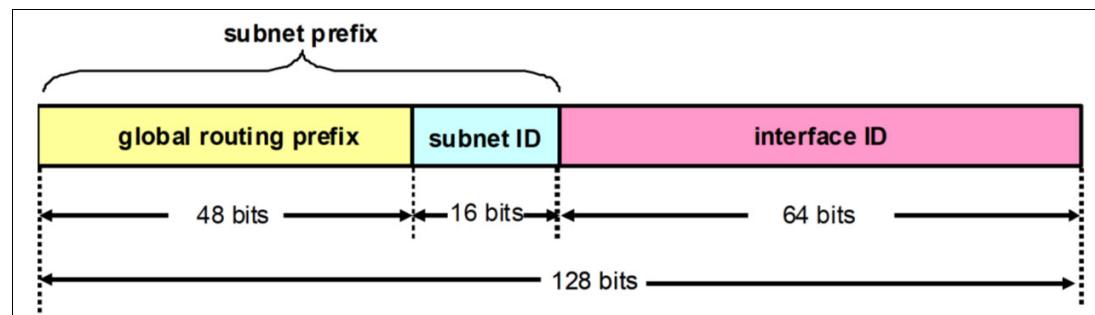


Figure 3-6 IPv6 address format

IPv6 addresses contain three parts:

- ▶ Global routing prefix
 - The first 48 bits (in general) are for a global prefix, and distributed for global routing.
- ▶ Subnet ID
 - The second 16 bits are freely configurable in a field that is available for definition within sites.
- ▶ Interface ID
 - The last 64 bits are for distributing for each network device.

Subnet prefix considerations

The subnet prefix, which corresponds to the subnet mask for IP version 4 (IPv4), is a combination of the *global routing prefix* and *subnet ID*. Although you may have longer subnet prefixes, in general 64 bits is a suitable length. IPv6 functions such as *Router Advertisement* are designed and assumed to use the 64-bit length subnet prefix. Also, the 16-bit subnet ID field allows 65,536 subnets, which are enough for general purposes.

IPv6 address considerations

The three basic IPv6 addresses are as follows:

- ▶ Link-local addresses

The IP addresses that are configured to communicate within the site (that cannot go outside the network router). This term also exists in IPv4. The IP range of the link-local address for IPv4 and IPv6 is as follows:

For IPv4: 169.254.0.0/16

For IPv6: fe08::/10

Although this address was optional in IPv4, it is required in IPv6. In AIX, IPv6 addresses are automatically generated based on the EUI-64 format, which uses the network card's MAC address. The logic of the link-local address creation is as follows:

For example, assume that you have a network card with a MAC address of 96:D8:A1:5D:5A:0C:

- a. Bit 7 is flipped and FFEE is added after bit 24. The result is as follows:

94:D8:A1:FF:EE:5D:5A:0C

- b. The subnet prefix fe08:: is added, which produces the following link-local address:

FE08::94D8:A1FF:EE5D:5A0C

In AIX, the **autoconf6** command is responsible for creating the link-local address.

- ▶ Global unicast addresses

The IP addresses that are configured to communicate outside of the router. The range 2000::/3 is provided for this purpose. The following addresses are predefined global unicast addresses:

Teredo address defined in RFC 4380 2001:0000::/32

Provided for document purposes defined in RFC 3849 2001:db8::/32

The 6 - 4 address defined in RFC 3056 2002::/16

- ▶ Loopback address

This term is the same one as for IPv4. It uses the following IP address:

::1/128

For PowerHA, you can have your boot IP addresses configured to the link-local address if it is suitable. However, for configurations involving sites, it is more suitable to configure boot IP addresses with global unicast addresses that can communicate with each other. The benefit is that you can have extra heartbeating paths, which help prevent cluster partitions.

The global unicast address can be configured *automatically* or *manually*:

- ▶ Automatically configured IPv6 global unicast address, either stateless or stateful:

- Stateless IP address

Global unicast addresses are provided through the Neighbor Discovery Protocol (NDP). Similar to link-local addresses, these IP addresses are generated based on the EUI-64 format. In comparison, the subnet prefix is provided by the network router. The client and the network router must be configured to communicate through the NDP for this address to be configured.

- Stateful IP address

Global unicast addresses are provided through an IPv6 DHCP server.

- ▶ Manually configured IPv6 global unicast address: The same term as for IPv4 static address.

As a best practice, use automatic IPv6 addresses for unmanaged devices such as client PCs and mobile devices, and use manual IPv6 addresses for managed devices, such as servers.

For PowerHA, you may have either automatic or manual IPv6 addresses. However, consider that automatic IP addresses are not guaranteed to persist. CAA restricts you to having the hostname labeled to a configured IP address, and you cannot change the IP addresses when the cluster services are active.

IPv4 and IPv6 dual-stack environment

When migrating to IPv6, usually you may keep your IPv4 networks. An environment that uses a mix of IP address families on the same network adapter is called a *dual-stack environment*.

With PowerHA, you may mix different IP address families on the same adapter (for example, an IPv6 service label in the network with IPv4 boot, or an IPv4 persistent label in the network with IPv6 boot). However, a best practice is to use the same family as the underlying network for simplifying planning and maintenance.

Figure 3-7 shows an example of this configuration.

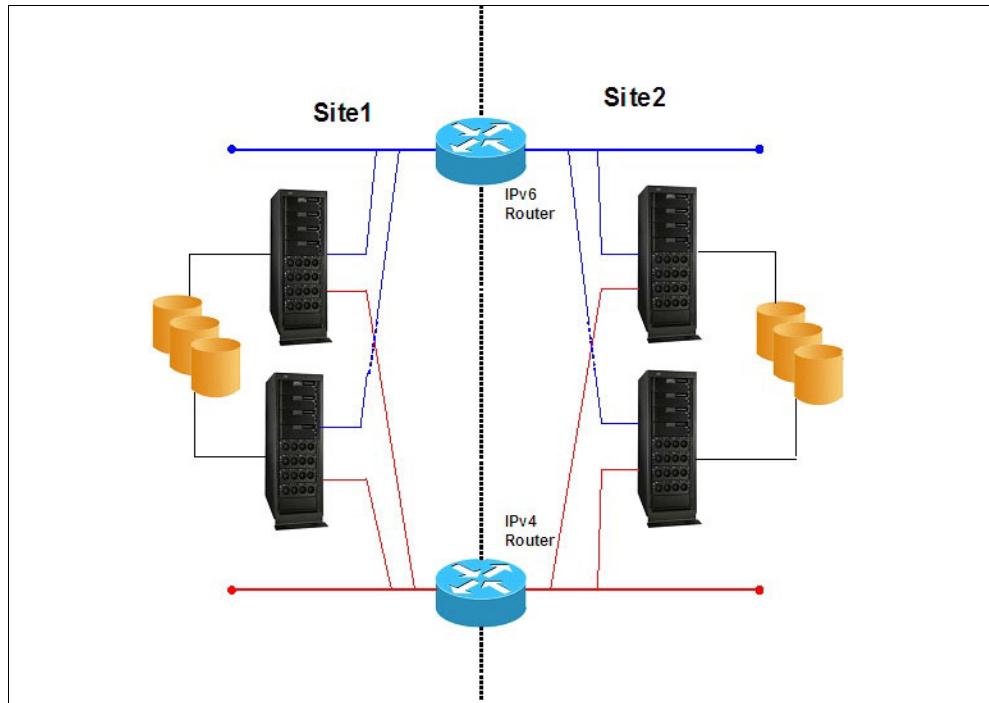


Figure 3-7 IPv6 dual stack environment

Multicast and IPv6

PowerHA SystemMirror 7.1.2 or later supports IPv6. However, you *cannot* explicitly specify the IPv6 multicast address. CAA uses an IPv6 multicast address that is derived from the IPv4 multicast address.

To determine the IPv6 multicast address, a standard prefix of 0xFF05 is combined by using the logical OR operator with the hexadecimal equivalent of the IPv4 address. For example, the IPv4 multicast address is 228.8.16.129 or 0xE4081081. The transformation by the logical OR operation with the standard prefix is 0xFF05:: | 0xE4081081. Thus, the resulting IPv6 multicast address is 0xFF05::E408:1081.

3.7.3 IP Address Takeover planning

IPAT is the mechanism that PowerHA uses to move service addresses between communication interfaces.

IPAT through aliasing is flexible. You can have multiple service addresses on the same adapter at any one time, and some time can be saved during failovers because PowerHA adds an alias rather than reconfigures the base IP address of an adapter.

With PowerHA, you can use IPAT through IP aliases with the following network types that support gratuitous ARP (in AIX):

- ▶ Ethernet
- ▶ XD_data
- ▶ XD_ip

By default, when PowerHA 7.2.7 starts, it automatically configures the service IP as an alias with the `firstalias` option. PowerHA 7.2.7 swaps the boot IP off the interface, puts the service IP in its place, and then aliases the boot IP on top of it.

Consider the following requirements to use IPAT through aliases:

- ▶ Subnet requirements:
 - Each base adapter must be on a separate subnet to allow heartbeating. The base addresses do not have to be routable outside of the cluster.
 - The service addresses are on a separate subnet from any of the base subnets when there are two or more interfaces per network configured. There can be multiple service addresses, and they can all be on the same subnet or different subnets.
 - The persistent alias can be in the same or different subnet as the service.
 - The subnet masks must all be the same.
- ▶ Multiple service labels can coexist as aliases on an interface.

In a multiple interface per network configuration, using a persistent alias and including it in the same subnet as your default route is common. The persistent address is included in the same subnet as the service addresses. The persistent alias can be used to access the node when PowerHA 7.2.7 is down and also overcome the default route issue.

You can configure a distribution preference for the placement of service IP labels that are configured in PowerHA 7.2.7. The placement of the alias is configurable through SMIT menus as follows:

- ▶ Anti-colocation

The default. PowerHA distributes the service IP labels across all boot IP interfaces in the same PowerHA network on the node. It also uses `firstalias` in its behavior for each interface.
- ▶ Colocation

PowerHA allocates all service IP addresses on the same boot IP interface.
- ▶ Colocation with persistent label

PowerHA allocates all service IP addresses on the boot IP interface that is hosting the persistent IP label, which can be useful in environments with a virtual private network (VPN) and firewall configuration, where only one interface is granted external connectivity. The persistent label is the source address.

- ▶ Colocation with source

Service labels are mapped by using the colocation preference. You can choose one service label as a source for outgoing communication. The service label that is chosen in the next field is the source address.

- ▶ Colocation with Persistent Label and Source

Service labels are mapped to the same physical interface that has the persistent IP label for this network. Use this option to choose one service label as the source for outgoing communication. The service label that is chosen in the next field is the source address.

- ▶ Anti-Colocation with source

Service labels are mapped by using the Anti-Colocation preference. If not enough adapters are available, more than one service label can be placed on one adapter. This choice allows one label to be selected as the source address for outgoing communication.

- ▶ Anti-Colocation with persistent label

PowerHA distributes all service IP labels across all boot IP interfaces (in the same logical network) that are *not* hosting the persistent IP label. If no other interfaces are available, the service IP labels share the adapter with the persistent IP label.

- ▶ Anti-Colocation with persistent label and source

Service labels are mapped by using the Anti-Colocation with Persistent preference. One service address can be chosen as a source address for a case where there are more service addresses than boot adapters.

- ▶ Disable Firstalias

PowerHA v7 automatically configures the service IP as an alias with the **firstalias** option regardless of the user's setting. However, in certain scenarios, such as Network Installation Management (NIM) operations, the default **firstalias** feature can cause errors. This option allows the user to disable **firstalias**, and thus retain the historic default original mode.

PowerHA allocates all service IP addresses on the boot IP interface that is hosting the persistent IP label, which can be useful in environments with a VPN and firewall configuration, where only one interface is granted external connectivity.

For more information and examples of using service distribution policies, see 12.2, “Distribution preference for service IP aliases” on page 501.

3.7.4 Additional network planning considerations

In addition to configuring the network topology, certain considerations apply when you use PowerHA 7.2.7 with a domain name server (DNS) and NIS. You must understand how to change the heartbeat rates and the importance of the `netmon.cf` file.

PowerHA with DNS and NIS

To ensure that cluster events complete successfully and quickly, PowerHA disables the NIS or DNS hostname resolution during service IP label swapping by setting the **NSORDER AIX** environment variable to `local`. Therefore, the `/etc/hosts` file of each cluster node must contain all the PowerHA defined IP labels for all cluster nodes.

After the swap completes, DNS access is restored.

As a best practice, place the following line into the /etc/netsvc.conf file to ensure that the /etc/hosts file is read before a DNS lookup is attempted:

```
hosts = local, bind4
```

Network failure detection tunables

PowerHA, mostly by using CAA functions, can modify the time that is involved to detect a network, a node, and others by using cluster tunables. For more information about these options, see 12.3.1, “Changing cluster-wide tunables” on page 506.

usr/es/sbin/cluster/netmon.cf

If a virtualized network environment, such as provided by VIOS, is used for one or more interfaces, PowerHA can have difficulty accurately determining a particular adapter failure. For these situations, use the netmon.cf file. For more information, see 12.5, “Understanding the netmon.cf file” on page 517.

3.7.5 Completing the network planning worksheets

The following worksheets (4 - 6) include the necessary network information.

The first worksheet (Table 3-6) shows the specifications for the Ethernet network that is used in our example.

Table 3-6 Cluster Ethernet networks

PowerHA 7.2.7 Cluster Worksheet - Part 4 Of 11 Cluster Ethernet Networks				Date: November 2022
Network Name	Network Type	Netmask	Node Names	
ether10	Ethernet (public)	255.255. 255.0	Node01, Node02	
Comments				

Table 3-7 documents the cluster repository disk in the cluster.

Table 3-7 Point-to-point networks

PowerHA 7.2.7 Cluster Worksheet - Part 5 Of 11 Cluster Repository Disk		Date: November 2022
Node Names	Devices	
Node01 Node02	hdisk2 hdisk2	
Comments		

After the networks are recorded, document the interfaces and IP addresses that are used by PowerHA, as shown in Table 3-8 on page 107.

Table 3-8 Cluster communication interfaces and IP addresses

PowerHA Cluster Worksheet - Part 6 Of 11 Interfaces And IP addresses					Date: November 2022
Node01					
IP Label	IP Alias Dist. Preference	Network Interface	Network Name	Interface Function	IP address /mask
Node01a	NA	en0	ether10	Base (non-service)	10.10.31.31 255.255.255.0
hanode1	Anti- Colocation (default)	NA	ether10	Persistent	192.168.100.31 255.255.255.0
app1svc	Anti- Colocation (default)	NA	ether10	Service	192.168.100.131 255.255.255.0
Node02					
IP Label	IP Alias Dist. Preference	Network Interface	Network Name	Interface Function	IP address /mask
Node02a	NA	en0	ether10	Base (non-service)	10.10.31.32 255.255.255.0
hanode2	Anti- Colocation (default)	NA	ether10	Persistent	192.168.100.32 255.255.255.0
app2svc	Anti- Colocation (default)	NA	ether10	Service	192.168.100.132 255.255.255.0
Comments	Each node contains 2 base adapters, each in their own subnet. Each node also contains a persistent (node bound) address and a service address. IPAT through aliases is used				

3.8 Planning storage requirements

When planning cluster storage, consider the following requirements:

- ▶ Physical disks:
 - Ensure that your disk solution is highly available by using mirroring, RAID, and redundant hardware.
 - Internal disks. These disks can be used for rootvg or any non-shared data. However, it is not common to use any internal disks except for VIOS.
 - External disks. These disks must be the location of all shared and application data.
- ▶ Logical Volume Manager (LVM) components:
 - All shared storage has a unique logical volume (LV), jfslog, and file system names.
 - VG major numbers are unique. Though only required for NFS, it is a best practice to have them match across nodes.
 - Determine whether mirroring of data is required.

3.8.1 Internal disks

Internal node disks typically contain rootvg and perhaps the application binary files. As a best practice, mirror the internal disks for higher availability to prevent a node failover because of a simple internal disk failure.

3.8.2 Cluster repository disk

PowerHA SystemMirror uses a shared disk to store CAA cluster configuration information. You must have at least 512 MB and no more than 460 GB of disk space that is allocated for the cluster repository disk. This feature requires that a dedicated shared disk be available to all nodes that are part of the cluster. This disk cannot be used for application storage or any other purpose. In all cases, you may assign more backup repository disks.

When planning for a repository disk for a multi-site cluster solution, you should understand the following cluster types:

- ▶ Stretched cluster
 - Requires and shares only one repository disk. When implementing the cluster configuration with multiple storages at different sites, consider allocating the CAA repository and the backup repositories in different storage across the sites to increase the availability of the repository disk if there is a storage failure. As an example, when using a cross-site LVM mirroring configuration with a storage subsystem at each site, you can allocate the primary disk repository at site 1 and the backup repository on the storage at site 2.
- ▶ Linked clusters
 - Requires a repository disk to be allocated to each site. If there is no other storage at a site, plan to allocate the backup repository disk on a different set of disks (other arrays) within the same storage to increase the repository disk availability if there are disk failures.

Considerations for a stretched cluster

When you implement a stretched cluster, these considerations apply:

- ▶ There is only *one* repository disk in a stretched cluster.
- ▶ Repository disks *cannot* be mirrored by using AIX LVM. Therefore, as a best practice, have it RAID-protected by a redundant and highly available storage configuration.
- ▶ All nodes must have access to the repository disk.
- ▶ If the repository disk *fails* or becomes *inaccessible* by one or more nodes, the nodes stay online and the cluster can still process events such as node, network, or adapter failures. On failure, the cluster ahaFS event REP_DOWN occurs. However, no cluster configuration changes can be performed in this state. Any attempt to do so is stopped with an error message.
- ▶ A backup repository disk can be defined if there is a failure. When planning the disks that you want to use as repository disks, you must plan for a backup or replacement disk that can be used if the primary repository disk fails. The backup disk must be the same size and type as the primary disk, but can be in a different physical storage. Update your administrative procedures and documentation with the backup disk information. You can also replace a working repository disk with a new one to increase the size or to change to a different storage subsystem. To replace a repository disk, you can use the SMIT interface or `c1mgr` command. The cluster ahaFS event REP_UP occurs on replacement.

Additional considerations for linked clusters

When you implement linked clusters, these extra considerations apply:

- ▶ The nodes *within a site* share a common repository disk with all the characteristics that were specified previously.
- ▶ The repositories between sites are kept in sync internally by CAA.

3.8.3 SAN-based heartbeat

PowerHA supports SAN-based heartbeat only within a site. The SAN heartbeat infrastructure can be accomplished in several ways:

- ▶ Using real adapters on the cluster nodes and enabling the storage framework capability (`sfwcomm device`) of the HBAs. At the time of writing, Fibre Channel (FC) and serial-attached SCSI (SAS) technologies are supported. For more information about supported HBAs and the steps to set up the storage framework and communication, see [Setting Cluster SAN Communication](#).
- ▶ In a virtual environment that uses NPIV or vSCSI with a VIOS, enabling the `sfwcomm` interface requires activating the target mode (the `tme` attribute) on the real adapter in the VIOS and defining a private VLAN (ID 3358) for communication between the partition containing the `sfwcomm` interface and the VIOS. The real adapter on the VIOS must be a supported HBA, as indicated in [Setting Cluster SAN Communication](#).

Note: For a demonstration about creating SAN heartbeating in a virtualized environment, see this [YouTube](#) video.

3.8.4 Shared disks

Application data is on the external disk so that it can be accessed by all required nodes. These disks are referred to as the *shared disks*.

Varied on: All shared disks must be “zoned” to any cluster nodes requiring access to the specific volumes, that is, the shared disks must be able to be varied on and accessed by any node that must run a specific application.

Be sure to verify that shared VGs can be manually varied on each node.

In a PowerHA cluster, shared disks are connected to more than one cluster node. In a non-concurrent configuration, only one node at a time owns the disks. If the owner node fails to restore service to clients, another cluster node in the RG node list acquires ownership of the shared disks and restarts applications.

When working with a shared VG, be sure to *not* perform any of the following actions:

- ▶ Do *not* use any internal disk in a shared VG because it will not be accessible by other nodes.
- ▶ Do *not* auto-varyon the shared VGs in a PowerHA cluster at system start. Ensure that the automatic varyon attribute in the AIX Object Data Manager (ODM) is set to No for shared VGs that are part of RGs. You can use the cluster verification utility to auto-change this attribute.

Important: If you define a VG to PowerHA, do not manage it manually on any node outside of PowerHA while PowerHA is running. This approach can lead to unpredictable results. Always use C-SPOC to maintain the shared VGs.

3.8.5 Enhanced concurrent mode volume groups

Any disk that is supported by PowerHA for attachment to multiple nodes can be used to create an ECM VG and used in either concurrent or non-concurrent environments:

Concurrent An application runs on all active cluster nodes concurrently. To allow such applications to access their data, concurrent VGs are varied on all active cluster nodes. Then, the application must ensure consistent data access.

Non-concurrent An application runs on one node at a time. The VGs are not concurrently accessed; they are still accessed by only one node at any time.

PowerHA requires that all shared VGs are enhanced concurrently regardless of how they are used. For typical non-concurrent configurations, enhanced concurrent enables the fast disk takeover feature. When the VG is activated in ECM, the LVM allows access to the VG on all nodes. However, LVM restricts the higher-level connections, such as Journaled File System (JFS) mounts and NFS mounts, on all nodes, and allows them only on the node that owns the VG.

Note: Although you must define ECM VGs, it *does not* necessarily mean that you use them for concurrent access. For example, you can still define and use these VGs as normal shared file system access. However, you *cannot* define file systems on VGs that are intended for concurrent access.

3.8.6 How fast disk takeover works

Fast disk takeover reduces the total failover time by providing faster acquisition of the disks without having to break SCSI reserves. It uses enhanced concurrent VGs (ECVGs), and more LVM enhancements that are provided by AIX.

ECVG can be activated in two modes:

- ▶ Active mode:
 - Operations on file systems, such as file system mounts
 - Operations on applications
 - Operations on LVs, such as creating LVs
 - Synchronizing VGs
- ▶ Passive mode:
 - LVM read-only access to the VG's special file
 - LVM read-only access to the first 4 KB of all LVs that are owned by the VG

The following operations are not allowed when a VG is varied on in the passive state:

- ▶ Operations on file systems, such as mount
- ▶ Any open or write operation on LVs
- ▶ Synchronizing VGs

Active mode is similar to a non-concurrent VG that is varied online with the **varyonvg** command. It provides full read/write access to all LVs and file systems, and it supports all LVM operations.

Passive mode is the LVM equivalent of disk fencing. Passive mode allows readability only of the Volume Group Descriptor Area (VGDA) and the first 4 KB of each LV. It does *not* allow read/write access to file systems or LVs. It also does not support LVM operations.

When an RG that contains an ECVG is brought online, the VG is first varied on in passive mode, and then it is varied on in active mode. The active mode state applies only to the current RG owner node. When any other RG member node joins the cluster, the VG is varied on in passive mode.

When the owning or home node fails, the failover node changes the VG state from passive mode to active mode through the LVM. This change takes approximately 10 seconds and is at the VG level. It can take longer with multiple VGs with multiple disks per VG. However, the time impact is minimal compared to the previous method of breaking SCSI reserves.

The active and passive mode flags for the **varyonvg** command are not documented because they should *not* be used outside a PowerHA environment. However, you can find them in the **hacmp.out** log.

- ▶ Active mode varyon command:
`varyonvg -n -c -A -0 app2vg`
- ▶ Passive mode varyon command:
`varyonvg -n -c -P app2vg`

Important: Also, do *not* run these commands unless directed to do so from IBM Support. Also, the cluster services must be running.

To determine whether the VG is online in active or passive mode, verify the VG PERMISSION field from the **lsvg** command output, as shown in Figure 3-8.

```
Maddi / > lspv
hdisk5      0022be2a8607249f      app2vg
hdisk6      0022be2a86607918      app1vg      concurrent
hdisk7      0022be2a8662ce0e      app2vg
hdisk8      0022be2a86630978      app1vg      concurrent

Maddi / > lsvg -o
rootvg

Maddi / > lsvg app1vg
VOLUME GROUP: app1vg          VG IDENTIFIER: 0022be2a00004c48
VG STATE: active             PP SIZE: 16 megabyte(s)
VG PERMISSION: passive-only
MAX LVs: 256                 TOTAL PPs: 1190
LVs: 0                       FREE PPs: 1180
OPEN LVs: 0                   USED PPs: 10
TOTAL PVs: 2                  QUORUM: 2
STALE PVs: 0                  VG DESCRIPTORS: 3
ACTIVE PVs: 2                 STALE PPs: 0
Concurrent: Enhanced-Capable AUTO ON: no
VG Mode: Concurrent          Auto-Concurrent: Disabled
Node ID: 6                     Active Nodes:
MAX PPs per PV: 1016          MAX PVs: 32
LTG size: 128 kilobyte(s)     AUTO SYNC: no
HOT SPARE: no                  BB POLICY: relocatable
```

Figure 3-8 Passive mode volume group status

There are other distinguishing LVM status features that you notice for VGs that are being used in a fast disk takeover configuration. For example, the VG shows online in concurrent mode on each active cluster member node by using the **lspv** command. However, the **lsvg -o** command reports only the VG online to the node that has it varied on in active mode. An example of how passive mode status is reported is shown in Figure 3-8.

3.8.7 Enabling fast disk takeover

No actual option or flag within the PowerHA cluster configuration is specifically related to fast disk takeover. It is used automatically when the shared VGs are ECVGs. These VGs are added as resources to a non-concurrent mode style RG. The combination of these two is how PowerHA determines to use the fast disk takeover method of VG acquisition.

When a non-concurrent style RG is brought online, PowerHA checks one of the VG member disks to determine whether it is an ECVG. PowerHA determines this configuration by using the **lqueryvg -p devicename -X** command. A return output of 0 (zero) indicates a regular non-concurrent VG. A return output of 32 indicates an ECVG.

In Figure 3-9 on page 113, hdisk0 is a rootvg member disk that is non-concurrent. The hdisk6 instance is an ECVG member disk.

```
Maddi / > lqueryvg -p hdisk0 -X  
0  
Maddi / >lqueryvg -p hdisk6 -X  
32
```

Figure 3-9 Example of how PowerHA determines the volume group type

3.8.8 Shared logical volumes

Planning for shared LVs is all about data availability. Making your data highly available by using mirroring or RAID is a key requirement. PowerHA relies on LVM and storage mechanisms (RAID) to protect against disk failures, therefore it is imperative that you make the disk infrastructure highly available.

Consider the following guidelines when planning shared LVM components:

- ▶ LV copies or RAID arrays can protect against loss of data from physical disk failure.
- ▶ All operating system files should be in the root VG (rootvg) and all user data should reside on a separate shared VG.
- ▶ VGs that contain at least three physical volumes (PVs) provide the maximum availability when implementing mirroring.
- ▶ If you plan to specify the “Use Forced Varyon of Volume Groups if Necessary” attribute for the VGs, you must use the super-strict disk allocation policy for mirrored LVs.
- ▶ With LVM mirroring, each PV containing a copy gets its power from a separate source. If one power source fails, separate power sources maintain the “no SPOF” objective.
- ▶ Consider quorum issues when laying out a VG. With quorum enabled, a two-disk VG has the risk of losing quorum and data access. Either build three-disk VGs (for example, by using a quorum buster disk or logical unit number (LUN)) or disable quorum.
- ▶ Consider the cluster configurations that you designed. A node whose resources are not taken over should not own critical VGs.
- ▶ Ensure that regular backups are scheduled.

After you establish a highly available disk infrastructure, also consider the following items when designing your shared VGs:

- ▶ All shared VGs have unique LV and file system names, which include the jfs and jfs2 log files.
PowerHA 7.2.7 also supports Journaled File System 2 (JFS2) with INLINE logs, which is a best practice instead of dedicated log devices.
- ▶ Major numbers for each VG are unique within a node. Though optional, unless using NFS, it is a best practice to make the major numbers for each VG match on all nodes.
- ▶ JFS2 Encrypted File Systems (EFSs) are supported. For more information about using EFS with PowerHA, see 8.5, “Federated security for cluster-wide security management” on page 358.

Figure 3-10 shows the basic components in the external storage. All LVs and file system names are unique, as is the major number for each VG. The data is made highly available by using SAN disk and redundant paths to the devices.

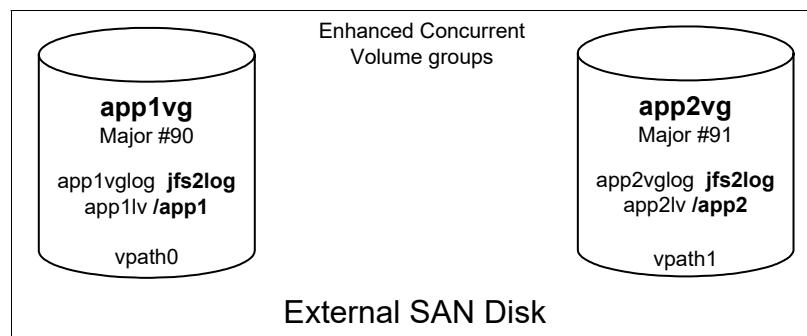


Figure 3-10 External disk

3.8.9 Completing the storage planning worksheets

The following worksheets (7 - 8) contain the required information about the shared VGs. Combined, they give you a good idea of the shared disk configuration.

Document the shared VGs and physical disks, as shown in Table 3-9.

Table 3-9 Shared disks

PowerHA 7.2.7 Cluster Worksheet - Part 7 Of 11 Shared Disks				Date: November 2022
Node01		Node02		
VG name	HDisk	HDisk	VG name	
app1vg	hdisk2, hdisk3	hdisk2, hdisk3		
	hdisk4, hdisk5	hdisk4, hdisk5	app2vg	
Comments	All disks are seen by both nodes. app1vg normally is on Node01, and app2vg normally is on Node02.			

Record the shared VG details, as shown in Table 3-10.

Table 3-10 Shared volume groups

PowerHA 7.2.7 Cluster Worksheet - Part 8 Of 11 Shared Volume Groups (Non-concurrent)		Date: November 2022
Resource Group	Volume Group 1	Volume Group 1
AESPArg	app1vg Major Number = 90 log = app1vglog LV 1 = app1lv1 File system 1 = /app1 (20 GB)	NA
NMIXXrg	app2vg Major Number = 91 log = app2vglog LV 1 = app2lv1 File system 1 = /app2 (20 GB)	NA
Comments	Create the shared VG by using C-SPOC after ensuring that the physical volume identifiers (PVIDs) exist on each node.	

3.9 Application planning

Most applications that run on a stand-alone AIX server can be integrated into a PowerHA 7.2.7 cluster because they are not aware of the underlining PowerHA functions. One key requirement for an application to be suitable for a PowerHA cluster is that it must be able to restart or recover from a failure without manual intervention because PowerHA simply runs a script that starts and stops an application.

When planning for an application to be highly available, be sure that you understand the resources that are required by the application and the location of these resources in the cluster. This approach helps you provide a solution for PowerHA to handle the application correctly if a node fails.

You must thoroughly understand how the application behaves in single-node and multi-node environments. Be sure that as part of preparing the application for PowerHA, you test the execution of the application manually on both nodes before turning it over to PowerHA to manage. Do not make assumptions about the application's behavior under failover conditions.

Note: The key prerequisite to making an application highly available is that it must first run correctly in stand-alone mode on each node on which it can reside.

Be sure that the application runs on all required nodes properly before configuring it to be managed by PowerHA.

Analyze and address the following aspects:

- ▶ Application code: Binary files, scripts, links, configuration files, and others.
- ▶ Environment variables: Any environment variable that must be passed to the application for proper execution.
- ▶ Application data.
- ▶ Networking setup: IP addresses and hostname.
- ▶ Application licensing.
- ▶ Application-defined system users.

When you plan for an application to be protected in a PowerHA 7.2.7 cluster, consider the following actions:

- ▶ Ensure that the application is compatible with the version of AIX that you use.
- ▶ Ensure that the application is compatible with the shared storage solution because it is where the application data is.
- ▶ Have adequate system resources (CPU and memory), especially when the same node is hosting all the applications that are part of the cluster.
- ▶ Ensure that the application runs successfully in a single-node environment. Debugging an application in a cluster is more difficult than debugging it on a single server.
- ▶ Lay out the application and its data so that only the data is on shared external disks. This arrangement prevents software license violations and simplifies failure recovery.
- ▶ If you plan to include mult-tiered applications in parent-child-dependent RGs in your cluster, such as a database and application server, PowerHA provides a SMIT menu where you can specify this relationship.
- ▶ Write robust scripts to both start and stop the application on the cluster nodes. The startup script must be able to recover the application from an abnormal termination. Ensure that they run properly in a single-node environment before including them in PowerHA.
- ▶ Confirm the application licensing requirements. Some vendors require a unique license for each processor that runs an application, which means that you must license-protect the application by incorporating processor-specific information into the application when it is installed.

As a result, even though the PowerHA software processes a node failure correctly, it might be unable to restart the application on the failover node because of a restriction on the number of licenses for that application that are available within the cluster. To avoid this problem, be sure that you have a license for each system unit in the cluster that might potentially run an application.

- ▶ Verify that the application uses a proprietary locking mechanism if you need concurrent access.

Tip: When you plan the application, if the application requires any manual intervention, it is not suitable for a PowerHA cluster.

3.9.1 Application controllers

In PowerHA, an application controller is a set of scripts that are used to start and stop an application.

Configure your application controller by creating a name to be used by PowerHA and associating a start script and a stop script.

After you create an application controller, associate it with an RG. PowerHA uses this information to control the application. For more information, see 2.4.4, “Application controller scripts” on page 51.

3.9.2 Application monitoring

PowerHA 7.2.7 can monitor your application by using either of two methods:

- ▶ Process monitoring: Detects the termination of a process by using the RSCT RMC capability.
- ▶ Custom monitoring: Monitors the health of an application by using a monitor method, such as a script that you define.

With PowerHA, you may have multiple monitors for an application.

When defining your custom monitoring method, consider the following points:

- ▶ You can configure multiple application monitors, each with unique names, and associate them with one or more application servers.
- ▶ The monitor method must be an executable program (such as a shell script) that tests the application and exits, and returns an integer value that indicates the application’s status. The return value must be zero if the application is healthy, and must be a nonzero value if the application failed.
- ▶ PowerHA does not pass arguments to the monitor method.
- ▶ The monitoring method logs messages to the following monitor log file:
`/var/hacmp/log/clappmond.application_name.resource_group_name.monitor.log`
Also, by default, each time the application runs, the monitor log file is overwritten.
- ▶ Do not over-complicate the monitor method. The monitor method is terminated if it does not return within the specified polling interval.

Important: Because the monitoring process is time-sensitive, *always* test your monitor method under different workloads to arrive at the best polling interval value.

For more information, see 7.7.9, “Application monitoring” on page 333.

3.9.3 Availability analysis tool

The application availability analysis tool can be used to measure the exact amount of time that any of your PowerHA 7.2.7 defined applications are available. The PowerHA software collects, timestamps, and logs the following items:

- ▶ An application monitor is defined, changed, or removed.
- ▶ An application starts, stops, or fails.
- ▶ A node fails, shuts down, or starts.
- ▶ An RG is taken offline or moved.
- ▶ Application monitoring through multiple monitors is suspended or resumed.

For more information, see 7.7.10, “Measuring application availability” on page 344.

3.9.4 Completing the application planning worksheets

The following worksheets (9 - 11) capture the required information for each application.

Update the application worksheet to include all required information, as shown in Table 3-11.

Table 3-11 Application worksheet

PowerHA 7.2.7 Cluster Worksheet - Part 9 Of 11 Application Worksheet				Date: November 2022
APP1				
Item	Directory	File System	Location	Sharing
Executable Files	/app1/bin	/app1	SAN Storage	Shared
Configuration Files	/app1/conf	/app1	SAN Storage	Shared
Data Files	/app1/data	/app1	SAN Storage	Shared
Log Files	/app1/logs	/app1	SAN Storage	Shared
Start Script	/cluster/local/app1/start.sh	/	rootvg	Not Shared (must be on both nodes)
Stop Script	/cluster/local/app1/stop.sh	/	rootvg	Not Shared (must be on both nodes)
Failover Strategy	Fall over to Node02.			
Normal Start Commands And Procedures	Ensure that the APP1 server is running.			
Verification Commands And Procedures	Run the following command and ensure that APP1 is active. If not, send a notification.			

PowerHA 7.2.7 Cluster Worksheet - Part 9 Of 11 Application Worksheet				Date: November 2022
Normal Start Commands And Procedures	Ensure APP1 stops properly.			
Node Reintegration	Must be reintegrated during a scheduled maintenance window to minimize client disruption.			
APP2				
Item	Directory	File System	Location	Sharing
Executable Files	/app2/bin	/app2	SAN Storage	Shared
Configuration Files	/app2/conf	/app2	SAN Storage	Shared
Data Files	/app2/data	/app2	SAN Storage	Shared
Log Files	/app2/logs	/app2	SAN Storage	Shared
Start Script	/cluster/local/app2/start.sh	/	rootvg	Not Shared (must be on both nodes)
Stop Script	/cluster/local/app2/stop.sh	/	rootvg	Not Shared (must be on both nodes)
Failover Strategy	Fall over to Node01.			
Normal Start Commands And Procedures	Ensure that the APP2 server is running.			
Verification Commands And Procedures	Run the following command and ensure that APP2 is active. If not, send a notification.			
Normal Start Commands And Procedures	Ensure APP2 stops properly.			
Node Reintegration	Must be reintegrated during a scheduled maintenance window to minimize client disruption.			
Comments	Summary of applications.			

Update the application monitoring worksheet to include all the information that is required for the application monitoring tools (Table 3-12).

Table 3-12 Application monitoring worksheet

PowerHA 7.2.7 Cluster Worksheet - Part 10 Of 11 Application Monitoring		Date: November 2022
APP1		
Can this Application Be Monitored with Process Monitor?		Yes
Processes to Monitor		app1
Process Owner		root
Instance Count		1
Stabilization Interval		30
Restart Count		3
Restart Interval		95
Action on Application Failure		Fallover
Notify Method		/usr/es/sbin/cluster/custom/notify_app1
Cleanup Method		/usr/es/sbin/cluster/custom/stop_app1
Restart Method		/usr/es/sbin/cluster/custom/start_app1
APP2		
Can this Application Be Monitored with Process Monitor?		Yes
Processes to Monitor		app2
Process Owner		root
Instance Count		1
Stabilization Interval		30
Restart Count		3
Restart Interval		95
Action on Application Failure		Fallover
Notify Method		/usr/es/sbin/cluster/custom/notify_app2
Cleanup Method		/usr/es/sbin/cluster/custom/stop_app2
Restart Method		/usr/es/sbin/cluster/custom/start_app2

3.10 Planning for resource groups

PowerHA 7.2.7 manages resources through the usage of RGs. Each RG is handled as a unit that can contain different types of resources. Some examples are IP labels, applications, file systems, and VGs. Each RG has preferences that define when and how it is acquired or released. You can fine-tune the non-concurrent RG behavior for node preferences when a node starts, when an RG falls over to another node in a node failure, or when the RG falls back to a reintegrating node.

The following rules and restrictions apply to resources and RGs:

- ▶ To be made highly available by PowerHA, a cluster resource must be part of an RG. If you want a resource to be kept separate, you can define a group for that resource alone. An RG can have one or more resources that are defined.
- ▶ A resource cannot be included in more than one RG.
- ▶ Put the application server and its required resources in the same RG (unless otherwise needed).
- ▶ If you include a node in participating node lists for more than one RG, make sure that the node can sustain all RGs simultaneously.

After you decide what components to group into an RG, plan the behavior of the RG.

Table 3-13 summarizes the basic startup, failover, and fallback behaviors for RGs in PowerHA.

Table 3-13 Resource group behavior

Startup behavior	Fallover behavior	Fallback behavior
Online on Home Node Only (OHNO) for the resource group	<ul style="list-style-type: none">▶ Fall over to the next priority node in the list.▶ Fall over by using dynamic node priority (DNP).	<ul style="list-style-type: none">▶ Never fall back.▶ Fall back to a higher priority node in the list.
Online using node distribution policy	<ul style="list-style-type: none">▶ Fall over to the next priority node in the list.▶ Fallover by using DNP.	Never fall back.
Online on First Available Node (OFAN)	<ul style="list-style-type: none">▶ Fall over to the next priority node in the list.▶ Fall over by using DNP.▶ Bring offline (on error node only).	<ul style="list-style-type: none">▶ Never fall back.▶ Fall back to a higher priority node in the list.
Online on all available nodes	Bring offline (on error node only).	Never fall back.

3.10.1 Resource group attributes

In the following sections, we describe the RG attribute setup.

Startup settling time

Settling time applies only to OFAN RGs, and lets PowerHA wait for a set amount of time before activating an RG. After the settling time, PowerHA activates the RG on the highest available priority node. Use this attribute to ensure that RGs do not bounce between nodes as nodes with increasing priority for the RG are brought online.

If the node that is starting is a home node for this RG, the settling time period is skipped and PowerHA immediately attempts to acquire the RG on this node. For more information about this feature, including how to modify it, see 10.2, “The settling time attribute” on page 442.

Note: This settling is a cluster-wide one that is set for all OFAN RGs.

Dynamic node priority policy

The default node priority order for an RG is the order in the participating node list. By implementing a DNP for an RG, you may go beyond the default failover policy behavior and influence the destination of an RG on failover. The two types of dynamic node priorities are as follows:

- ▶ Predefined RMC based: Standard with the PowerHA base product.
- ▶ Adaptive failover: These additional priorities require customization by the user.

If you decide to define DNP policies by using RMC resource variables to determine the failover node for an RG, consider the following points about the DNP policy:

- ▶ It is most useful in a cluster where all nodes have equal processing power and memory.
- ▶ It is irrelevant for clusters of fewer than three nodes.
- ▶ It is irrelevant for concurrent RGs.

Selecting a takeover node also depends on conditions such as the availability of a network interface on that node. For more information about configuring DNP with PowerHA, see 10.5, “Dynamic node priority” on page 452.

Delayed fallback timer

The delayed fallback timer lets an RG fall back to a higher priority node at a time that you specify. The RG that has a delayed fallback timer that is configured and that is on a non-home node falls back to the higher priority node at the specified time. For more information about this feature, see 10.6, “Delayed fallback timer” on page 463.

Resource group dependencies

PowerHA 7.2.7 offers a wide variety of configurations where you can specify the relationships between RGs that you want to maintain at startup, failover, and fallback. You can configure these items:

- ▶ Parent-child dependencies so that related applications in different RGs are processed in the proper order.
- ▶ Location dependencies so that certain applications in different RGs stay online together on a node or on a site, or stay online on different nodes.
- ▶ Start after/stop after dependencies to allow a more specific and granular option of when to process the RGs.

By default, all RGs are processed in parallel. PowerHA processes dependent RGs according to the order that is dictated by the dependency, and not necessarily in parallel. RG dependencies are acknowledged cluster-wide and override any customization for serial order of processing of any RGs that are included in the dependency. Dependencies between RGs offer a predictable and reliable way of building clusters with multi-tiered applications.

For more information about RG dependencies, see 10.7, “Resource group dependencies” on page 466.

3.10.2 Completing the planning worksheet

The RGs worksheet (Table 3-14) captures all the required planning information for the RGs.

Table 3-14 Resource groups worksheets

PowerHA 7.2.7 Cluster Worksheet - Part 11 Of 11 Resource Groups)		Date: November 2022
Resource Name	AESPArg	NMIXXrg
Participating Node Names	Node01 Node02	Node02 Node01
Inter-Site Management Policy	ignore	ignore
Startup Policy	OHNO	OHNO
Fallover Policy	Fallover to Next Priority Node in List (FONP)	Fallover to Next Priority Node in List (FONP)
Fallback Policy	Fallback to Higher Priority Node (FBHP)	FBHP
Delayed Fallback Timer		
Settling Time		
Runtime Policies		
Dynamic Node Priority Policy		
Processing Order (Parallel, Serial, or Customized)		
Service IP Label	app1svc	app2svc
File Systems	/app1	/app2
File System Consistency Check	fsck	fsck
File Systems Recovery Method	sequential	sequential
File Systems or Directories to Export		
File Systems or Directories to NFS mount (NFSv2/v3)		
File Systems or Directories to NFS mount (NFSv4)		
Stable Storage Path (NFSv4)		
Network for NFS mount		0
VGs	app1vg	app2vg
Concurrent VGs		
Raw Disk PVIDs		
Raw Disk universally unique identifiers (UUIDs)/hdisk		
Tape Resources		
Application Controller	app1	app2

PowerHA 7.2.7 Cluster Worksheet - Part 11 Of 11 Resource Groups)		Date: November 2022
Service IP Label or Address	hanode1	hanode2
Primary Workload Manager Class		
Secondary Workload Manager Class		
Miscellaneous Data		
Auto Import VGs	false	false
User-Defined Resources		
File Systems Mounted before IP Configured.	false	false
Comments	Overview of the 2 Resource Groups	

3.11 Detailed cluster design

Pulling it all together by using the information that was collected during the preceding cluster planning and documented in the planning worksheets, we can now build a detailed cluster diagram. Figure 3-11 on page 125 contains a detailed cluster diagram for our example. This diagram is helpful when you configure the cluster and diagnose problems.

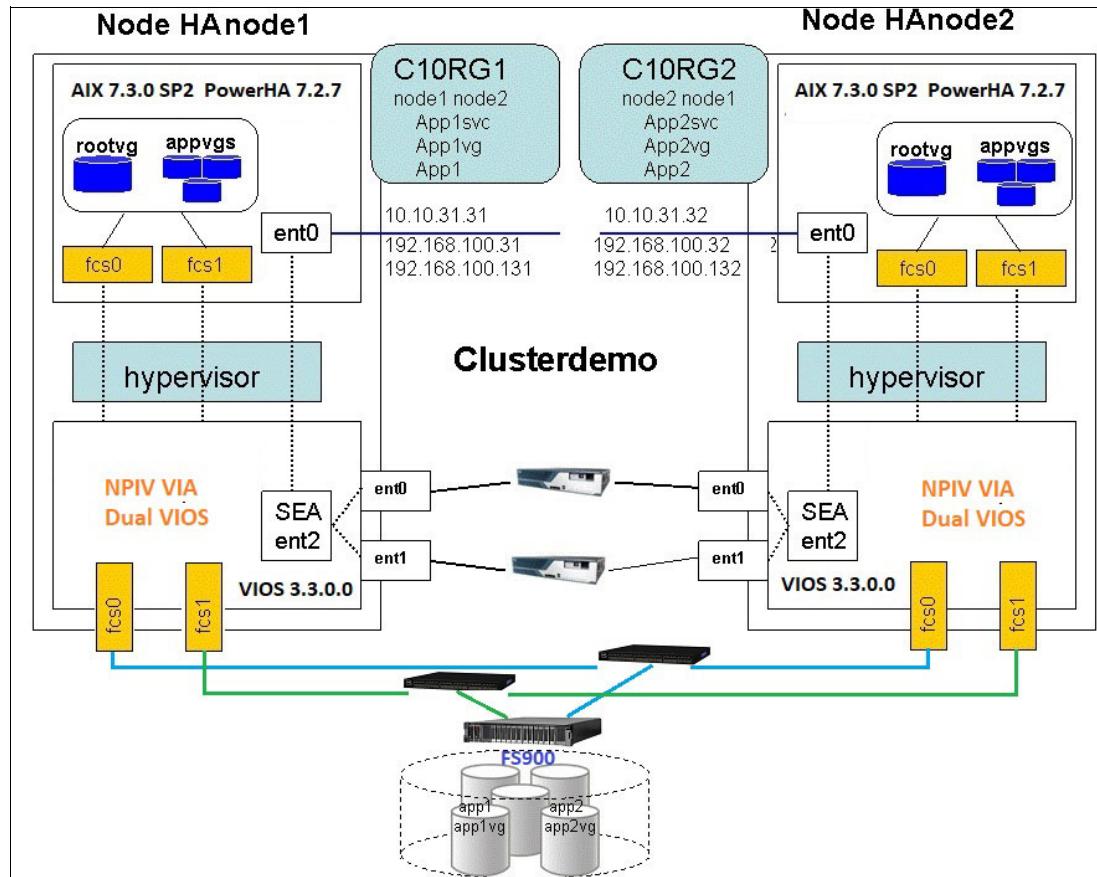


Figure 3-11 Detailed cluster design

3.12 Developing a cluster test plan

As important as planning and configuring your PowerHA 7.2.7 cluster is developing an appropriate test plan to validate the cluster under failure situations, that is, to determine whether the cluster can handle failures as expected. Test, or validate, the cluster recovery before the cluster becomes part of your production environment.

3.12.1 Custom test plan

As with previous releases of PowerHA, you should develop a local set of tests to verify the integrity of the cluster, which typically involves unplugging network cables, downing interfaces, and shutting down cluster nodes to verify cluster recovery. This exercise is still useful because you have the opportunity to simulate failures and watch the cluster behavior. If something does not respond correctly or as expected, stop the tests and investigate the problem. After all tests complete successfully, the cluster can be moved to production.

Table 3-15 outlines a sample test plan that can be used to test our cluster.

Table 3-15 Sample test plan

Cluster Test Plan			
Test #	Test Description	Comments	Results
1	Start PowerHA on Node01.	Node01 starts and acquires the AESPArg RG.	
2	Start PowerHA on Node02.	Node02 starts and acquires the NMIXXrg RG.	
3	Perform a graceful stop without takeover on Node01.	RG AESPArg goes offline.	
4	Start PowerHA on Node01.	Node01 starts and acquires the AESPArg RG.	
5	Perform a graceful stop with takeover on Node01.	RG AESPArg moves to Node02.	
6	Start PowerHA on Node01	Node01 starts and acquires the AESPArg RG.	
7	Fail the service interface on Node01. Unplug if possible.	The service IP moves to the second base adapter.	
8	Reconnect the service interface on Node01.	The service IP remains on the second base adapter.	
9	Fail (unplug) the service interface on Node01 (now on the second adapter).	The service IP (and persistent) moves to the first base adapter.	
10	On Node01 issue a <code>reboot -q</code> to force down the operating system.	Node01 halts and restarts. RG AESPArg moves to Node02.	
11	Restart PowerHA on Node01.	After PowerHA starts, Node01 acquires AESPArg.	
12	Perform a graceful stop without takeover on Node02.	RG NMIXXrg goes offline.	
13	Start PowerHA on Node02.	Node02 starts and acquires the NMIXXrg RG.	
14	Perform a graceful stop with takeover on Node02.	RG NMIXXrg moves to Node01.	
15	Start PowerHA on Node02.	Node02 starts and reacquires the NMIXXrg RG.	
16	Fail the service interface on Node02. Unplug if possible	The service IP moves to the second base adapter.	
17	Reconnect the service interface on Node02.	The service IP remains on the second base adapter.	
18	Fail (unplug) the service interface on Node02 (now on the second adapter).	The service IP (and persistent) moves to the first base adapter.	

Cluster Test Plan			
Test #	Test Description	Comments	Results
19	On Node02, issue <code>reboot -q</code> to force down the operating system.	Node02 restarts RG NMIXXrg and moves to Node01.	
20	Restart Node02 and restart PowerHA.	After PowerHA starts, Node02 reacquires NMIXXrg.	

3.12.2 Cluster Test Tool

PowerHA 7.2.7 includes a Cluster Test Tool to help you test the functions of a cluster before it becomes part of your production environment. The tool can run in two ways:

- ▶ Automated testing

Use the automated test procedure (a predefined set of tests) that is supplied with the tool to perform basic cluster testing on any cluster. No setup is required. You can run the test from SMIT and view the test results from the Cluster Test Tool log file.

- ▶ Custom testing

If you are an experienced PowerHA administrator and want to tailor cluster testing to your environment, you can create custom tests that can be run from SMIT. After you set up your custom test environment, you run the test procedure from SMIT and view test results in the Cluster Test Tool log file.

The Cluster Test Tool uses the PowerHA Cluster Communications daemon to communicate between cluster nodes to protect the security of your PowerHA cluster.

For more information about using the Cluster Test Tool, and details about the tests that it can run, see 6.8, “Cluster Test Tool” on page 226.

3.13 Developing a PowerHA 7.2.7 installation plan

Now that you planned the configuration of the cluster and documented the design, prepare for your installation. If you implement PowerHA 7.2.7 on existing servers, be sure to schedule an adequate maintenance window to allow for the installation, configuration, and testing of the cluster.

If this installation is a new one, allow time to configure and test the basic cluster. After the cluster is configured and tested, you can integrate the required applications during a scheduled maintenance window.

Referring to Figure 3-1 on page 75, you see that there is a preparation step before installing PowerHA. This step is intended to ensure that the infrastructure is ready for PowerHA, which typically involves using your planning worksheets and cluster diagram to prepare the nodes for installation. Ensure that these items are in place:

- ▶ The node software and operating system prerequisites are installed.
- ▶ The network connectivity is properly configured.
- ▶ The shared disks are properly configured.
- ▶ The chosen applications are able to run on either node.

The preparation step can take some time, depending on the complexity of your environment and the number of RGs and nodes to be used. Take your time preparing the environment because there is no purpose in trying to install PowerHA in an environment that is not ready; you will spend your time troubleshooting a poor installation. A well-configured cluster is built on a solid infrastructure.

After the cluster planning is complete and the environment is prepared, the nodes are ready for PowerHA to be installed.

The installation of PowerHA 7.2.7 code is straight forward. If you use the installation CD, use SMIT to install the required file sets. If you use a software repository, you can use NFS to mount the directory and use SMIT to install from this directory. You can also install through NIM.

Ensure that you have licenses for any features you install, such as PowerHA 7.2.7 Enterprise Edition.

After you install the required file sets on all cluster nodes, use the previously completed planning worksheets to configure your cluster. Here, you have a few tools that are available to configure the cluster:

- ▶ The PowerHA SystemMirror User Interface (SMUI)
- ▶ The ASCII SMIT menus
- ▶ The `clmgr` command

Note: When you configure the cluster, be sure to start by configuring the cluster topology, which consists of the nodes, repository disk, and heartbeat type. After the cluster topology is configured, verify and synchronize the cluster, which creates the CAA cluster.

After the topology is successfully verified and synchronized, start the cluster services and verify that all is running as expected. Identify any networking issues before configuring the cluster resources.

After you configure, verify, and synchronize the cluster, run the automated Cluster Test Tool to validate the cluster functions. Review the results of the test tool, and if it was successful, run any custom tests that you want to perform further verification. Verify any error notification that you included.

After successful testing, take an `mksysb` of each node and a cluster snapshot from one of the cluster nodes. The cluster is ready for production. Standard change and problem management processes now apply to maintain application availability.

3.14 Backing up the cluster configuration

The primary tool for backing up the PowerHA 7.2.7 cluster is the cluster snapshot. The primary information that is saved in a cluster snapshot is the data that is stored in the High Availability Cluster Multiprocessing (HACMP) Configuration Database classes (such as HACMPcluster, HACMPnode, HACMPnetwork, and HACMPdaemons). This information is used to re-create the cluster configuration when a cluster snapshot is applied.

The cluster snapshot does not save any user-customized scripts, applications, or other non PowerHA configuration parameters. For example, the names of application servers and the locations of their start and stop scripts are stored in the HACMPserver Configuration Database object class. However, the scripts themselves and also any applications they might call are not saved.

The cluster snapshot utility stores the data that it saves in two separate files:

- ▶ ODM data file (.odm):

This file contains all the data that is stored in the HACMP Configuration Database object classes for the cluster. This file is given a user-defined basename with the .odm file extension. Because the Configuration Database information is largely the same on every cluster node, the cluster snapshot saves the values from only one node.

- ▶ Cluster state information file (.info):

This file contains the output from standard AIX and PowerHA commands. This file is given the same user-defined base name with the .info file extension. By default, this file no longer contains cluster log information. You can specify in SMIT that PowerHA collects cluster logs in this file when the cluster snapshot is created.

For a complete backup, take an **mksysb** of each cluster node according to your standard practices. Pick one node to perform a cluster snapshot and save the snapshot to a safe location for DR purposes. As a best practice, create the snapshot before taking the **mksysb** of the node so that it is included in the system backup.

Important: You can take a snapshot from any node in the cluster, even if PowerHA is down. However, you can apply a snapshot to a cluster only if all nodes are running the same version of PowerHA and all are available. For more information about creating a snapshot, see [Creating a snapshot of the cluster configuration](#).

Although not related to PowerHA configuration data specifically, PowerHA 7.2.3 and later offers the capability to perform cloud backups to IBM Cloud and Amazon Web Services (AWS). For more information, see [Planning for backing up data by using cloud storage in PowerHA SystemMirror](#).

3.15 Documenting the cluster

It is important to document the cluster configuration to effectively manage the cluster. From managing cluster changes to troubleshooting problems, a well-documented cluster results in better change control and quicker problem resolution.

As a best practice, maintain an accurate cluster diagram that can be used for change and problem management. In addition, PowerHA provides updates to the **c1mgr** command so that you can create an HTML-based report from the cluster.

3.15.1 Native HTML report

The cluster manager command (**c1mgr**) can generate native HTML output. It was first introduced with the base product starting with PowerHA 7.1.3.

Its benefits are as follows:

- ▶ Contains more cluster configuration information than any previous native report.
- ▶ Can be scheduled to run automatically through AIX core abilities (**cron**).
- ▶ Is portable and can be emailed without loss of information.
- ▶ Is fully converted.
- ▶ Allows for inclusion of a company name or logo into the report header.

Its limitations are as follows:

- ▶ A per-node operation means no centralized management.
- ▶ A relatively modern browser is required to use tabs.
- ▶ Officially supported only on Microsoft Internet Explorer and Mozilla Firefox.

The output can be generated for the whole cluster configuration or limited to special configuration items, such as these:

- ▶ nodeinfo
- ▶ rginfo
- ▶ lvinfo
- ▶ fsinfo
- ▶ vginfo
- ▶ Dependencies

Figure 3-12 shows the generated report. The report is far longer than depicted. On a real report, you can scroll through the report page for more details.

The report was generated by running the following command:

```
c1mgr view report cluster file=/tmp/democluster.html type=html
```

Demonstration: For a demonstration about creating a similar cluster report, see this [YouTube video](#).

Name	Status	Site IP Address	Nodes at This Site	Recovery Priority	Primary
Dallas	OFFLINE	-	jessica	1	-
FortWorth	OFFLINE	-	jordan	2	-

Figure 3-12 Sample configuration report

Also, the availability report can be generated in HTML format.

Tip: For a full list of available options, run the following `c1mgr` command:

```
c1mgr view report -h
```

3.16 Change and problem management

When the cluster is running, the job of managing changes and problems begins.

Effective change and problem management processes are imperative to maintaining cluster availability. To be effective, you must have a current cluster configuration handy. You can use the `c1mgr` HTML tool to create an HTML version of the configuration and a current cluster diagram.

Any changes to the cluster should be fully investigated regarding their effect on the cluster functions. Even changes that do not directly affect PowerHA, such as the addition of extra non PowerHA workloads, can affect the cluster. The changes should be planned, scheduled, documented, and then tested on a test cluster before implementing them in production.

To simplify the implementation of changes to the cluster, PowerHA provides the C-SPOC SMIT menus. Whenever possible, use the C-SPOC menus to make changes. With C-SPOC, you can make changes from one node and the change is propagated to the other cluster nodes.

Problems with the cluster should be quickly investigated and corrected. Because the primary job of PowerHA is to mask any errors from applications, it is possible that unless you have monitoring tools in place, you might be unaware of a failover. Ensure that you use error notifications to notify the appropriate staff of failures.

3.17 Planning tools

This section covers some planning tools in greater detail. A sample cluster diagram is provided.

3.17.1 Paper planning worksheets

Planning worksheets provide a basis for planning your environment and also create documentation for later use. We include some paper planning sheets in Appendix A, “Paper planning worksheets” on page 609. We found that tailoring these worksheets into a format that fits our environment is useful.

3.17.2 Cluster diagram

Diagramming the PowerHA 7.2.7 cluster helps you have a clear understanding of the behavior of the cluster and helps identify SPOFs. A sample two-node cluster diagram is shown in Figure 3-13.

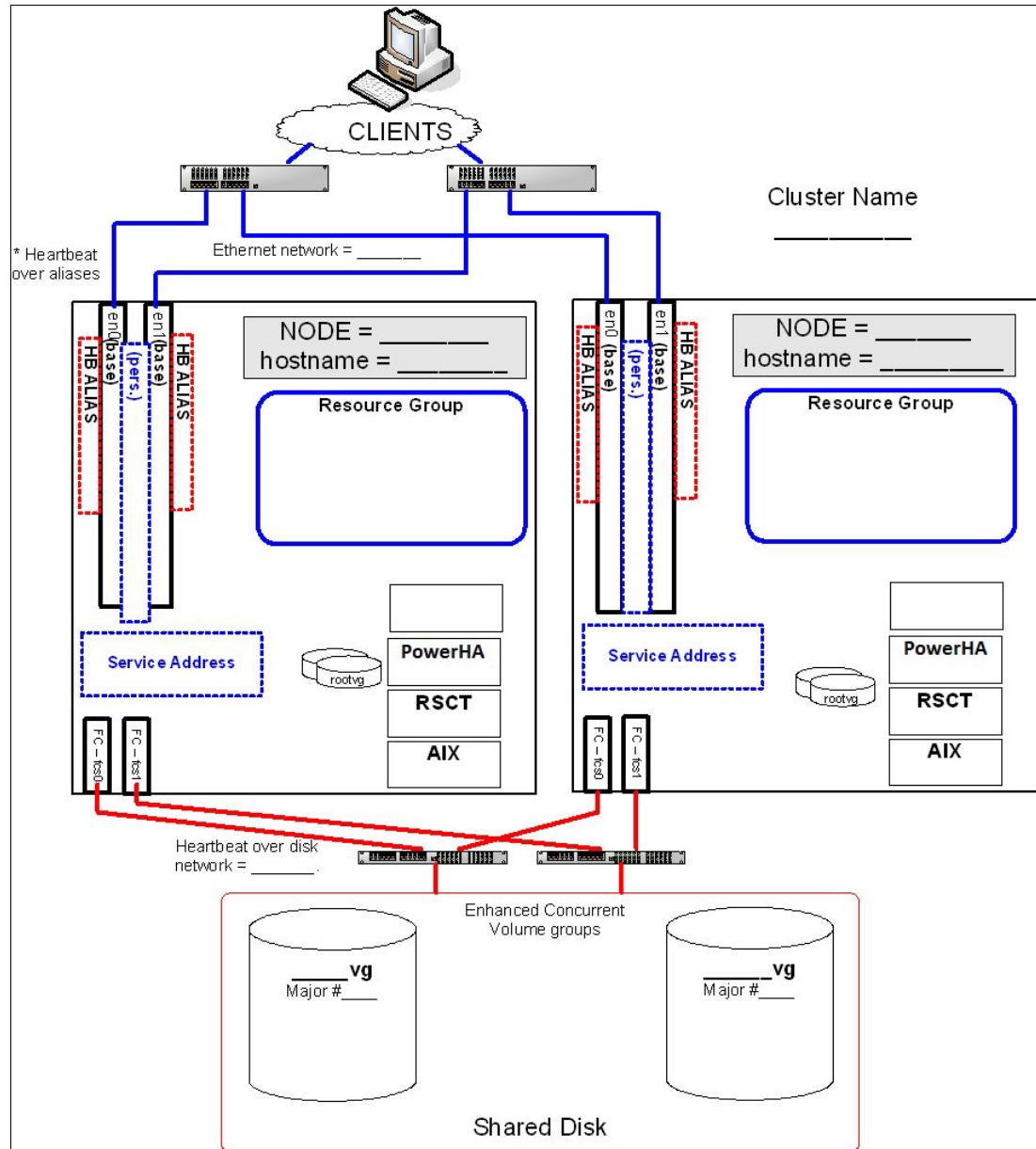


Figure 3-13 Sample cluster diagram



Installation and configuration

In this chapter, we describe the general steps for implementing a simple, two-node PowerHA cluster, including all the needed preparation from both the hardware and software perspectives.

This chapter contains the following topics:

- ▶ Basic steps to implement a PowerHA cluster
- ▶ Configuring PowerHA
- ▶ Installing the SMUI

4.1 Basic steps to implement a PowerHA cluster

In this section, we present the basic steps to implement a PowerHA cluster. Although the target configuration might differ slightly from one implementation to another one, the basic steps are the same, with certain sequence changes.

The basic steps for implementing a high-availability cluster are as follows:

1. Plan.

This step is perhaps the most critical because it requires knowledge and understanding of your environment. You create the full documentation of the entire environment and have a better overview.

Thorough planning is the key for a successful cluster implementation. For more information about planning, see 3.1, “High availability planning” on page 74.

Note: In addition to the cluster configuration, the planning phase should also provide a cluster testing plan. Use this testing plan in the final implementation phase, and also during periodic cluster validations.

2. Install and connect the hardware.

In this step, prepare your hardware environment according to the configuration that you identified during the planning phase. Perform the following tasks:

- a. Install the server hardware (racks, power, Hardware Management Console (HMC), and other items).
 - b. Connect systems to the local networking environment.
 - c. Connect systems to a storage area network (SAN).
 - d. Install and configure Virtual I/O Server (VIOS) (if applicable).
 - e. Create and configure the logical partition (LPAR) profile and do the mapping (if applicable).
 - f. Do the Client/VIOS Fibre Channel Switch (FCS) mapping and create a virtual network (if applicable).
3. Install and configure the base operating system (BOS) (AIX) and PowerHA prerequisites by performing the following steps:
 - a. Install the BOS, applications, and PowerHA prerequisites by using a DVD, vtopt, or Network Install Manager (NIM) according to local rules.
 - b. Configure the local networking environment (the TCP/IP configuration, which includes interfaces, name resolution, and others).
 - c. Configure users, groups, authentication, and others.
 4. Validate the Reliable Scalable Cluster Technology (RSCT) and Cluster Aware AIX (CAA) file sets, check and install any needed Hiper APARs, and check that all interim fixes are installed on all cluster nodes.

5. Configure the shared storage.

Depending on the storage subsystem, the storage configuration can consist of these tasks:

- Configure the storage device drivers and multipath extensions (if applicable), or use AIX multipath I/O (MPIO).
- Configure physical-to-logical storage, such as redundant array of independent disks (RAID) arrays, logical unit numbers (LUNs), storage protection, and others.
- Configure storage security, such as LUN masking, SAN zoning, and I/O groups (where applicable).
- Configure the storage access method for the application (file systems, raw logical volumes (LVs), or raw disks for ASM).
- Configure an extra 1 GB shared LUN between all cluster nodes for the cluster repository disk.

6. Install and configure the application software.

The application software must be configured and tested to run as a stand-alone system. Also, you perform manual movement and testing of the application on all nodes that are designated for the application in the high availability (HA) cluster, as follows:

- a. Create and test the application start and stop scripts (before integrating the application into PowerHA). Make sure that the application can recover from unexpected failures, and that the application start and stop scripts function as expected on all nodes that are designated for running this application. Also, check the time for the start and stop execution for event timeout tunables.
- b. Create and test the application monitoring scripts (if you want) on all nodes that are designated to run the application. The script can be a simple script to monitor a `#ps -ef` process if still running.

7. Install the PowerHA software.

Installing PowerHA can be performed by using the System Management Interface Tool (SMIT), `installp`, or NIM. A restart is no longer required by PowerHA. However, a restart might be required by RSCT prerequisites.

Example 4-1 shows a list of installed PowerHA file sets.

Example 4-1 PowerHA file sets installed

# ls1pp -L cluster.*	Fileset	Level	State	Type	Description (Uninstaller)
<hr/>					
	cluster.adt.es.client.include	7.2.7.0	C	F	PowerHA SystemMirror Client Include Files
<hr/>					
	cluster.adt.es.client.samples.clinfo	7.2.7.0	C	F	PowerHA SystemMirror Client CLINFO Samples
<hr/>					
	cluster.adt.es.client.samples.clstat	7.2.7.0	C	F	PowerHA SystemMirror Client Clstat Samples
<hr/>					
	cluster.adt.es.client.samples.libcl	7.2.7.0	C	F	PowerHA SystemMirror Client LIBCL Samples
<hr/>					
	cluster.es.client.clcomd	7.2.7.0	C	F	Cluster Communication Infrastructure
<hr/>					
	cluster.es.client.lib	7.2.7.0	C	F	PowerHA SystemMirror Client

					Libraries
cluster.es.client.rte	7.2.7.0	C	F	PowerHA SystemMirror Client Runtime	PowerHA SystemMirror Client Runtime
cluster.es.client.utils	7.2.7.0	C	F	PowerHA SystemMirror Client Utilities	PowerHA SystemMirror Client Utilities
cluster.es.cspoc.cmds	7.2.7.0	C	F	CSPOC Commands	CSPOC Commands
cluster.es.cspoc.rte	7.2.7.0	C	F	CSPOC Runtime Commands	CSPOC Runtime Commands
cluster.es.migcheck	7.2.7.0	C	F	PowerHA SystemMirror Migration support	PowerHA SystemMirror Migration support
cluster.es.server.diag	7.2.7.0	C	F	Server Diags	Server Diags
cluster.es.server.events	7.2.7.0	C	F	Server Events	Server Events
cluster.es.server.rte	7.2.7.0	C	F	Base Server Runtime	Base Server Runtime
cluster.es.server.testtool	7.2.7.0	C	F	Cluster Test Tool	Cluster Test Tool
cluster.es.server.utils	7.2.7.0	C	F	Server Utilities	Server Utilities
cluster.es.smui.agent	7.2.7.0	C	F	SystemMirror User Interface - agent part	SystemMirror User Interface - agent part
cluster.es.smui.common	7.2.7.0	C	F	SystemMirror User Interface - common part	SystemMirror User Interface - common part
cluster.license	7.2.7.0	C	F	PowerHA SystemMirror Electronic License	PowerHA SystemMirror Electronic License

8. Define the base cluster.

Because PowerHA provides various configuration tools, such as auto-discovery and Cluster Single Point of Control (C-SPOC), you can choose between creating most of your shared resource environment either within or outside of PowerHA. However, using the tools within PowerHA often makes it less likely something is missed or forgotten.

During cluster creation, you can choose to use either a *Standard Configuration*, or the *Custom Configuration* for more complex configurations. Also, you can choose between manually entering all topology data or by using the PowerHA discovery feature, which eases cluster configuration by building picklists to use. This approach makes it less likely to enter a typographical error while configuring.

9. Synchronize the cluster topology and start PowerHA services (on all nodes).

Verify and synchronize the cluster topology and start the cluster services. Verifying and synchronizing at this stage eases the subsequent implementation steps because detecting configuration mistakes and correcting them in this phase provides a reliable cluster topology for further resource configuration.

PowerHA is designed to auto-sync and verify before it starts cluster services. During this automatic verification and synchronization, PowerHA SystemMirror can discover and corrects several common configuration issues. However, these auto-corrective actions work only when the entire cluster is offline. If any node is already active in the cluster while activating another node, the auto-corrective actions are *not* available.

This automatic behavior ensures that if you did not manually verify and synchronize your cluster before you start cluster services, PowerHA SystemMirror does so. Throughout this section, automatic verification and synchronization is often referred to as verification.

10. Configure the cluster resources.

Although there are numerous resource types that are available, the most common are:

- Service IP addresses (labels).
- Application controllers (application start and stop scripts).
- Application monitors (application monitoring scripts and actions).

11. Configure the cluster resource groups (RGs) and shared storage.

Cluster RGs can be seen as containers that group all the cluster resources to be managed by PowerHA. Initially, the RGs are defined as empty containers. Complete the following steps:

- a. Define the RGs.
- b. Define the shared storage (volume groups (VGs), LVs, file systems, OEM disk methods, and others).
- c. Add the resources to the RG, such as service IP labels, application servers, VGs, and application monitors.

12. Synchronize the cluster.

Because the PowerHA topology is already configured and assuming that the PowerHA services were started, after synchronizing the cluster, the RGs are brought online through dynamic reconfiguration.

Note: It is a best practice to perform a cluster synchronization after each modification so that the number of errors can be minimized and you can more efficiently troubleshoot them. Although changes can be made on any node in the cluster, any changes that are made should be limited to *only* one node in between cluster syncs.

13. Test the cluster.

After the cluster is in the *stable* state, test the cluster.

Note: There is a cluster test tool that is included, as described in 6.8, “Cluster Test Tool” on page 226, to simulate cluster component failure, but it is a best practice to perform a manual testing of the cluster.

Testing includes these activities:

- Stop and start the cluster on each node.
- Perform controlled RG moves between nodes.
- Simulate a hard failover by halting an RG-owning node.
- Simulate an interface or network failure.
- Simulate a storage access loss.
- Simulate an application failure when using application monitoring.
- Document the tests and results.
- Update the cluster documentation.

4.2 Configuring PowerHA

This section shows how to create a basic cluster configuration by using various tools and menus that are provided. You can perform a cluster configuration in two ways:

- ▶ Standard
- ▶ Custom

Before you decide which approach to use, make sure that you do the necessary planning, and that the documentation for your cluster is available for use. For more information, see Chapter 3, “Planning” on page 73.

You can use the `c1mgr` utility to configure a cluster. The commands for creating a basic cluster are shared throughout this chapter. For more information about using `c1mgr`, see the [PowerHA SystemMirror for AIX guide](#).

Demonstration: For a demonstration about configuring a two-node PowerHA cluster by using the SystemMirror User Interface (SMUI), see this [YouTube video](#).

Regardless of which method is used, make sure that the following items are true:

- ▶ `/etc/hosts` is populated with all the cluster used IP addresses.
- ▶ `/etc/cluster/rhosts` file is populated with either the IP address or the hostname of each node in the cluster and that `c1comd` refreshed, as shown in Example 4-3 on page 140.

In the following scenario, we configure a typical two-node hot standby cluster by using the standard method within SMIT.

4.2.1 General considerations for each configuration method

Consider the information in this section to help you determine which one of the configuration methods should be used for your cluster.

When to use the standard configuration path

By using the standard configuration path, you have the opportunity to add the basic components to the PowerHA Configuration Database (ODM) in a few simple steps. This configuration path automates the discovery and configuration information selection, and chooses default behaviors for networks and RGs.

The following prerequisites, assumptions, and defaults apply for the Standard Configuration Path:

- ▶ PowerHA software must be installed on all nodes of the cluster, as shown in Example 4-1 on page 135.
- ▶ All network interfaces must be configured at the operating system level. All the nodes must be able to communicate with each other.
- ▶ Check the cluster node communication on both nodes by running the following commands:

```
#clrsh -n <node1_name> date  
#clrsh -n <node2_name> date
```
- ▶ All boot and service IP addresses must be configured in `/etc/hosts`.

- ▶ When you use the standard configuration path, and the information that is required for configuration is on remote nodes, PowerHA automatically discovers the necessary cluster information for you. Cluster discovery runs automatically when you use the standard configuration path.
- ▶ PowerHA assumes that all network interfaces on a physical network belong to the same PowerHA network. If you do not want an interface to be part of the cluster, it must be listed in the /etc/cluster/ifrestrict file, or added to a *private* network.
- ▶ The hostname must resolve to an interface, which by default is the same interface as the cluster node names.
- ▶ One available shared disk of at least 1 GB is used for the cluster repository disk.
- ▶ CAA service checks are performed on all nodes as follows:

```
# egrep "caa|clusterconf" /etc/services /etc/inetd.conf /etc/inittab
/etc/services:clcomd_caa          16191/tcp
/etc/services:caa_cfg            6181/tcp
/etc/inetd.conf:caa_cfg stream  tcp6    nowait  root   /usr/sbin/clusterconf
clusterconf >>/var/adm/ras/clusterconf.log 2>&1
/etc/inittab:clusterconf:23456789:once:/usr/sbin/clusterconf
```

- ▶ To check the physical volume identifiers (PVIDs), VG state, universally unique identifiers (UUIDs), and UDIDs of the physical volumes (PVs) run **1spv -u**.
- ▶ Verify that all shared disks, including the repository, have ODM Reservation Policy is set to NO RESERVE on all nodes, as shown in Example 4-2. If any of them say anything else. the attribute should be changed where needed by running this command:

```
chdev -l hdisk# -a reserve_policy=no_reserve -P
```

The **-P** option is needed only if the disk is busy in an active volume. The option sets the value to become active on the next restart.

Example 4-2 Reserve policy check on all shared disks

```
[karimB]@[HAXD7270/AIX73-00 @ hacmp59] / : # devrsrv -c query -l hdisk0
Device Reservation State Information
=====
Device Name          : hdisk0
Device Open On Current Host? : YES
ODM Reservation Policy : NO RESERVE
Device Reservation State : NO RESERVE
```

- ▶ If you use virtual Ethernet interfaces, configure the netmon.cf file, as described in 12.5, “Understanding the netmon.cf file” on page 517, or set your virtual adapters to use the poll_uplink capability, as described in 12.6, “Using poll_uplink” on page 520.

When to use the custom configuration path

To configure the less common cluster elements, or if connectivity to each of the cluster nodes is not available at configuration time, you can manually enter the information by using the custom configuration path.

Using the options under the Custom Configuration menu, you can add the basic components of a cluster to the PowerHA configuration database, and also other types of behaviors and resources. Use the custom configuration path to customize the cluster components, such as policies and options that are not included in the standard menu.

Use the custom configuration path if you plan to use any of the following options:

- ▶ Custom cluster initial setup
- ▶ Custom cluster Tunables and Heartbeat
- ▶ Custom Disk Methods
- ▶ Custom Volume Group Methods
- ▶ Custom File System Methods
- ▶ Customize Resource Recovery
- ▶ Customize Inter-Site Resource Group Recovery
- ▶ Create User-Defined Events
- ▶ Modify Pre/Post Event commands
- ▶ Remote Notification Warnings
- ▶ Change Warning Notification time
- ▶ Change System Events (rootvg)
- ▶ Advance method of Cluster Verification and Synchronization

4.2.2 Standard configuration path

When using the standard configuration path, discovery is automatically done on the nodes for network interfaces and shared disks. This discovery process helps generate picklists in other SMIT menus so that the device can be chosen instead of manually entered, which helps minimize user error.

Before configuring the cluster, make sure that /etc/cluster/rhosts is populated with the hostname IP addresses for each node in the cluster on every node in the cluster. Also, the **c1cmd** daemon must be running and refreshed on each node, as shown in Example 4-3.

Example 4-3 Configuring /etc/hosts and /etc/cluster/rhosts

```
[karimB]@[HAXD7270/AIX73-00 @ hacmp59] / : # more /etc/cluster/rhosts
10.1.1.59
10.1.1.60
192.168.100.59
192.168.100.60

[karimB]@[HAXD7270/AIX73-00 @ hacmp59] / : #refresh -s c1cmd
0513-095 The request for subsystem refresh was completed successfully.
```

4.2.3 Defining clusters, nodes, and networks

To define cluster, nodes, and networks, run the **smitty sysmirror** command, select **Cluster Nodes and Networks** → **Standard Cluster Deployment** → **Setup a Cluster, Nodes and Networks**, and then press Enter. The final SMIT menu is displayed, as shown in Figure 4-1 on page 141. Select the options that will make the changes, and press Enter. The discovery process runs, which automatically configures the networks so you do not have to do it manually.

When you use the standard configuration path, the node name and system hostname are expected to be the same. If you want them to be different, change them manually.

Setup a Cluster, Nodes, and Networks

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

* Cluster Name	[Entry Fields] [r-testcluster]
New Nodes (via selected communication paths)	[hacmp60]
+ Currently Configured Nodes	hacmp59

Figure 4-1 Adding a cluster and nodes

To create the base cluster with a repository disk by using the **clmgr** command, run the following command:

```
clmgr add cluster <clusternumber> repository=hdiskX
nodes=<node1_hostname>,<node2_hostname> HEARTBEAT_TYPE={unicast|multicast}
TYPE={NSC|SC|LC}
```

For the TYPE field, if you use Stretched Cluster (SC) or Linked Cluster (LC) then the sites must also be defined. If you use LC, then a repository disk at each site must also be defined.

4.2.4 Configuring the repository and the heartbeat method

Continuing from Figure 4-1, you can either back up one panel by pressing the F3 key, or start again from the beginning by running **smitty sysmirror**, selecting **Cluster Nodes and Networks** → **Standard Cluster Deployment** → **Define Repository Disk and Cluster IP Address**, and pressing Enter. The final SMIT menu is shown in Figure 4-2.

Define Repository Disk and Cluster IP Address

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

* Cluster Name	[Entry Fields] r-testcluster
* Heartbeat Mechanism	Unicast +
* Repository Disk	[(00f87c4bf90dad3b)] +
Cluster Multicast Address (Used only for multicast heartbeat)	[]

Figure 4-2 Adding a repository and a heartbeat method

In our example, we use unicast instead of multicast. For the repository disk, we highlight the field and press F4 to see a list of all available shared disks between the two nodes. The disk size must be at least 1 GB, but PowerHA discovery does not verify that the size is adequate.

Verifying and synchronizing a cluster configuration

Synchronize the cluster. The initial synchronization creates the CAA cluster, so if any errors occur during creation, troubleshooting is more efficient.

Run **smitty sysmirror**, select **Cluster Nodes and Networks** → **Verify and Synchronize Cluster Configuration**, and press Enter three times for synchronization to begin. Synchronization can take several minutes to create the CAA cluster.

Alternatively, you can run synchronization by running the **clmgr sync cluster** command.

Verifying the CAA and cluster topology configuration

You can verify whether the CAA cluster was created and what the existing topology is, as shown in Example 4-4.

Example 4-4 CAA cluster and topology information

```
[karimB]@[HAXD7270/AIX73-00 @ hacmp59] / # clcmd lspv |grep caa
hdisk51          00f87c4bf90dad3b                  caavg_private    active
hdisk61          00f87c4bf90dad3b                  caavg_private    active

[karimB]@[HAXD7270/AIX73-00 @ hacmp59] / # lsvg -l caavg_private
caavg_private:
LV NAME           TYPE     LPs    PPs    PVs   LV STATE    MOUNT POINT
caalv_private1   boot      1      1      1    closed/syncd  N/A
caalv_private2   boot      1      1      1    closed/syncd  N/A
caalv_private3   boot      4      4      1    open/syncd   N/A
powerha_crlv    boot      1      1      1    closed/syncd  N/A

[[karimB]@[HAXD7270/AIX73-00 @ hacmp59] / # cltopinfo
Cluster Name: r-testcluster
Cluster Type: Standard
Heartbeat Type: Unicast
Repository Disk: hdisk51 (00f87c4bf90dad3b)
```

There are 2 nodes and 2 networks defined

```
NODE hacmp59:
  Network net_ether_01
    hacmp59 10.1.1.59
  Network net_ether_02
    hacmp59_hb 192.168.100.59

NODE hacmp60:
  Network net_ether_01
    hacmp60 10.1.1.60
  Network net_ether_02
    hacmp60 192.168.100.60
```

No resource groups defined

Verifying the existing disk configuration

To configure the shared Logical Volume Manager (LVM) components, you must make sure that all nodes have access to all shared disks. To do so, run the **lspv** command on both nodes and compare the output of the command to be sure that the unassigned PVs have the same PVID. If any hdisk device does not have a PVID assigned, you can assign one. For example, to assign a PVID to hdisk9, run the following command:

```
chdev -l hdisk9 -a pv=yes
```

The command must be run on all the nodes.

Same PVID: Many versions of PowerHA, both in C-SPOC and even in the initial cluster creation when you choose a repository disk, scan all disks with no PVID and match them up by UUID, and then automatically assign PVIDs to them.

4.2.5 Creating service IP labels

The service IP addresses are logically associated with a specific application. After they are created and assigned to an RG, the service IP is protected by PowerHA. These addresses and labels must be in /etc/hosts.

To create a service IP label, run `smitty sysmirror`, select **Cluster Applications and Resources** → **Resources** → **Configure Service IP Labels/Addresses** → **Add a Service IP Label/Address**, and then press Enter. Then, choose the network from the list. The final SMIT menu is displayed, as shown in Figure 4-3.

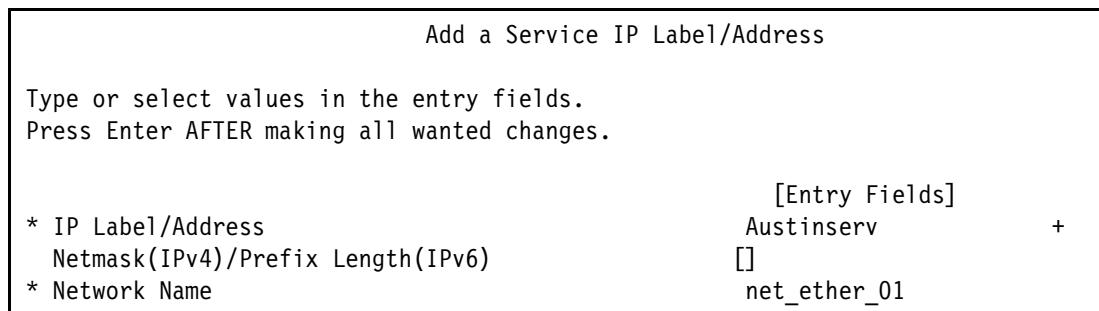


Figure 4-3 Add a Service IP Label/Address panel

For the IP Label/Address field, press F4, and a list is generated from the entries in the /etc/hosts file that are not already defined to the cluster.

To create a service IP label by using the `clmgr` command, run the following command:
`clmgr add service_ip Austinserv network=net_ether_01`

4.2.6 Creating a resource group

When we created our shared VG, an RG that is named demoRG was created automatically. If you want to create an RG manually or other RGs, run `smitty sysmirror`, select **Cluster Applications and Resources** → **Resource Groups** → **Add a Resource Group**, and then press Enter.

The final SMIT menu is shown in Figure 4-4.

Add a Resource Group		
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.		
* Resource Group Name		[Entry Fields]
* Participating Nodes (Default Node Priority)		[demoRG] [hacmp59 hacmp60] +
Startup Policy	Online On Home Node 0> +	
Fallover Policy	Fallover To Next Prio> +	
Fallback Policy	Never Fallback +	

Figure 4-4 Add a Resource Group panel

Complete the fields as shown in Figure 4-4. To complete the Participating Nodes field, enter the information separated by a space, or select from a list by first highlighting the field and then pressing the F4 key.

Important: When you select nodes from the list, they are both displayed, and are listed in alphanumeric order in the Participating Nodes field. This arrangement might lead to an unintended result. For the example in Figure 4-4, if we chose both nodes from the list, they will be in the order that is shown. If we wanted hacmp60 to be listed first, we must manually type it in the field.

For more information about RG policy options, see 2.4.8, “Resource groups” on page 53.

To create an RG by using the **clmgr** command, run the following command:

```
clmgr add rg demoRG nodes=hacmp59,hacmp60 startup=ohn fallback=nfb
```

4.2.7 Creating a shared volume group

PowerHA support requires that you use enhanced concurrent volume groups (ECVGs) as shared VGs. This type of VG can be included in both shared, non-concurrent, and concurrent RGs. Because disk discovery was performed and the disks are known, you can use C-SPOC to create the shared VGs.

Run **smitty cspoc**, select **Storage → Volume Groups → Create a Volume Group**, choose both nodes, choose one or more disks from the picklist, and choose a VG type from the list. The final SMIT menu is displayed, as shown in Figure 4-5 on page 145.

Create a Scalable Volume Group	
Type or select values in the entry fields.	
Press Enter AFTER making all wanted changes.	
[TOP]	[Entry Fields]
Node Names	hacmp59,hacmp60
Resource Group Name	[demoRG] +
PVID	00f87c4bf90dadd2
VOLUME GROUP name	[demovg]
Physical partition SIZE in megabytes	64 +
Volume group MAJOR NUMBER	[41] #
Enable Fast Disk Takeover or Concurrent Access	Fast Disk Takeover +
Volume Group Type	Scalable
CRITICAL volume group?	no +
Max PPs per VG in units of 1024	32 +
Max Logical Volumes	256

Figure 4-5 Creating a shared volume group

Notice the Resource Group Name field. This field provides the option to automatically create the RG and put the VG resource into the RG.

Important: When you choose to create an RG from C-SPOC, the RG is created with the following default policies:

- ▶ Startup: Online on Home Node Only (OHNO)
- ▶ Failover: Failover to next priority node in the list
- ▶ Fallback: Never fallback

You may change these options.

Repeat this procedure for all VGs that will be configured in the cluster.

To create a shared VG by using the **clmgr** command, run the following command:

```
clmgr add volume_group [ <vgname> ] \
    [ NODES=<node#1>,<node#2>[,...]>" ] \
    [ PHYSICAL_VOLUMES=<hdisk#1>[,<hdisk#2>,...]" ] \
    [ TYPE={original|big|scalable|legacy} ] \
    [ RESOURCE_GROUP=<RESOURCE_GROUP> ] \
    [ PPART_SIZE={4|1|2|8|16|32|64|128|256|512|1024} ] \
    [ MAJOR_NUMBER=## ] \
    [ MAX_PHYSICAL_PARTITIONS={32|64|128|256|512|768|1024} ] \
    [ MAX_LOGICAL_VOLUMES={256|512|1024|2048} ] \
    [ CRITICAL={false|true} ] \
    [ ENABLE_LV_ENCRYPTION={yes|no} ]
```

4.2.8 Creating shared logical volumes

After all shared VGs are created, define the LVs that will be part of your VGs by completing the following steps:

1. Enter **smitty cspoc**, and then select **Storage → Logical Volumes**.
2. In this example, we select the VG demovg from the list.
3. On the next panel, we select devices for LV allocation, as shown in Example 4-5.

Example 4-5 C-SPOC creating logical volume disk list

```
+-----+
| Select the Physical Volumes to hold the new Logical Volume |
| Move the cursor to the wanted item and press F7.           |
|                                                               |
| ONE OR MORE items can be selected.                         |
| Press Enter AFTER making all selections.                   |
|                                                               |
| # Reference node      Physical Volume Name               |
|   hacmp59            hdisk6                            |
|                                                               |
| F1=Help            F2=Refresh             F3=Cancel          |
| F7>Select          F8=Image              F10=Exit           |
| F1 Enter=Do        /=Find                n=Find Next       |
| F9+-----+
```

4. We populated the necessary fields, as shown in Example 4-6.

Example 4-6 C-SPOC creating a logical volume

```
Add a Logical Volume

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

[TOP]                                     [Entry Fields]
Resource Group Name                      demoRG
VOLUME GROUP name                       demovg
Node List                                hacmp59,hacmp60
Reference node                           hacmp59
* Number of LOGICAL PARTITIONS          [24]          #
PHYSICAL VOLUME names                  hdisk9
Logical volume NAME                     [karimlv]
Logical volume TYPE                     [jfs2]          +
POSITION on physical volume            outer_middle  +
RANGE of physical volumes              minimum       +
MAXIMUM NUMBER of PHYSICAL VOLUMES    []            #
to use for allocation
Number of COPIES of each logical      1            +
```

The new LV, karimlv, is created, and information is propagated on the other cluster nodes. Repeat this step as needed for each LV.

To create a shared LV by using the **clmgr** command, run the following command:

```
clmgr add logical_volume [ <lvname> ] \
    VOLUME_GROUP=<vgname> \
    LOGICAL_PARTITIONS=## \
    [ DISKS="<hdisk#1>[,<hdisk#2>,...]" ] \
    [ TYPE={jfs|jfs2|sysdump|paging|jfslog|jfs2log|aio_cache|boot} ] \
    [ POSITION={outer_middle|outer_edge|center|inner_middle|inner_edge } ] \
    [ ENABLE_LV_ENCRYPTION={yes|no} ] \
    [ AUTH_METHOD={keyserv|pkcs} ] \
    [ METHOD_DETAILS=<key server ID> ] \
    [ AUTH_METHOD_NAME=<Alias name for auth method>
```

4.2.9 Creating a shared jfs2log logical volume

To create a jfs2log LV (**kloglv** in the demovg VG), we repeat the same procedure as described in 4.2.8, “Creating shared logical volumes” on page 146. In the SMIT panel, we select **jfs2log** as the LV type, as shown in Example 4-7.

Tip: If you intend to use inline logs, you can skip this procedure because it is specified at the time of file system creation.

Example 4-7 C-SPOC creating jfslog logical volumes

Add a Logical Volume

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

[TOP]	[Entry Fields]
Resource Group Name	demoRG
VOLUME GROUP name	demovg
Node List	hacmp59,hacmp60
Reference node	hacmp59
* Number of LOGICAL PARTITIONS	[1] #
PHYSICAL VOLUME names	hdisk9
Logical volume NAME	[kloglv]
Logical volume TYPE	[jfs2log] +
POSITION on physical volume	outer_middle +
RANGE of physical volumes	minimum +
MAXIMUM NUMBER of PHYSICAL VOLUMES	[] #
to use for allocation	
Number of COPIES of each logical	1

Important: If LV type **jfs2log** is created, C-SPOC automatically runs the **logform** command so that the type can be used.

4.2.10 Creating a shared file system

To create a Journaled File System 2 (JFS2) file system on a previously defined LV, complete the following steps:

1. Enter **smitty cspoc** and select **Storage → File Systems**.
2. Choose the VG from the list. The VG in our case is demovg.
3. Choose the type of File System (Enhanced, Standard, Compressed, or Large File Enabled).
4. Select the previously created LV karimlv from the list.
5. Complete the necessary fields, as shown in Example 4-8.

Example 4-8 C-SPOC creating a jfs2 file system on an existing logical volume

Add an Enhanced Journaled File System on a Previously Defined Logical Volume

Type or select values in the entry fields.

Press Enter AFTER making all wanted changes.

[TOP]	[Entry Fields]
Resource Group	demoRG
* Node Names	hacmp59,hacmp60
Logical Volume name	karimlv
Volume Group	demovg
* MOUNT POINT	/
PERMISSIONS	read/write
Mount OPTIONS	[]
Block Size (bytes)	4096
Inline Log?	no
Inline Log size (MBytes)	[]
Logical Volume for Log	kloglv
Extended Attribute Format	Version 1
ENABLE Quota Management?	no
Enable EFS?	no
F1=Help	F2=Refresh
F5=Reset	F6=Command
F9=Shell	F10=Exit
	F3=Cancel
	F7>Edit
	F4>List
	F8=Image
	Enter=Do

Important: File systems are not allowed on VGs that are a resource in an “Online on All Available Nodes” type RG.

The /demofs file system is now created. The contents of /etc/filesystems on both nodes are now updated with the correct jfs2log. If the RG and VG are online, the file system is mounted automatically after creation.

Tip: With JFS2, you may use inline logs that can be configured from the options in the previous example.

Make sure that the mount point name is unique across the cluster. Repeat this procedure as needed for each file system.

To create a shared file system by using the **clmgr** command, run the following command:

```
clmgr add file_system <fsname> \
    VOLUME_GROUP=<group> \
    TYPE=enforced \
    UNITS=## \
    [ SIZE_PER_UNIT={megabytes|gigabytes|512bytes} ] \
    [ PERMISSIONS={rw|ro} ] \
    [ OPTIONS={nodev,nosuid,all} ] \
    [ BLOCK_SIZE={4096|512|1024|2048} ] \
    [ LV_FOR_LOG={<lvname> | "INLINE" } ] \
    [ INLINE_LOG_SIZE=#### ] \
    [ ENABLE_EFS={false|true} ]
```

4.2.11 Creating more application controllers

PowerHA needs a start and a stop script that is used to automatically start and stop the application that is part of the RG. You must make sure that the scripts produce the expected results. Be sure that the scripts exist, are executable, and are greater than zero in size.

To add an application controller, run **smitty sysmirror**, select **Cluster Applications and Resources** → **Resources** → **Configure User Applications (Scripts and Monitors)** → **Application Controller Scripts** → **Add Application Controller Scripts**, and then press Enter.

The final SMIT menu is shown in Figure 4-6.

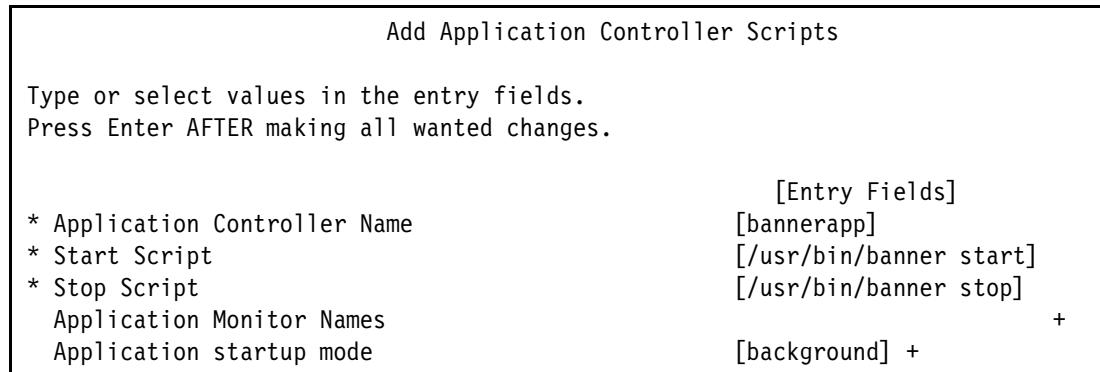


Figure 4-6 Creating an application controller

In our case, we do not have a real application, so we use the **banner** command instead. Repeat as needed for each application.

To create the application controller by using the **clmgr** command, run the following command:

```
clmgr add application_controller bannerapp startscript="/usr/bin/banner start"
stopscript="/usr/bin/banner stop".
```

4.2.12 Adding resources into a resource group

Add your application controller and service IP address into the RG by running **smitty sysmirror**, and then select **Cluster Applications and Resources** → **Resource Groups** → **Change/Show Resources and Attributes for a Resource Group**.

Choose the RG from the list. The final SMIT menu opens, as shown in Figure 4-7.

Change/Show All Resources and Attributes for a Resource Group		
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.		
[TOP]	[Entry Fields]	
Resource Group Name	demoRG	
Participating Nodes (Default Node Priority)	hacmp59 hacmp60	
Startup Policy	Online On Home Node 0>	
Failover Policy	Failover To Next Prio>	
Fallback Policy	Never Fallback	
...		
Service IP Labels/Addresses	[Austinserv]	+
Application Controllers	[bannerapp]	+
..		
Volume Groups	[demovg]	+
Use forced varyon of volume groups, if necessary	false	+
Automatically Import Volume Groups	false	+

Figure 4-7 Adding resources to a resource group

You can press F4 on each of the resource types and choose from the generated list of previously created resources to make sure that they exist and to minimize the chance of errors, such as a typographical error.

To perform this step by using the **clmgr** command, run the following command:

```
clmgr modify rg demoRG service_label=Austinserv volume_group=demovg  
application=bannerapp
```

4.2.13 Verifying and synchronizing a cluster configuration

The configuration of a basic, two-node hot standby cluster is complete. Now, you must synchronize the cluster. This time, use the advanced verify and synchronize option.

Run **smitty sysmirror**, select **Custom Cluster Configuration → Verify and Synchronize Cluster Configuration (Advanced)**, and then press Enter. The menu of options is listed in Figure 4-8 on page 151.

Although most options are self-explanatory, one needs further explanation: Automatically correct errors found during verification. This option is useful and can be used only from this advanced option. It can correct certain problems automatically, or if you run it *interactively*, it prompts you for approval before correcting.

For more information about this option, see 7.6.6, “Running automatic corrective actions during verification” on page 310.

PowerHA SystemMirror Verification and Synchronization		
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.		
[Entry Fields]		
* Verify, Synchronize or Both	[Both]	+
* Include custom verification library checks	[Yes]	+
* Automatically correct errors found during verification?	[Interactively]	+
* Force synchronization if verification fails?	[No]	+
* Verify changes only?	[No]	+
* Logging	[Standard]	

Figure 4-8 Add resources to a resource group

To perform this step by using the clmgr command, run the following command:

```
clmgr sync cluster FIX=yes VERIFY=yes
```

After successful synchronization, you can start testing the cluster. For more information about cluster testing, see 6.8, “Cluster Test Tool” on page 226.

4.3 Installing the SMUI

The PowerHA SMUI was released with PowerHA SystemMirror 7.2.1. It focused on initial customer requests by offering health monitoring and a centralized log view for resolving PowerHA cluster problems. It has evolved across every version to include more administrative tasks. The section covers planning and installing both the SMUI server and the clients.

4.3.1 Planning the SMUI installation

Before you can install the PowerHA SMUI, make sure that you do proper planning to meet certain requirements. The PowerHA SMUI server monitors clusters that are installed with PowerHA SystemMirror 7.2.0 Service Pack (SP) 3 or later releases. The server levels, like NIM, should be the highest of all clusters that are being monitored and maintained by it.

The SMUI server does *not* need to be a cluster node member, and often it is a stand-alone AIX virtual server. It does not have to be dedicated as a SMUI server. For example, in our environment, it is also running on the same system as our NIM server.

Browser support

The PowerHA SMUI is supported on the following web browsers:

- ▶ Google Chrome 57 or later
- ▶ Firefox 54 or later

AIX requirements

For the cluster nodes, the `cluster.es.smui.agent` and `cluster.es.smui.common` file sets must be running one of the following versions of the AIX operating system:

- ▶ AIX 7.1 SP 6, or later
- ▶ AIX 7.2 SP 1 or later
- ▶ AIX 7.3 SP 1 or later
- ▶ OpenSSH on all cluster nodes and the SMUI server
- ▶ OpenSSL on the SMUI server

SMUI file sets

The SMUI file sets are within the PowerHA installation images. The file sets and their details are as follows:

<code>cluster.es.smui.agent</code> :	The SMUI agent file set should be installed on all nodes to be managed by the PowerHA SystemMirror GU.
<code>cluster.es.smui.common</code> :	As the name implies, this file set is common to the cluster nodes and the SMUI server and should be installed on both.
<code>cluster.es.smui.server</code> :	This file set needs to be installed only on the designated SMUI server system.

4.3.2 Installing SMUI clients (cluster nodes)

The `cluster.es.smui.common` and `cluster.es.smui.agent` file sets are part of the group of PowerHA SystemMirror file sets and are installed automatically while installing PowerHA. To check whether the SMUI file sets are installed, run the command that is shown in Example 4-9.

Example 4-9 Checking whether SMUI client file sets are installed

```
# lslpp -L |grep -i smui
cluster.es.smui.agent      7.2.7.0    C      F      SystemMirror User Interface -
cluster.es.smui.common     7.2.7.0    C      F      SystemMirror User Interface -
```

If the SMUI file sets are already installed on all cluster nodes, then skip this section and install the SMUI server. If the file sets are not installed, go to the PowerHA installation media path and complete the following steps:

1. Run `smitty install_all`.
2. Enter the full path of the PowerHA installation images.
3. Press F4 to see the list of file sets.
4. Find and select `cluster.es.smui`, and press F7, as shown in Figure 4-9 on page 153.
5. Press Enter twice to finish.

On successful installation, no further action is required on the SMUI clients.

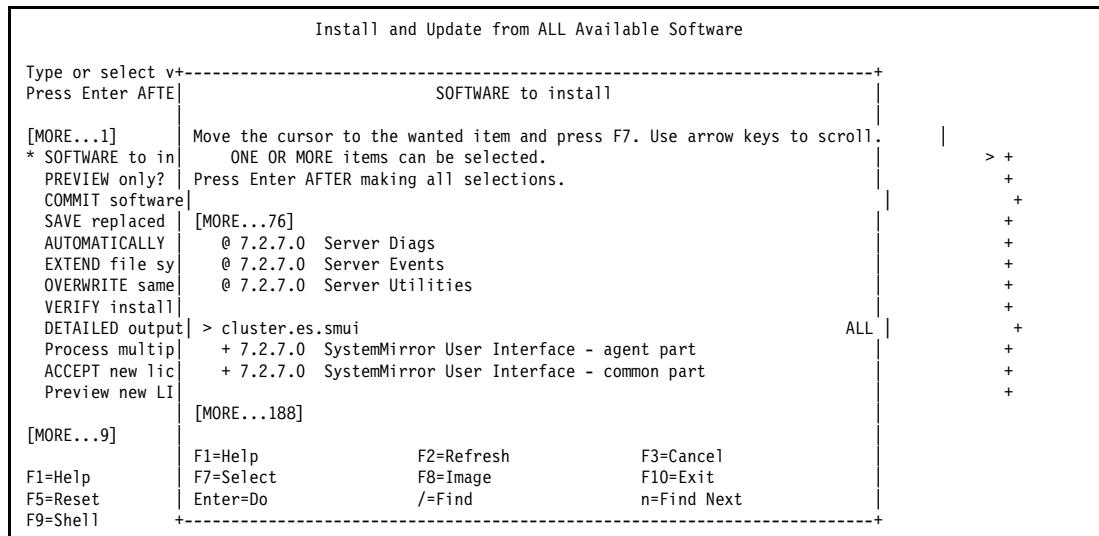


Figure 4-9 Installing the SMUI client file sets

4.3.3 Installing the SMUI server

For the SMUI server, both the `cluster.es.smui.common` and `cluster.es.smui.server` file sets must be installed. To do so, complete the following steps:

1. Run `smitty install_all`.
2. Enter the full path of the PowerHA installation images.
3. Press F4 to see the list of file sets.
4. Find and highlight `cluster.es.smui.common` and `cluster.es.smui.server`, press F7 on each one, as shown in Figure 4-10.
5. Press Enter twice to finish.

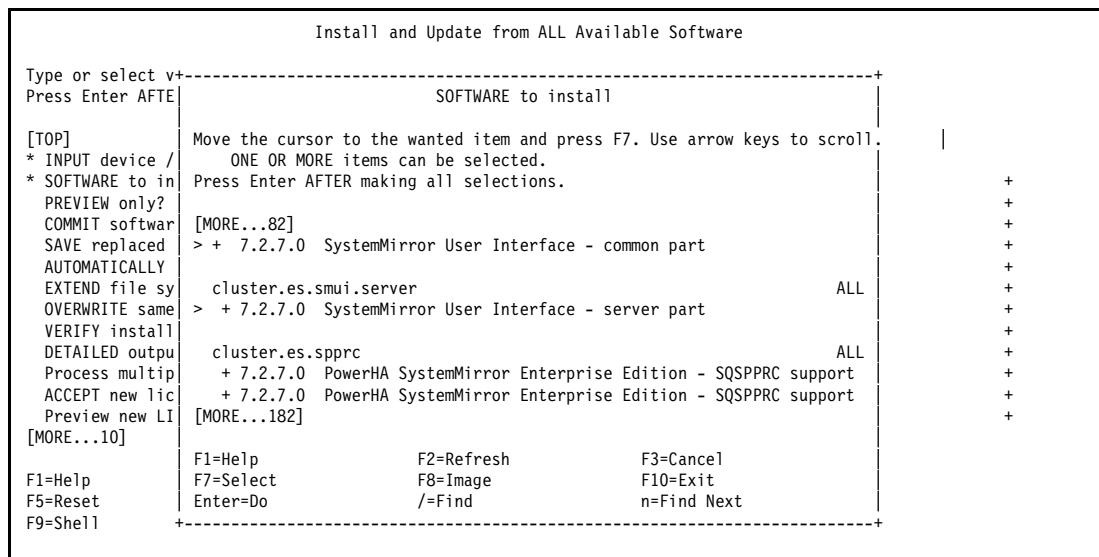


Figure 4-10 Installing the SMUI server file sets

After the installation completes, verify that the file sets are installed, as shown in Example 4-10. Unlike the clients, there is extra work that is required on the server. Run the **smuiinst.ksh** script. By default, this script requires internet access to download more required packages. However, if the server does not have internet access, you can do an offline installation by following the steps in “Offline installation” on page 155.

Example 4-10 Checking whether the SMUI server file sets are installed

```
# lslpp -L |grep -i smui
cluster.es.smui.common    7.2.7.0   C     F     SystemMirror User Interface -
cluster.es.smui.server    7.2.7.0   C     F     SystemMirror User Interface -
```

More packages are required, as shown in Example 4-11. The levels that are installed vary based on the exact PowerHA and AIX level. For example, at the time of writing, the script was *not* downloading AIX 7.3 specific rpms, and they had to be installed manually.

Example 4-11 Rpm packages downloaded by smuiinst.ksh

```
# ./smuiinst.ksh -d ./smui727

Checking if the prerequisites have already been downloaded...
** "info-6.4-1.aix6.1.ppc.rpm" needs to be retrieved.
** "cpio-2.12-2.aix6.1.ppc.rpm" needs to be retrieved.
** "readline-7.0-5.aix6.1.ppc.rpm" needs to be retrieved.
** "libiconv-1.14-1.aix6.1.ppc.rpm" needs to be retrieved.
** "bash-4.4-3.aix6.1.ppc.rpm" needs to be retrieved.
** "gettext-0.19.8.1-3.aix6.1.ppc.rpm" needs to be retrieved.
** "libgcc-4.8.5-1.aix6.1.ppc.rpm" needs to be retrieved.
** "libgcc-4.8.5-1.aix7.1.ppc.rpm" needs to be retrieved.
** "libgcc-8.3.0-2.aix6.1.ppc.rpm" needs to be retrieved.
** "libgcc-8.3.0-2.aix7.1.ppc.rpm" needs to be retrieved.
** "libgcc-8.1.0-2.aix7.2.ppc.rpm" needs to be retrieved.
** "libstdcplusplus-4.8.5-1.aix6.1.ppc.rpm" needs to be retrieved.
** "libstdcplusplus-4.8.5-1.aix7.1.ppc.rpm" needs to be retrieved.
** "libstdcplusplus-8.3.0-2.aix6.1.ppc.rpm" needs to be retrieved.
** "libstdcplusplus-8.3.0-2.aix7.1.ppc.rpm" needs to be retrieved.
** "libstdcplusplus-8.1.0-2.aix7.2.ppc.rpm" needs to be retrieved.
```

Online installation

If the designated SMUI server does have internet access, then you can run the **smuiinst.ksh** script locally. The script automatically downloads and installs the required packages, starts the SMUI server (in this example, phauiserver), and provides the URL information that is needed to access the SMUI.

After the SMUI is installed and you access the URL that is provided, a login window opens, as shown in Figure 4-11 on page 155. The initial login credentials use root and its password. Afterward, more users can be configured.

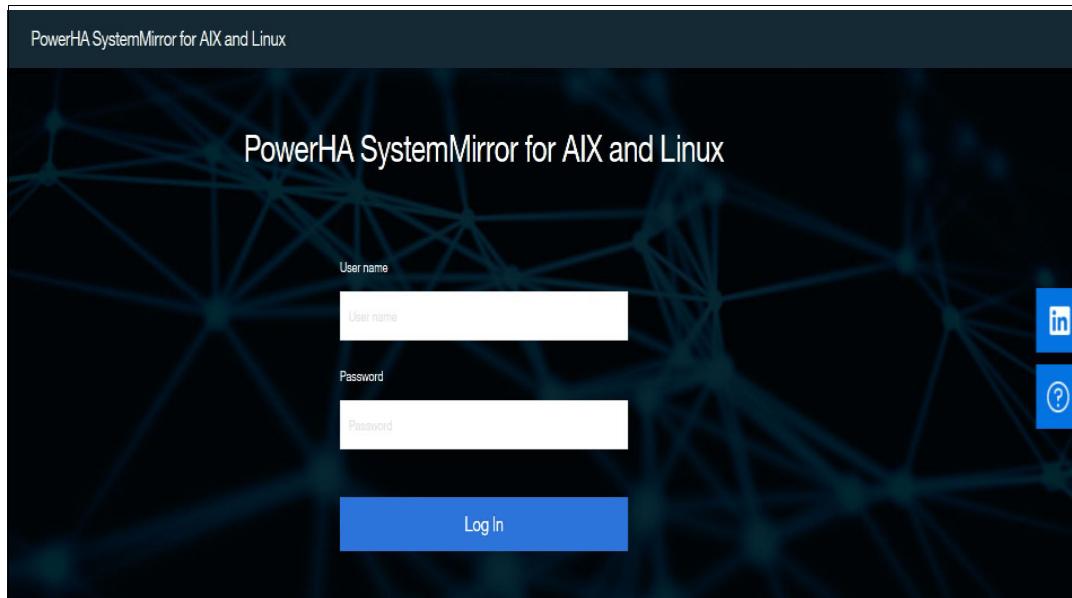


Figure 4-11 SMUI login window

Offline installation

To perform an offline installation by using the **smuiinst.ksh** script, complete the following steps:

1. Copy the `/usr/es/sbin/cluster/ui/server/bin/smuiinst.ksh` file from the PowerHA SMUI server to a system that is running the same operating system and has internet access.
2. From the system that has internet access, run the **smuiinst.ksh -d /directory** command, where **/directory** is the location where you want to download the files that are listed in Example 4-11 on page 154.
3. Copy the downloaded files from `/directory` to a directory on the PowerHA SMUI server. In our example scenario, it is `/home/sbodily/smui727`.
4. From the SMUI server, run the **smuiinst.ksh -i /directory** command, where **/directory** is the location where you copied the downloaded files.

Demonstration: To see a demonstration of performing an offline installation of the PowerHA SMUI, see this [YouTube video](#).

During the **smuiinst.ksh** run time, the rpms are installed and the SMUI server service is started. The script shows a URL for the PowerHA SMUI server, as shown in Example 4-12.

Example 4-12 Rpm packages that are downloaded by smuiinst.ksh

```
#./smuiinst.ksh -i /home/sbodily/smui727
Attempting to install any needed prerequisites.
Attempting to install info-6.4-1...
    Verifying... #####
    Preparing... #####
    Updating / installing...
info-6.4-1 #####
Please check that /etc/info-dir does exist.
You might have to rename it from /etc/info-dir.rpmsave to /etc/info-dir.
"info-6.4-1" installed? Yes.
```

```

Attempting to install cpio-2.12-2...
    Verifying... #####
    Preparing... #####
    Updating / installing... #####
    cpio-2.12-2 #####
        "cpio-2.12-2" installed? Yes.
Attempting to install readline-7.0-5...
    Verifying... #####
    Preparing... #####
    Updating / installing... #####
    readline-7.0-5 #####
        "readline-7.0-5" installed? Yes.
Attempting to install libiconv-1.14-1...
    Verifying... #####
    Preparing... #####
    Updating / installing... #####
    libiconv-1.14-1 #####
        add shr4.o shared members from /usr/lib/libiconv.a to /opt/freeware/lib/libiconv.a
        add shr.o shared members from /usr/lib/libiconv.a to /opt/freeware/lib/libiconv.a
        add shr4_64.o shared members from /usr/lib/libiconv.a to
        /opt/freeware/lib/libiconv.a
        /
        "libiconv-1.14-1" installed? Yes.
Attempting to install gettext-0.19.8.1-3...
    Verifying... #####
    Preparing... #####
    Updating / installing... #####
    gettext-0.19.8.1-3 #####
        "gettext-0.19.8.1-3" installed? Yes.
Attempting to install bash-4.4-3...
    Verifying... #####
    Preparing... #####
    Updating / installing... #####
    bash-4.4-3 #####
        ## Binary "bash" is available in 32 bit and 64 bit ##
        The default that is used is 64 bit
        If 32 bit is needed, change the symbolic link
        for "bash" in the /bin directory
        To do that type:
            # rm -f /bin/bash
            # ln -sf /opt/freeware/bin/bash_32 /bin/bash
        "bash-4.4-3" installed? Yes.

Packaging the Node.js 6 libraries for AIX 6.1...
Packaging the Node.js 16 libraries for AIX 6.1...
Packaging the Node.js 6 libraries for AIX 7.1...
Packaging the Node.js 16 libraries for AIX 7.1...
Packaging the Node.js 16 libraries for AIX 7.2...
Packaging the Node.js libraries for AIX 7.3...
Packaging the Node.js 6 libraries for AIX 7.2...
Packaging the Node.js 6 libraries for AIX 7.3...
Warning: "/usr/es/sbin/cluster/ui/data/sqlite" does not exist. Creating...
"/usr/es/sbin/cluster/ui/data/sqlite" has been created.

Configuring the database in "/usr/es/sbin/cluster/ui/data/sqlite/smui.db" using
"/usr/es/sbin/cluster/ui/server/node_modules/smui-server/resources/0.14.7-ddl.sql"...

```

The database is now configured.

Attempting to start the server...
The server was successfully started.

The installation was completed successfully. To use the PowerHA SMUI,
open a web browser and enter the following URL:

<https://sbodilysmui.labsys.com:8080/#/login>

After you log in, you can add existing clusters in your environment to the
PowerHA SMUI.



Migration

This chapter covers the most common migration scenarios to PowerHA 7.2.7.

This chapter contains the following topics:

- ▶ Migration planning
- ▶ Understanding PowerHA 7.2 migration options
- ▶ Migration scenarios
- ▶ Other migration options
- ▶ Common migration errors

5.1 Migration planning

Proper planning before migrating your existing cluster to IBM PowerHA SystemMirror 7.2.7 is important. A successful migration depends on the basic requirements that are listed in this section.

Before the migration, always have a backout plan in case any problems are encountered. Here are some best practices to use in your backout plan:

- ▶ Create a backup of rootvg.

In most cases of upgrading PowerHA, updating or upgrading AIX is also required. So, always save your existing rootvg. A good method is to create a clone by using **alt_disk_copy** to back up rootvg to another disk on the system. A simple change to the bootlist and a restart can return the system to its original state.

Other options are available, such as **mksysb**, **alt_disk_install**, and **multibos**.

- ▶ Save the existing cluster configuration.

Create a cluster snapshot before the migration. By default, it is stored in the following directory. Copy the directory, and also save a copy from the cluster nodes for more insurance.

/usr/es/sbin/cluster/snapshots

- ▶ Save any user-provided scripts.

Custom events, pre- and post-events, application controller, and application monitoring scripts.

Verify, by using the **1s1pp -h cluster.*** command, that the current version of PowerHA is in the COMMIT state and not in the APPLY state. If not, run **smit install_commit** before you install the most recent software version.

5.1.1 PowerHA SystemMirror 7.2.7 requirements

Various software and hardware requirements must be met for PowerHA SystemMirror 7.2.7.

Software requirements

At the bare minimum, you must use one of these AIX levels:

- ▶ AIX7.1 Technology Level (TL) 05 SP10
- ▶ AIX7.2 TL01 SP6
- ▶ AIX7.2 TL02 SP6
- ▶ AIX7.2 TL03 SP7
- ▶ AIX7.2 TL04 SP6
- ▶ AIX7.2 TL05 SP5
- ▶ AIX7.3 TL00 SP2
- ▶ AIX7.3 TL01 SP1

Important: More APARs are recommended, but the list changes on occasion. For more information, see [PowerHA SystemMirror for AIX Reference Information](#).

Hardware requirements

Use IBM systems that run IBM Power8®, IBM Power9, or IBM Power10 technology-based processors.

Hardware requirements for storage framework communications (optional)

The following adapters are supported by Cluster Aware AIX (CAA) for use as sfwcomm CAA adapters. Some of these adapters are no longer available. However, the sfwcomm feature is rarely used.

- ▶ 4 GB Single-Port Fibre Channel PCI-X 2.0 DDR Adapter (Feature Code 1905; CCIN 1910)
- ▶ 4 GB Single-Port Fibre Channel PCI-X 2.0 DDR Adapter (Feature Code 5758; CCIN 280D)
- ▶ 4 GB Single-Port Fibre Channel PCI-X Adapter (Feature Code 5773; CCIN 5773)
- ▶ 4 GB Dual-Port Fibre Channel PCI-X Adapter (Feature Code 5774; CCIN 5774)
- ▶ 4 Gb Dual-Port Fibre Channel PCI-X 2.0 DDR Adapter (Feature Code 1910; CCIN 1910)
- ▶ 4 Gb Dual-Port Fibre Channel PCI-X 2.0 DDR Adapter (Feature Code 5759; CCIN 5759)
- ▶ 4-Port 8 Gb PCIe2 FH Fibre Channel Adapter (Feature Code 5729)
- ▶ 8 Gb PCI Express Dual Port Fibre Channel Adapter (Feature Code 5735; CCIN 577D)
- ▶ 8 Gb PCI Express Dual Port Fibre Channel Adapter 1Xe Blade (Feature Code 2B3A; CCIN 2607)
- ▶ 3 Gb Dual-Port SAS Adapter PCI-X DDR External (Feature Code 5900 and 5912; CCIN 572A)

Note: The TME attribute is not supported on 16 Gb or faster Fibre Channel (FC) adapters. For the most current list of supported FC adapters, contact your IBM representative.

For more information, see the following resources:

- ▶ [Cluster communication](#)
- ▶ [Setting up cluster storage communication](#)

Cluster repository disk

PowerHA SystemMirror 7.1 and later use a shared disk to store CAA cluster configuration information. Allocate at least 512 MB but no more than 460 GB of disk space for the cluster repository disk. This feature requires that a dedicated shared disk is available to all nodes that are part of the cluster. This disk cannot be used for application storage or any other purpose.

Consider the following information about the repository disk:

- ▶ There is only *one* repository disk.
- ▶ Repository disks *cannot* be mirrored by using AIX Logical Volume Manager (LVM). Therefore, it is a best practice to have a redundant array of independent disks (RAID) that is protected by a redundant and highly available storage configuration.
- ▶ All nodes must have access to the repository disk.

Note: In our experience, even a three-node cluster does not use more than 500 MB; it uses 448 MB of the repository disk. However, as a best practice, use a 1 GB disk for up to four-node clusters, and add 256 MB for each extra node.

Multicast or unicast

With PowerHA 7.2.x, you can use either multicast or unicast for heartbeating. However, if you want to use multicast, make sure that your network both supports and has multicasting enabled. For more information, see 12.1, “Multicast considerations” on page 498.

5.2 Understanding PowerHA 7.2 migration options

Before you begin the migration, you should understand the migration process and all migration scenarios.

5.2.1 Migration options

Consider the following key terms and the options that are available for migrating:

Offline	As the name implies, the cluster must be brought offline on all nodes before performing the migration. During this time, the resources are unavailable, which results in a planned outage. However, the upgrades can be performed in parallel on all nodes in the cluster, which results in the least amount of total time that is required to perform an upgrade, but at the cost of downtime.
Rolling	A rolling migration involves upgrading one node at a time. A node is upgraded and reintegrated into the cluster before the next node is upgraded. A rolling migration requires little downtime, mostly by moving the resources between nodes while each node is being upgraded.
Snapshot	A migration type from one PowerHA version to another one, during which you take a snapshot of the current cluster configuration, stop cluster services on all nodes, install the preferred version of PowerHA SystemMirror, and then convert the snapshot by running the <code>c1convert_snapshot</code> utility. Then, you restore the cluster configuration from the converted snapshot.
Nondisruptive	A node can be <i>unmanaged</i> so that all resources on that node remain operational when cluster services are stopped. This option can be used when applying Service Packs (SPs) to the cluster.
cl_ezupdate	This option is an automated tool that provides the capability to apply both updates or perform upgrades on part of or the entire cluster, often without any downtime that is required.

Important: Nodes in a cluster running two separate versions of PowerHA are considered to be in a *mixed cluster state*. A cluster in this state does not support any configuration changes until all the nodes are migrated. Also, the SystemMirror User Interface (SMUI) is inoperable during this time.

5.3 Migration scenarios

This section describes the test scenarios that are used for each of these migrations options:

- ▶ Rolling migration
- ▶ Snapshot migration
- ▶ Offline migration
- ▶ Nondisruptive migration
- ▶ **c1_ezupdate**
- ▶ Other options

5.3.1 Migration matrix to PowerHA SystemMirror 7.2.7

Table 5-1 shows the migration options between versions of PowerHA.

Table 5-1 Migration matrix table

PowerHA ^a	To 7.2.3	To 7.2.4	To 7.2.5	To 7.2.6	To 7.2.7
From 7.2.2	R,S,O,N ^b , or Z ^b	R,S,O,N ^b , pr Z ^b	R,S,O	Migrate to 7.2.4 or 7.2.5 first so that all options are available.	Migrate to 7.2.4 or 7.2.5 first so that all options are available.
From 7.2.3	N/A	R,S,O,N ^b , or Z ^b	R,S,O,N ^b , or Z ^b	R,S, or O	Migrate to 7.2.5/7.2.6 first so that all options are available.
From 7.2.4	N/A	N/A	R,S,O,N ^b , or Z ^b	R,S,O,N ^b , or Z ^b	R,S, or O
From 7.2.5	N/A	N/A	N/A	R,S,O,N ^b , or Z ^b	R,S,O,N ^b , or Z ^b
From 7.2.6	N/A	N/A	N/A	N/A	R,S,O,N ^b , or Z ^b

a. R=Rolling, S=Snapshot, O=Offline, N=Nondisruptive, and Z=c1_ezupdate.

b. Not officially supported beyond two versions. Assumes that the AIX levels are high enough to support the target PowerHA version.

5.3.2 Rolling migration

This section describes the steps that are involved in performing a rolling migration. The test environment consists of the following levels:

- ▶ Beginning:
 - AIX 7.2 TL2 SP4
 - PowerHA 7.2.4 SP2
- ▶ Ending:
 - AIX 7.2 TL5 SP4
 - PowerHA 7.2.7

Demonstration: For a demonstration of a rolling migration, see this [YouTube video](#).

Complete the following steps:

1. On node jordan (the standby node that will be migrated), stop the cluster services by running **smitty clstop** or **clmgr stop node jordan**.
2. For recovery purposes, create a snapshot. A snapshot can be created at any time, but the nodes themselves should be running.
To create a snapshot by using System Management Interface Tool (SMIT), run **smitty cm_cfg_snap_menu**, and select Create a Snapshot of the Cluster Configuration.
Enter the snapshot name and something for the description, as shown in Example 5-8 on page 170. Press Enter twice.
3. Update or upgrade AIX as needed.

Important: As a best practice, do *not* update PowerHA in this step because it might artificially update the cluster version for both nodes and the cluster in the Object Data Manager (ODM), which causes the following startup verification error:

ERROR: an incomplete cluster migration has been detected.

4. After the restart, verify that caavg_private is active by running **1spv** or **1scluster -i**.
5. Verify that **clcmd** is active by running the following command:
1ssrc -s clcmd
If **clcmd** is not active, activate it by running **startsrc -s clcmd**.
6. Verify the contents of /etc/cluster/rhosts.
Enter either the cluster node hostnames or IP addresses, and only one per line.
7. If the rhosts file was updated, then also refresh **clcmd** by running the following command:
refresh -s clcmd
8. Install all PowerHA 7.2.x file sets (run **smitty update_all**). Make sure to accept the new license agreements.
9. Verify that all cluster file sets are at the new level by running **1slpp -l cluster.***, as shown in Example 5-1.

Example 5-1 Verifying the cluster file set levels

```
# 1slpp -l cluster.*  
File set           Level  State      Description  
-----  
Path: /usr/lib/objrepos  
cluster.adt.es.client.include  
    7.2.7.0  COMMITTED  PowerHA SystemMirror Client  
              Include Files  
cluster.adt.es.client.samples.clinfo  
    7.2.7.0  COMMITTED  PowerHA SystemMirror Client  
              CLINFO Samples  
cluster.adt.es.client.samples.clstat  
    7.2.7.0  COMMITTED  PowerHA SystemMirror Client  
              Clstat Samples  
cluster.adt.es.client.samples.libcl  
    7.2.7.0  COMMITTED  PowerHA SystemMirror Client  
              LIBCL Samples  
cluster.doc.en_US.assist.smartassists.pdf  
    7.2.7.0  COMMITTED  PowerHA SystemMirror Smart  
                        Assists PDF Documentation -  
                        US English
```

cluster.doc.en_US.es.pdf	7.2.7.0	COMMITTED	PowerHA SystemMirror PDF Documentation - US English
cluster.es.assist.common	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist Common Files
cluster.es.assist.db2	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Db2
cluster.es.assist.dhcp	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for DHCP
cluster.es.assist.dns	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for DNS
cluster.es.assist.domino	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM Lotus Domino Server
cluster.es.assist.filenet	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for FileNet P8
cluster.es.assist.ihs	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for IBM HTTP Server
cluster.es.assist.maxdb	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for SAP MaxDB
cluster.es.assist.oraappsrv	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Oracle Application Server
cluster.es.assist.oracle	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Oracle
cluster.es.assist.printServer	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Print Subsystem
cluster.es.assist.sap	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for SAP
cluster.es.assist.tds	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for IBM Tivoli Directory Server
cluster.es.assist.tsadmin	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM Tivoli Storage Manager
Admin			center
cluster.es.assist.tsmclient	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM TSM Client
cluster.es.assist.tsmserver	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM TSM Server
cluster.es.assist.websphere	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for WebSphere
cluster.es.assist.wmq	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for IBM MQ
cluster.es.client.clcomd	7.2.7.0	COMMITTED	Cluster Communication Infrastructure
cluster.es.client.lib	7.2.7.0	COMMITTED	PowerHA SystemMirror Client Libraries
cluster.es.client.rte	7.2.7.0	COMMITTED	PowerHA SystemMirror Client Runtime
cluster.es.client.utils	7.2.7.0	COMMITTED	PowerHA SystemMirror Client Utilities
cluster.es.cspoc.cmds	7.2.7.0	COMMITTED	CSPOC Commands
cluster.es.cspoc.rte	7.2.7.0	COMMITTED	CSPOC Runtime Commands
cluster.es.migcheck	7.2.7.0	COMMITTED	PowerHA SystemMirror Migration

			support
cluster.es.nfs.rte	7.2.7.0	COMMITTED	NFS Support
cluster.es.server.diag	7.2.7.0	COMMITTED	Server Diags
cluster.es.server.events	7.2.7.0	COMMITTED	Server Events
cluster.es.server.rte	7.2.7.0	COMMITTED	Base Server Runtime
cluster.es.server.testtool	7.2.7.0	COMMITTED	Cluster Test Tool
cluster.es.server.utils	7.2.7.0	COMMITTED	Server Utilities
cluster.es.smui.agent	7.2.7.0	COMMITTED	SystemMirror User Interface - agent part
cluster.es.smui.common	7.2.7.0	COMMITTED	SystemMirror User Interface - common part
cluster.license	7.2.7.0	COMMITTED	PowerHA SystemMirror Electronic License
cluster.msg.en_US.es.client	7.2.7.0	COMMITTED	PowerHA SystemMirror Client Messages - US English
cluster.msg.en_US.es.server	7.2.7.0	COMMITTED	Recovery Driver Messages - US English
Path: /etc/objrepos			
cluster.es.assist.db2	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Db2
cluster.es.assist.dhcp	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for DHCP
cluster.es.assist.dns	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for DNS
cluster.es.assist.domino	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM Lotus Domino Server
cluster.es.assist.filenet	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for FileNet P8
cluster.es.assist.ihs	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for IBM HTTP Server
cluster.es.assist.maxdb	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for SAP MaxDB
cluster.es.assist.oraappsrv	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Oracle Application Server
cluster.es.assist.oracle	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Oracle
cluster.es.assist.printServer	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for Print Subsystem
cluster.es.assist.sap	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for SAP
cluster.es.assist.tds	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for IBM Tivoli Directory Server
cluster.es.assist.tsmadmin	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM TSM Admin center
cluster.es.assist.tsmclient	7.2.7.0	COMMITTED	PowerHA SystemMirror SmartAssist for IBM TSM Client
cluster.es.assist.tsmserver	7.2.7.0	COMMITTED	PowerHA SystemMirror

	SmartAssist for IBM TSM Server		
cluster.es.assist.websphere	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for WebSphere
cluster.es.assist.wmq	7.2.7.0	COMMITTED	PowerHA SystemMirror Smart Assist for IBM MQ
cluster.es.client.clcomd	7.2.7.0	COMMITTED	Cluster Communication Infrastructure
cluster.es.client.lib	7.2.7.0	COMMITTED	PowerHA SystemMirror Client Libraries
cluster.es.client.rte	7.2.7.0	COMMITTED	PowerHA SystemMirror Client Runtime
cluster.es.cspoc.rte	7.2.7.0	COMMITTED	CSPOC Runtime Commands
cluster.es.migcheck	7.2.7.0	COMMITTED	PowerHA SystemMirror Migration support
cluster.es.nfs.rte	7.2.7.0	COMMITTED	NFS Support
cluster.es.server.diag	7.2.7.0	COMMITTED	Server Diags
cluster.es.server.events	7.2.7.0	COMMITTED	Server Events
cluster.es.server.rte	7.2.7.0	COMMITTED	Base Server Runtime
cluster.es.server.utils	7.2.7.0	COMMITTED	Server Utilities
cluster.es.smui.agent	7.2.7.0	COMMITTED	SystemMirror User Interface - agent part

Path: /usr/share/lib/objrepos

cluster.man.en_US.es.data	7.2.7.0	COMMITTED	Man Pages - US English
---------------------------	---------	-----------	------------------------

10. Verify that the halevel is correct by running **halevel -s**, as shown in Example 5-2.

Example 5-2 Verifying the halevel output

```
# halevel -s
7.2.7 GA
```

11. Verify that the clconvert.log output on node jessica reports no errors in /tmp/clconvert.log, as shown in snippet Example 5-3.

Example 5-3 The clconvert.log output on node jessica

```
----- log file for cl_convert: Tue Nov 1 17:22:14 CDT 2022
```

Command line is:
`/usr/es/sbin/cluster/conversion/cl_convert -F -v 7.2.6`

No source product was specified.
 Assume that the source and target are the same product.
 Parameters read in from command line are:

Source Product is HAES.
 Source Version is 7.2.6.
 Target Product is HAES.
 Target Version is 7.2.7.
 Force Flag is set.

Cleanup:
 Writing resulting odms to /etc/es/objrepos.
 Restoring the original ODMDIR to /etc/es/objrepos.
 Removing the temporary directory /tmp/tmpodmdir.

Exiting cl_convert.

Exiting with error code 0. Completed successfully.

----- end of log file for cl_convert: Tue Nov 1 17:22:16 CDT 2022

12. Start the cluster services on node jordan by running **smitty clstart** or **clmgr start jordan**.

Because the cluster versions are mixed, cluster verification is skipped during startup, and the startup information states that the verification was skipped, as shown in Example 5-15 on page 176.

The output of the **lssrc -ls clstrmgrES** command on node jordan is shown in Example 5-4.

Example 5-4 The lssrc -ls clstrmgrES output from node jessica

```
## lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#) 8273f4f 43haes/usr/sbin/cluster/hacmpd/main.C, 727, 2238D_aha727, Aug 24
2022 10:16 PM"
build = "Sep 23 2022 08:32:34 2238D_aha727"
CLversion: 20 <--> The migration is still in progress.
local node vrmf is: 7270
cluster fix level is: "0"
```

13. On jessica, stop the cluster services with the Move Resource Groups option (**smitty clstop**). The resource becomes active on node jordan, as shown in Example 5-5.

Example 5-5 Resource group post-move to standby node jordan

```
# clRGinfo
-----
Group Name          Group State    Node
-----
bdbrg              OFFLINE       jessica
                    ONLINE        jordan
```

14. If your environment requires updating or upgrading AIX, perform that step now, but do *not* upgrade PowerHA yet.
15. After the restart, verify that caavg_private is active by running **1spv** or **1scluster -i**.
16. Verify that **clcmd** is active by running **lssrc -s clcmd**.
If **clcmd** is not active, activate it by running **startsrc -s clcmd**.
17. Verify the contents of /etc/cluster/rhosts.
Enter either the cluster node hostnames or IP addresses, one per line.
18. If the rhosts file was updated, then refresh **clcmd** by running **refresh -s clcmd**.
19. Install all PowerHA 7.2.x file sets (run **smitty update_all**). Make sure to accept the new license agreements.
20. Verify that all cluster file sets are at the new level by running **1s1pp -1 cluster.***, as shown in Example 5-1 on page 164.
21. Verify that the halevel level is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
22. Verify that no errors were reported in /tmp/clconvert.log, as shown in Example 5-3 on page 167.
23. Start the cluster services on node jessica by running **smitty clstart** or **clmgr start node jessica**.

24. Verify that the cluster completed the migration on both nodes, as shown in Example 5-6.

Example 5-6 Verifying migration completion on both nodes

```
# clcmd odmget HACMPnode|grep version|uniq  
version = 23  
  
#clcmd odmget HACMPcluster|grep cluster_version  
cluster_version = 23  
cluster_version = 23
```

25. Check for the updated **clstrmgrES** information, as shown in Example 5-7.

Example 5-7 Checking the updated clstrmgrES information

```
# lssrc -ls clstrmgrES  
Current state: ST_STABLE  
sccsid = "@(#) 8273f4f 43haes/usr/sbin/cluster/hacmprd/main.C, 727, 2238D_aha727, Aug 24  
2022 10:16 PM"  
build = "Sep 23 2022 08:32:34 2238D_aha727"  
CLversion: 23 <---- The migration completed.  
local node vrmf is: 7270  
cluster fix level is: "0"  
i_local_nodeid 0, i_local_siteid -1, my_handle 1  
m1_idx[1]=0 m1_idx[2]=1  
There are 0 events on the Ibcast queue  
There are 0 events on the RM Ibcast queue
```

Note: Both nodes must show the same CLversion. If they do not, the migration did not complete successfully. Call IBM Support.

26. Because node jessica was the original hosting or primary node, you might want to move the resource group (RG). To do so, run **clmgr move rg bdbrg node=jessica**.

27. Verify that the cluster is stable and that the RG is online, as shown in Example 5-22 on page 179.

On completing migration, if more SPs are available and must be installed, see 5.3.5, “Nondisruptive migration” on page 175 or 5.3.6, “Migration by using cl_ezupdate” on page 179 (no downtime required).

5.3.3 Snapshot migration

This section describes how to perform a snapshot migration. The test environment consists of the following levels:

- ▶ Beginning:
 - AIX 7.1 TL5 SP1
 - PowerHA 7.2.3 SP3
- ▶ Ending:
 - AIX 7.2 TL5 SP4
 - PowerHA 7.2.7

Demonstration: For a demonstration of a snapshot migration, see this [YouTube video](#).

Complete the following steps:

1. Create a cluster snapshot (if you have not previously created one) and save copies of it off the cluster nodes. A snapshot can be created at any time, but the nodes themselves should be running.

To create a snapshot by using SMIT, run **smitty cm_cfg_snap_menu**, and then select Create a Snapshot of the Cluster Configuration.

Enter snapshot name and something for the description, as shown in Example 5-8. Press Enter twice.

Example 5-8 Create a snapshot by using SMIT

Create a Snapshot of the Cluster Configuration

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

* Cluster Snapshot Name Custom-Defined Snapshot Methods * Cluster Snapshot Description	[Entry Fields] [itzysnapshot] / [] + [see name]
--	--

The snapshot is automatically saved in `/usr/es/sbin/cluster/snapshots`. The output from the snapshot creation is shown in Example 5-9.

Example 5-9 Snapshot creation output

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, more instructions might appear below.

```
clsnapshot: Creating file /usr/es/sbin/cluster/snapshots/itzysnapshot.odm.
clsnapshot: Creating file /usr/es/sbin/cluster/snapshots/itzysnapshot.info.
clsnapshot: Running clsnapshotinfo command on node: jessica...
clsnapshot: Running clsnapshotinfo command on node: jordan...
jordan
```

```
clsnapshot: Succeeded creating Cluster Snapshot: itzysnapshot
```

2. Stop the cluster services on all nodes and bring the RGs offline by running **smitty clstop** or **clmgr stop cluster**.
3. Upgrade AIX (if needed). For more information, see “Software requirements” on page 160. If you are not using Live Update, then restart the system after updating AIX.
4. Verify that caavg_private is active by running **lspv** or **lscluster -i**.
5. Verify that **clcmd** is active by running **lssrc -s clcmd**.
If it is not active, activate it by running **startsrc -s clcmd**.
6. Verify the contents of `/etc/cluster/rhosts`.
Enter either the cluster node hostnames or IP addresses (only one per line).
7. If the rhosts file was updated, then also refresh **clcmd** by running **refresh -s clcmd**.
8. Uninstall the current version of PowerHA by using **smitty remove** and specify the **cluster.*** option.

9. Remove the CAA cluster (optional) by running **rmcluster -r reposhdisckname**.
10. Install the new PowerHA version, including SPs, by running **smitty install_all**. Make sure to accept the new license agreements.
11. Verify that all the cluster file sets are at the new level by running **lslpp -l cluster.***, as shown in Example 5-1 on page 164.
12. Verify that the halevel is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
13. Run the **clconvert_snapshot** command, which is in /usr/es/sbin/cluster/conversion. Use the syntax that is shown in Example 5-10.

Example 5-10 The clconvert_snapshot execution and output

```
# ./clconvert_snapshot -v 7.2.3 -s itzysnapshot
Extracting ODMs from snapshot file... done.
Converting extracted ODM's... done.
Rebuilding the snapshot file... done.
```

14. Verify that no errors were reported in /tmp/clconvert.log, as shown in Example 5-11.

Example 5-11 The clconvert.log output

```
----- log file for clconvert_snapshot: Wed Nov 2 18:31:24 CDT 2022

Command line is:
./clconvert_snapshot -v 7.2.3 -s itzysnapshot

Cbm snapshot file itzysnapshot_cbm.xml or
/usr/es/sbin/cluster/snapshots/itzysnapshot_cbm.xml does not exist.
No source product was specified.
Assumethat the source and the target are the same product.
Parameters read in from command line are:
    Source Product is HAES.
    Source Version is 7.2.3.
    Target Product is HAES.
    Target Version is 7.2.7.
    Snapshot File Flag is set: /usr/es/sbin/cluster/snapshots/itzysnapshot.odm
Initiating execution of cl_convert.
Command line is:
/usr/es/sbin/cluster/conversion/cl_convert -i -F -v 7.2.3 -E

Parameters read in from command line are:
    Source Product is HAES.
    Source Version is 7.2.3.
    Target Product is HAES.
    Target Version is 7.2.7.
    Force Flag is set.
    Ignore Copy Flag is set.

Setup:
Create a temporary directory: /tmp/tmpodmdir
Original directory: /tmp/tmpsnapshotdir
Copy odm's from original to temporary directory.
Changing ODMDIR to /tmp/tmpodmdir.

Initiating execution of ODM manipulator scripts:
    Running script /usr/es/sbin/cluster/conversion/scripts/HAES723toHAES724

    Running script /usr/es/sbin/cluster/conversion/scripts/HAES724toHAES725
```

```

*****
*** ODM Manager version 0.2 ***
*****

Processing script: /usr/es/sbin/cluster/conversion/scripts/HAES724toHAES725

*****
*** End of ODM Manager ***
*****


        Running script /usr/es/sbin/cluster/conversion/scripts/HAES725toHAES726

*****
*** ODM Manager version 0.2 ***
*****


Processing script: /usr/es/sbin/cluster/conversion/scripts/HAES725toHAES726

*****
*** End of ODM Manager ***
*****


        Running script /usr/es/sbin/cluster/conversion/scripts/HAES726toHAES727

*****
*** ODM Manager version 0.2 ***
*****


Processing script: /usr/es/sbin/cluster/conversion/scripts/HAES726toHAES727
......


Cleanup:
    Restoring the original ODMDIR to /etc/objrepos.
    Removing temporary directory /tmp/tm(snapshotdir.

Exiting clconvert_snapshot with error code 0. Completed successfully.

----- end of log file for clconvert_snapshot: Wed Nov  2 18:31:28 CDT 2022

```

15. Restore the cluster configuration from the converted snapshot by running the **smitty sysmirror** command and then selecting **Cluster Nodes and Networks** → **Manage the Cluster** → **Snapshot Configuration** → **Restore the Cluster Configuration From a Snapshot**. Choose the previously created snapshot from the list, and verify the SMIT panel. See Example 5-12.

Example 5-12 Restore the Cluster Snapshot panel

Restore the Cluster Snapshot

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

[Entry Fields]		
Cluster Snapshot Name	itzysnapshot	
Cluster Snapshot Description	see name	
Un/Configure Cluster Resources?	[Yes]	+
Force apply if verify fails?	[No]	+

The restore process automatically synchronizes the cluster, as shown in the output in Example 5-13.

Example 5-13 Restoring the snapshot configuration execution output

COMMAND	STATUS
Command: OK	stdout: yes stderr: no
Before command completion, more instructions might appear below.	
clsnapshot: Removing any existing temporary PowerHA SystemMirror ODM entries...	
clsnapshot: Creating temporary PowerHA SystemMirror ODM object classes...	
clsnapshot: Adding PowerHA SystemMirror ODM entries to a temporary directory.	
clsnapshot: Verifying configuration by using temporary PowerHA SystemMirror ODM entries...	
Verification to be performed on the following:	
Cluster Topology	
Cluster Resources	
Retrieving data from available cluster nodes. This might take a few minutes. Start data collection on node jessica	
Start data collection on node jordan	
Collector on node jordan completed	
Collector on node jessica completed	
Data collection complete	
WARNING: No backup repository disk is UP and not already part of a VG for nodes:	
- jessica	
- jordan	
.....	
cldare: Configuring a 2 node cluster in AIX might take up to 2 minutes. Please wait.	
1 tunable updated on cluster redbook_cluster.	
Adding any necessary PowerHA SystemMirror for AIX entries to /etc/inittab and /etc/rc.net for IP Address Takeover on node jordan.	
Verification completed normally.	
clsnapshot: Creating file /var/hacmp/clverify/pass/clver_pass_snapshot.odm.	
clsnapshot: Succeeded creating Cluster Snapshot: clver_pass_snapshot	
clsnapshot: Synchronizing cluster configuration to all cluster nodes...	
This might take some time. Refer to /var/hacmp/clverify/clsnapshot_cldare_log for more information.	
clsnapshot: Succeeded applying Cluster Snapshot: itzysnapshot	
.....	

16. Verify that the cluster version is the same on all nodes, as shown in Example 5-6 on page 169.
17. Restart the cluster services by running **smitty clstart** or **clmgr start cluster**.
18. Verify that the cluster is stable and that the RG is still online, as shown in Example 5-22 on page 179.
19. Perform cluster validation testing.

5.3.4 Offline migration

This scenario describes how to perform an offline migration. These steps are often performed in parallel because the entire cluster is offline. This method can be the most time-efficient method to perform an upgrade, but it does require a planned outage to do so.

Demonstration: For a demonstration of an offline migration, see this [YouTube video](#).

Complete the following steps:

1. For recovery purposes, create a snapshot. Create a cluster snapshot (if you have not previously created one) and save copies of it off the cluster nodes. A snapshot can be created at any time, but the nodes themselves should be running.

To create a snapshot by using SMIT, run **smitty cm_cfg_snap_menu**, and then select Create a Snapshot of the Cluster Configuration.

Enter the snapshot name and something for the description, as shown in Example 5-8 on page 170. Press Enter twice.

2. Stop the cluster services on all nodes by running **smitty clstop**. Bring all the RGs offline. Also, you can stop the cluster services by running **clmgr stop cluster**.
3. Upgrade AIX (if needed). For more information, see “Software requirements” on page 160. If you are not using Live Update, then you must restart the system after updating AIX.
4. Verify that **caavg_private** is active by running **lspv** or **lsccluster -i**.
5. Verify that **clcomd** is active by running **lssrc -s clcomd**. If **clcomd** is not active, activate it by running **startsrc -s clcomd**.
6. Verify the contents of /etc/cluster/rhosts.

Enter either the cluster node hostnames or IP addresses (one per line).

7. If the rhosts file was updated, then refresh **clcomd** by running **refresh -s clcomd**.
8. Install all PowerHA 7.2.x file sets by running **smitty update_all**.
9. Verify that all the cluster file sets are at the new level by running **ls1pp -l cluster.***, as shown in Example 5-1 on page 164.
10. Verify that the halevel is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
11. Verify that the cluster version is the same on all nodes, as shown in Example 5-6 on page 169.
12. Verify that no errors are reported in /tmp/clconvert.log, as shown in Example 5-3 on page 167.
13. Restart the cluster services by running **smitty clstart** or **clmgr start cluster**.
14. Verify that the cluster is stable and that the RG is still online, as shown in Example 5-22 on page 179.
15. Perform cluster validation testing.

On completing migration, if more SPs are available and must be installed, you can install them by using the non-disruptive or **cl_ezupdate** method without any downtime.

5.3.5 Nondisruptive migration

This scenario describes how to perform a nondisruptive migration.

Note: To use the nondisruptive option, the AIX levels must already be at the supported levels that are required for the version of PowerHA to which you are migrating.

The test environment consists of the following levels:

- ▶ Beginning:
 - AIX 7.3 SP2
 - PowerHA 7.2.6 SP1
- ▶ Ending:
 - AIX 7.3 SP2
 - PowerHA 7.2.7

Demonstration: For a demonstration of a nondisruptive upgrade, see this [YouTube video](#).

Complete the following steps:

1. For recovery purposes, create a snapshot. create a cluster snapshot (if you have not previously created one), and save copies of it off the cluster nodes. A snapshot can be created at any time, but the nodes themselves should be running.

To create a snapshot by using SMIT, run **smitty cm_cfg_snap_menu**, and select Create a Snapshot of the Cluster Configuration.

Enter the snapshot name and something for the description, as shown in Example 5-8 on page 170. Press Enter twice.

2. On the first node **jessica**, which is the primary node in this example, stop the cluster services by running **smitty clstop** with the Unmanage Resource Groups option. You can also stop the node by running **c1mgr stop node jessica manage=unmanage**. If the node is a hot standby node or *not* hosting an RG, then a general stop of the cluster on the node can be performed instead of the **unmanage** option.

Example 5-14 shows node in the node list for *forced down*, which is legacy terminology that is synonymous with unmanaged. The RG and its resources are still online.

The output of the **lssrc -ls clstrmgrES** (note the CLversion and vrfm) **c1RGinfo** commands on node **jessica** is shown in Example 5-14.

Example 5-14 Verifying the node and resources as unmanaged

```
lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#) 7d4c34b 43haes/usr/sbin/cluster/hacmprd/main.C, 61aha_r726,
2205I_aha726, Feb 05 2021 09:50 PM"
build = "Jun 21 2022 10:12:48 2205I_aha726"
CLversion: 22 <----Notice level to compare against throughout migration
local node vrmf is: 7261 <---- Starting level
cluster fix level is: "1"
i_local_nodeid 1, i_local_siteid 1, my_handle 2
m1_idx[1]=0      m1_idx[2]=1      m1_idx[3]=2
Forced down node list: jessica
There are 0 events on the Ibcast queue
There are 0 events on the RM Ibcast queue
```

```
# c1RGinfo
-----
Group Name          Group State    Node
-----
bdbrg              UNMANAGED     jessica
                                UNMANAGED     jordan
-----
```

3. Install all the PowerHA 7.2.x file sets by running **smitty update_all**. Make sure to accept the new license agreements.
4. Verify that all the cluster file sets are at the new level by running **1s1pp -l cluster.***, as shown in Example 5-1 on page 164.
5. Verify that the halevel is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
6. Verify that there are no errors reported in /tmp/clconvert.log, as shown in Example 5-3 on page 167.
7. Start the cluster services on node jessica by running **smitty clstart** or **clmgr start node jessica**.

Important: If you are not using the application monitoring tool within PowerHA and you stop and unmanage a node with an RG that contains an application controller, then a cluster start with the default of auto-manage for the RG starts the application start script. This approach might lead to unwanted results.

Ideally, the application controller should be a smart start script that checks whether the app is running and exits as needed. However, you can edit the script and insert an “exit 0” at the top of the script. When the cluster stabilizes, you can remove this line.

If application monitoring is used, this issue should not happen.

Also on startup, because the cluster versions are mixed, cluster verification is skipped and the startup information will state that verification was skipped, as shown in Example 5-15.

Example 5-15 Verification is skipped on startup of a mixed cluster

Verifying cluster configuration before starting cluster services

Cluster services are running at different levels across the cluster. Verification will not be invoked in this environment.
jessica: start_cluster: Starting PowerHA SystemMirror
...
"jessica" is now online.

The output of the **1ssrc -ls clstrmgrES** and **c1RGinfo** commands on node jessica is shown in Example 5-16. The vrmf changed, but the CLversion did not because all the nodes in the cluster are not upgraded yet.

Example 5-16 Output of 1ssrc -ls clstrmgrES and c1RGinfo

```
Current state: ST_STABLE
sccsid = "@(#) 8273f4f 43haes/usr/sbin/cluster/hacmprd/main.C, 727, 2238D_aha727, Aug 24
2022 10:16 PM"
build = "Sep 23 2022 08:32:34 2238D_aha727"
CLversion: 22 <----- Notice it has NOT changed yet
local node vrmf is: 7270 <-----Notice this IS correct
```

```

cluster fix level is: "0"
i_local_nodeid 1, i_local_siteid 1, my_handle 2
m1_idx[1]=0      m1_idx[2]=1      m1_idx[3]=2
There are 0 events on the Ibcast queue
There are 0 events on the RM Ibcast queue
The following timers are active:

```

```
# clRGinfo
```

Group Name	Group State	Node
bdbrg	ONLINE	jessica
	OFFLINE	jordan

Important: While the cluster is in a mixed state, do *not* make any cluster changes, including Cluster Single Point of Control (C-SPOC), or synchronize the cluster.

8. On node jordan (the second and last node to be migrated), stop the cluster services by running **smitty clstop** with the Unmanage Resource Groups option. You can also stop the cluster services by running **clmgr stop node jordan manage=unmanage**. After the command completes, node jordan is listed in the Forced down list, and the RG is unmanaged, as shown in Example 5-17. However, because jordan is the standby node, it is not hosting any RGs; it might have stopped normally without the **unmanage** option.

Example 5-17 Stopping the cluster services on secondary node jordan

```

# lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#) 7d4c34b 43haes/usr/sbin/cluster/hacmprd/main.C, 61aha_r726,
2205I_aha726, Feb 05 2021 09:50 PM"
build = "Jun 21 2022 10:12:48 2205I_aha726"
CLversion: 22
local node vrmf is: 7261
cluster fix level is: "0"
i_local_nodeid 0, i_local_siteid 1, my_handle 1
m1_idx[1]=0      m1_idx[2]=1      m1_idx[3]=2
Forced down node list: jordan <-----Notice it is in the list
There are 0 events on the Ibcast queue
There are 0 events on the RM Ibcast queue

```

9. Install all PowerHA 7.2.x file sets by running **smitty update_all**. Make sure to accept the new license agreements.
10. Verify that all the cluster file sets are at the new level by running **1s1pp -l cluster.***, as shown in Example 5-1 on page 164.
11. Verify that the hallevel is correct by running **hallevel -s**, as shown in Example 5-2 on page 167.
12. Verify that no errors are reported in /tmp/clconvert.log, as shown in Example 5-18.

Example 5-18 The clconvert.log output on node jordan

```
----- log file for cl_convert: Wed Nov 2 09:15:52 CDT 2022
```

```
Command line is:
/usr/es/sbin/cluster/conversion/cl_convert -F -v 7.2.6
```

```
No source product was specified.  
Assume that the source and the target are the same product.  
Parameters read in from command line are:  
    Source Product is HAES.  
    Source Version is 7.2.6.  
    Target Product is HAES.  
    Target Version is 7.2.7.  
    Force Flag is set.  
Cleanup:  
    Writing resulting odms to /etc/es/objrepos.  
    Restoring the original ODMDIR to /etc/es/objrepos.  
    Removing the temporary directory /tmp/tmpodmdir.
```

Exiting cl_convert.

Exiting with error code 0. Completed successfully.

```
----- end of log file for cl_convert: Wed Nov 2 09:15:55 CDT 2022
```

13. Start the cluster services on the secondary node jordan by running **smitty clstart** or **clmgr start jordan**. On node startup, cluster verification is skipped, as shown in Example 5-19.

Example 5-19 Verification output skipped on last node

Verifying cluster configuration before starting cluster services

```
Cluster services are running at different levels across  
the cluster. Verification will not be invoked in this environment.  
jordan: start_cluster: Starting PowerHA SystemMirror  
..  
"jordan" is now online.
```

14. Check for the updated CLversion information from clstrmgrES information, as shown in Example 5-20.

Example 5-20 Checking the updated clstrmgrES information

```
# lssrc -ls clstrmgrES  
Current state: ST_STABLE  
sccsid = "@(#) 8273f4f 43haes/usr/sbin/cluster/hacmprd/main.C, 727, 2238D_aha727,  
Aug 24 2022 10:16 PM"  
build = "Sep 23 2022 08:32:34 2238D_aha727"  
CLversion: 23 <-----This shows migration is completed  
local node vrmf is: 7270  
cluster fix level is: "0"  
i_local_nodeid 2, i_local_siteid 2, my_handle 3  
m1_idx[1]=0      m1_idx[2]=1      m1_idx[3]=2  
There are 0 events on the Ibcast queue  
There are 0 events on the RM Ibcast queue
```

15. Verify that the cluster completed migration on both nodes, as shown in Example 5-21 on page 179.

Note: Both nodes must show CLversion: 23; otherwise, the migration did not complete successfully. Call IBM Support if necessary.

Example 5-21 Verifying the cluster versions on both nodes

```
# clcmd odmget HACMPnode|grep -i vers|uniq  
version = 23  
  
# clcmd odmget HACMPcluster|grep -i version|uniq  
cluster_version = 23  
  
# clcmd halevel -s  
-----  
NODE jordan  
-----  
7.2.7 GA  
  
-----  
NODE jessica  
-----  
7.2.7 GA
```

16. Verify that the cluster is stable and that the RG is still online, as shown in Example 5-22.

Example 5-22 Checking the stable cluster and online resource group

```
# clcmd lssrc -ls clstrmgrES|grep -i state  
Current state: ST_STABLE  
Current state: ST_STABLE  
  
# c1RGinfo  
-----  
Group Name          Group State      Node  
-----  
bdbrg              ONLINE          jessica  
                      OFFLINE         jordan
```

17. Perform cluster validation testing.

5.3.6 Migration by using **cl_ezupdate**

The **/usr/es/sbin/cluster/utilities/cl_ezupdate** command was first introduced in the first SP of PowerHA 7.2.1, and it became standard in PowerHA 7.2.2. Therefore, versions are the bare minimum PowerHA levels that can use the tool. **cl_ezupdate** provides the semi-automated capability to update the software for the entire cluster or a subset of nodes in the cluster, often in a non-disruptive fashion. Though **cl_ezupdate** has *update* in its name, it can be used for both applying SP updates and performing an upgrade. The updates can be on local file systems, a Network File System (NFS), or even a Network Installation Management (NIM) resource. Output from the **cl_ezupdate** command is captured in the **/var/hacmp/EZUpdate/EZUpdate.log** file.

Demonstration: For a demonstration of a **cl_ezupdate** upgrade, see this [YouTube video](#).

Capabilities of `c1_ezupdate`

The `c1_ezupdate` command can be used to perform the following tasks:

- ▶ Query information about the cluster, nodes, NIM server, or SPs and interim fixes that are in a specified installation location. The query can be run on the entire cluster or on a specific subset of nodes in the cluster.
- ▶ Apply and reject updates for AIX SPs or interim fixes. The `c1_ezupdate` command cannot be used to update the cluster to newer AIX TLs.
- ▶ Apply and reject updates for PowerHA SystemMirror SPs and TLs, and interim fixes in a specified installation location. This process is performed on the entire cluster or on a specific subset of nodes in the cluster. You can also apply updates in preview mode. When you use preview mode, all the prerequisites for installation process are checked, but the cluster updates are not installed on the system.
- ▶ Reject AIX SPs, PowerHA SPs, and interim fixes that were already installed on the system. This task is performed on the entire cluster or on a specific subset of cluster nodes.

The initial and most common use case is to perform a non-disruptive upgrade (NDU) across the cluster. Although it can be used for AIX updates, to use it for NDU, the AIX levels must already be at the required levels to support the planned update or upgrade. Like an NDU, `c1_ezupdate` performs the following tasks:

- ▶ If a node is hosting an RG, `c1_ezupdate` stops the cluster in the UNMANAGED state.
- ▶ If a node is *not* hosting an RG, `c1_ezupdate` gracefully stops a cluster on that node.
- ▶ Performs the update (`update_all`).
- ▶ If a node was gracefully stopped, it is restarted in Manual mode.
- ▶ If a node was stopped UNMANAGED (forced), it is restarted in Automatic mode.

Important: If you are not using the application monitoring tool within PowerHA and you stop and unmanage a node with an RG that contains an application controller, then a cluster start with the default of auto-manage for the RG invokes the application start script. This approach might lead to unwanted results.

Ideally, the application controller should be a smart start script that checks whether the app is running and exits as needed. However, you can edit the script and insert an “exit 0” at the top of the script. When the cluster stabilizes, you can remove this line.

If application monitoring is used, this issue should not happen.

Pre-update checks

Similar to when you update AIX, a preview installation is available to maximize the chance for success. During a preview, and also in an actual update, numerous checks are performed, which include the following items (and others):

- ▶ PowerHA images are supported on the AIX levels that are installed in the cluster.
- ▶ `c1cmd` communications are functional.
- ▶ The cluster, node, and RG states are checked.
- ▶ NIM server communication is functional, and a NIM resource exists and is usable.
- ▶ NFS mounting from the NIM server is tested.
- ▶ The preview compares and validates that the installed PowerHA file sets are the same on all nodes.

- ▶ The preview makes sure that no current PowerHA file sets must be Committed or Rejected.
- ▶ The preview performs a preview installation of the update package.

If you run the **c1_ezupdate** tool and an error occurs in a node during an installation or uninstallation process, you can use the rollback feature of the **c1_ezupdate** tool to return the node to the previous state. When you use the rollback feature, you can choose to roll back only the node that encountered the error or roll back all nodes that were updated.

The rollback process creates a copy of the rootvg volume group (VG) on each node by using the **alt_disk_copy** command. The rollback process restarts the copy of the rootvg VG when an error occurs during the installation or removal of service images. For the rollback process to work, one hdisk must be present on each node that can contain a copy of the rootvg VG.

Limitations of c1_ezupdate

The **c1_ezupdate** command has the following limitations:

- ▶ If you previously installed any interim fixes, those fixes might be overwritten or removed when you apply a new SP. If a previously installed interim fix locked the file set, you can override that lock and install the SP by using the **-F** flag.
- ▶ You cannot install a new PowerHA SystemMirror TL in the Applied state. File sets that are installed as part of new TL are automatically moved into the Committed state, which means that the installation image cannot be rejected. The **c1_ezupdate** tool cannot be used to uninstall TLs.
- ▶ If you want to update the software by using a NIM resource, the NIM client must be configured first and available to all nodes where you want to use the **c1_ezupdate** tool.
- ▶ The **c1_ezupdate** tool requires an existing PowerHA SystemMirror and CAA cluster definition.
- ▶ The Cluster Communications daemon (**c1cmd**) must be enabled to communicate with all nodes in the cluster. The **c1_ezupdate** tool attempts to verify **c1cmd** communications before installing any updates.
- ▶ If a cluster node update operation fails, the **c1_ezupdate** script ends immediately and exits with an error. To troubleshoot the issue, an administrator must restart the update operation or undo the completed update operations.
- ▶ Place any interim fixes in the emgr/ppc directory of the NIM 1pp_source resource.
- ▶ The **c1_ezupdate** tool runs only on AIX 7 or later.
- ▶ The **c1_ezupdate** tool *cannot* be used with the AIX **multibos** utility.
- ▶ If you are running the **c1_ezupdate** tool on a cluster node that is not included as an option of the **-N** flag and if the **-S** flag specifies the file system path as an option, the cluster node on which you are running the command is the source node for installation image propagation. This cluster node must have the file system path specified in the **-S** option.

Syntax of c1_ezupdate

Here are the flags and options that are available for the **c1_ezupdate** command in PowerHA 7.2.7. If you have a different version, consult the man page on your cluster.

- | | |
|-----------|---|
| -A | Applies the updates that are available in the location that is specified by the -S flag. |
| -C | Commits software updates to the latest installed version of PowerHA SystemMirror or the AIX operating system. |

-F	Forces installation of the SP. If an interim fix has locked a file set and the updates are stopped from installing, this flag removes the lock and installs the SP. This flag must always be used with the -A flag.
-H	Displays the help information for the c1_ezupdate command.
-I	Specifies an interactive mode. If you specify the value as yes, you must specify whether the rollback feature must continue to run when an error is shown. The interactive mode is active by default. If you specify the value as no, the interactive mode is turned off, and you are not prompted before you start the rollback operation.
-N	Specifies the node names where you want to install updates. If you specify multiple node names, you must separate each node name with a comma. By default, updates are installed on all nodes in a cluster. If the -U or -u flag is specified to enable the rollback feature, the -N flag specifies a <node name>:hdisk pair. If a node has multiple hdisks for rootvg VG, multiple -N arguments are required to map the node to each of the hdisks. For example:
	<pre>-N node1:hdisk1 N node1:hdisk2 N node1:hdisk3 N node2:hdisk1</pre>
-P	Runs the cluster installation in preview mode. When you use preview mode, all the installation prerequisites are checked, but updates are not installed on the system.
-Q	Queries the status of the NIM setup, cluster software, or available updates. The value options are cluster, node, nim, or lpp.
-R	Rejects a non-committed SP that is installed and stored in the location that is specified by the -S flag.
-S	Specifies the location of where the update image will be installed. If you specify a file system name, the path must begin with a forward slash (/). If you do not specify a forward slash, the lpp_source location of the NIM server will be used for installing updates.
-T	Specifies the timeout value for the backup operation of the rootvg VG in minutes. If the rootvg VG was not copied before the specified timeout value, the operation exits. The default value of this flag is infinite.
-U	Enables rollback of all modified nodes when an error occurs during an Apply or Reject operation.
-u	Enables rollback of only the node that encountered an error during an Apply or Reject operation.
-V	Displays extended help information.
-X	Exits after creating a copy of rootvg VG by using the alt_disk_copy command on each node. Use the -x argument to use the alternative copies of rootvg VG for the rollback operation on subsequent runs.
-x	Specifies to not create a copy of the rootvg VG by using the alt_disk_copy command on each node for rollback operation. If the rootvg VG fails, you can use disks that are specified in the -N argument for the rollback operation.

Migration example with cl_ezupdate

This section describes performing a cluster-wide migration by using the **cl_ezupdate** tool. The test environment consists of the following levels:

- ▶ Beginning:
 - AIX 7.2 TL5 SP4
 - PowerHA 7.2.5 SP3
- ▶ Ending:
 - AIX 7.2 TL5 SP4
 - PowerHA 7.2.7

In this example, the cluster is active on both nodes, and the resources are hosted on node jessica, as shown in Figure 5-1. The PowerHA updated file sets are in an NFS mount (/mnt) from the NIM server. Another option is that it might be in a defined resource to the NIM server.

```
Cluster: redbook_cluster (7253)
          00:55:06 04Nov22

jessica  iState: ST_STABLE
bdbrg      ONLINE
          en0 bdbsvc jessica
- leevg(4) -

jordan   iState: ST_STABLE
          en0 jordan
```

Figure 5-1 Beginning cluster state

Perform a preview installation to find any existing issues by running **cl_ezupdate -PS /mnt**, as shown in Example 5-23.

Example 5-23 Preview installation by running cl_ezupdate

```
# cl_ezupdate -PS /mnt
Checking for root authority...
        Running as root.
Checking for AIX level...
        The installed AIX version is supported.
Checking for PowerHA SystemMirror version.
        The installed PowerHA SystemMirror version is supported.
Checking for clcomd communication on all nodes...
        clcomd on each node can both send and receive messages.
INFO: The cluster: redbook_cluster is in state: STABLE
INFO: The node: jessica is in state: NORMAL
INFO: The node: jordan is in state: NORMAL
Checking for lpps and interim fixes from source: /mnt...
Build lists of file sets that can be apply reject or commit on node jessica
        Fileset list to apply on node jessica: cluster.es.client.clcomd
cluster.es.client.lib cluster.es.client.rte cluster.es.client.utils
cluster.es.cspoc.cmds cluster.es.cspoc.rte cluster.es.migcheck
cluster.es.server.diag cluster.es.server.events cluster.es.server.rte
cluster.es.server.testtool cluster.es.server.utils cluster.es.smui.agent
cluster.es.smui.common cluster.license cluster.man.en_US.es.data
cluster.msg.en_US.es.client cluster.msg.en_US.es.server
Before to install file sets and or interim fixes, the node: jessica will be
stopped in unmanage mode.
```

```
There is nothing to commit or reject on node: jessica from source: /mnt  
Build lists of file sets that can be apply reject or commit on node jordan  
File set list to apply on node jordan: cluster.es.client.clcomd  
cluster.es.client.lib cluster.es.client.rte cluster.es.client.utils  
cluster.es.cspoc.cmds cluster.es.cspoc.rte cluster.es.migcheck cluster.es.nfs.rte  
cluster.es.server.diag cluster.es.server.events cluster.es.server.rte  
cluster.es.server.testtool cluster.es.server.utils cluster.es.smui.agent  
cluster.es.smui.common cluster.license cluster.man.en_US.es.data  
cluster.msg.en_US.es.client cluster.msg.en_US.es.server
```

```
Before to install file sets and or interim fixes, the node: jordan will be  
stopped in unmanage mode.
```

```
There is nothing to commit or reject on node: jordan from source: /mnt  
Installing file set updates in preview mode on node: jessica...  
Succeeded to install preview updates on node: jessica.  
Installing file set updates in preview mode on node: jordan...  
Succeeded to install preview updates on node: jordan.
```

After you review the output, and if no errors are reported, you can initiate the migration by running **cl_ezupdate -AS /mnt**. This command performs all the same checks as the preview installation did, however, these checks were omitted from the output in Example 5-24 for clarity.

Node jessica is stopped, unmanaged, upgraded, and restarted. Then, the node jordan is stopped fully, not unmanaged, upgraded, and restarted. When the cluster stabilizes, the cluster migration is complete. In this example, the entire process is approximately 10 minutes.

Example 5-24 The cl_ezupdate migration output

```
Stopping the node jessica...  
Stopping PowerHA cluster services on node: jessica in unmanage mode...  
  
INFO: As TIMEOUT is not specified, clmgr cognitive mechanism has predicted  
TIMEOUT=120.  
  
jessica: 0513-044 The clevmgrdES Subsystem was requested to stop.  
"jessica" is now unmanaged.  
  
jessica: Nov  4 2022 01:44:48/usr/es/sbin/cluster/utilities/clstop: called with  
flags -N -f  
Applying updates on node: jessica...  
Succeeded to apply updates on node: jessica.  
Starting the node: jessica...  
Starting PowerHA cluster services on node: jessica in auto mode...  
Verifying cluster configuration before starting cluster services  
  
Cluster services are running at different levels across  
the cluster. Verification will not be invoked in this environment.  
jessica: start_cluster: Starting PowerHA SystemMirror  
jessica: rc.cluster: Successfully ran varyonvg -a command for volume group leevg.  
...  
"jessica" is now online.  
  
Starting Cluster Services on node: jessica  
This may take a few minutes. Please wait...  
jessica: Nov  4 2022 01:47:49Starting execution of  
/usr/es/sbin/cluster/etc/rc.cluster
```

```

jessica: with parameters: -boot -N -b -P cl_rc_cluster -A
jessica:
jessica: Nov  4 2022 01:47:49Checking for srcmstr active...
jessica: Nov  4 2022 01:47:49complete.
Stopping the node jordan...
Stopping PowerHA cluster services on node: jordan in offline mode...

INFO: As TIMEOUT is not specified, clmgr cognitive mechanism has predicted
      TIMEOUT=170.

Broadcast message from root@jordan (tty) at 01:48:33 ...

PowerHA SystemMirror on jordan shutting down. Please exit any cluster
applications...
jordan: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.
jordan: 0513-044 The clevmgrdES Subsystem was requested to stop.
"jordan" is now offline.

jordan: Nov  4 2022 01:48:33/usr/es/sbin/cluster/utilities/clstop: called with
flags -N -g
Applying updates on node: jordan...
Succeeded to apply updates on node: jordan.
Starting the node: jordan...
Starting PowerHA cluster services on node: jordan in manual mode...
Verifying cluster configuration before starting cluster services

Cluster services are running at different levels across
the cluster. Verification will not be invoked in this environment.
jordan: start_cluster: Starting PowerHA SystemMirror
jordan: 4915652      - 0:00 syslogd
jordan: Setting routerevalidate to 1

Broadcast message from root@jordan (tty) at 01:51:21 ...

Starting Event Manager (clevmgrdES) subsystem on jordan

jordan: 0513-059 The clevmgrdES Subsystem has been started. Subsystem PID is
16908768.
jordan: PowerHA: Cluster services started on Fri Nov  4 01:51:21 CDT 2022
jordan: event serial number 22791
..
"jordan" is now online.

Starting Cluster Services on node: jordan
This may take a few minutes. Please wait...
jordan: Nov  4 2022 01:51:18Starting execution of
/usr/es/sbin/cluster/etc/rc.cluster
jordan: with parameters: -boot -N -b -P cl_rc_cluster -M
jordan:
jordan: Nov  4 2022 01:51:18Checking for srcmstr active...
jordan: Nov  4 2022 01:51:18complete.
jordan: Nov  4 2022 01:51:20/usr/es/sbin/cluster/utilities/clstart: called with
flags -m -G -b -P cl_rc_cluster -B -M
jordan: Nov  4 2022 01:51:21Completed execution of
/usr/es/sbin/cluster/etc/rc.cluster

```

```
jordan: with parameters: -boot -N -b -P cl_rc_cluster -M.  
jordan: Exit status = 0  
jordan:
```

When the process completes, perform the same cluster migration level verification checks as with any other migration type:

1. Verify that all cluster file sets are at the new level by running `1s1pp -l cluster.*`, as shown in Example 5-1 on page 164.
2. Verify that the `halevel` is correct by running `halevel -s`, as shown in Example 5-2 on page 167.
3. Verify that no errors are reported in `/tmp/clconvert.log`, as shown in Example 5-18 on page 177.
4. Verify that the cluster completed the migration on both nodes, as shown in Example 5-21 on page 179.
5. Verify that the cluster is stable and the RG is still online, as shown in Example 5-22 on page 179.
6. Perform cluster validation testing.

5.4 Other migration options

This section offers some more options for performing a migration. Although they are known to have worked, do not assume that they are officially supported. As is the case with any migration method, it should be thoroughly tested before performing it in production.

5.4.1 Using `alt_disk_copy`

This method can be used as either a rolling or offline migration. The main difference is that instead of updating the live running environment, you can use an alternative rootvg to upgrade both AIX and PowerHA levels to the target levels. You can use this method when updating AIX, but not when you upgrade to major versions. For more information about major AIX version upgrades, see 5.4.2, “Using nimadm” on page 188.

Important: Although this procedure can work, it is *as-is*. It is not an officially supported method for performing a PowerHA upgrade, so if you have any problems, do not expect any help from IBM Support.

One key benefit in using this method is that the total downtime that is involved in performing the upgrade is reduced because most of the work and time that is involved is performed before you stop any node. This method requires an extra available disk that is large enough to accommodate rootvg and the extra updates to be installed.

This migration scenario closely mimics the steps of an offline migration. The test environment consists of the following levels:

- ▶ Beginning:
 - AIX 7.1 TL5 SP1
 - PowerHA 7.2.4 SP5
- ▶ Ending:
 - AIX 7.1 TL5 SP6
 - PowerHA 7.2.7

Complete the following steps:

1. Create a cloned rootvg and apply AIX updates. All updates are in an NFS mount of /mnt/aixupdates. This task can be performed while the cluster is still active. In this scenario, we ran the commands that are shown in Example 5-25.

Example 5-25 Upgrading by cloning rootvg and alt_disk_copy

```
jordan# alt_disk_copy -b update_all -l /mnt/aixupdates -d hdisk9
```

```
jessica# alt_disk_copy -b update_all -l /mnt/aixupdates -d hdisk5
```

2. Validate that the bootlist is now set to the newly created cloned and updated alternative rootvg, as shown in Example 5-26.

Example 5-26 Bootlist verification

```
jordan# bootlist -om normal
hdisk9 b1v=hd5 pathid=0
hdisk9 b1v=hd5 pathid=1
hdisk9 b1v=hd5 pathid=2
hdisk9 b1v=hd5 pathid=3
```

```
jessica# bootlist -om normal
hdisk5 b1v=hd5 pathid=0
hdisk5 b1v=hd5 pathid=1
hdisk5 b1v=hd5 pathid=2
hdisk5 b1v=hd5 pathid=3
```

3. Stop the cluster services on all nodes by running **smitty clstop**, and bring the RGs offline. You can also do this step by running **clmgr stop cluster**.
4. Restart from the newly cloned and updated rootvg disk by running **shutdown -Fr**.
5. Verify that caavg_private is active by running **1spv** or **lscluster -i**.
6. Verify that **c1cmd** is active by running **1ssrc -s c1cmd**. If it is not active, activate it by running **startsrc -s c1cmd**.
7. Install all PowerHA 7.2.x file sets by running **smitty update_all**. Make sure to accept the new license agreements.
8. Verify that all the cluster file sets are at the new level by running **1s1pp -l cluster.***, as shown in Example 5-1 on page 164.
9. Verify that the halevel is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
10. Verify that no errors are reported in /tmp/c1convert.log, as shown in Example 5-3 on page 167.
11. Restart the cluster services by running **smitty clstart** or **clmgr start cluster**.
12. Verify that the cluster completed migration on both nodes, as shown in Example 5-21 on page 179.
13. Verify that the cluster is stable and that the RG is still online, as shown in Example 5-22 on page 179.
14. Perform cluster validation testing.

5.4.2 Using nimadm

Upgrading by using **nimadm** is similar to using the **alt_disk_copy** method that is described in 5.4.1, “Using alt_disk_copy” on page 186. The existing running environment is used for recovery purposes because all upgrades and updates are performed on a cloned copy. The key difference is that **nimadm** can be used to also perform an AIX upgrade instead of an update. It also requires a NIM server to be configured with a proper lppsource and an available disk to accommodate the copied rootvg from each node.

Important: Although this procedure can work, it is *as-is*. It is not an officially supported method for performing a PowerHA upgrade, so if you have any problems, do not expect any help from IBM Support.

This migration scenario closely mimics the steps of a rolling migration. The test environment consists of the following levels:

- ▶ Beginning:
 - AIX 7.1 TL5 SP1
 - PowerHA 7.2.4 SP5
- ▶ Ending:
 - AIX 7.3 TL0 SP2
 - PowerHA 7.2.7

Complete the following steps:

1. Perform AIX upgrades by running **nimadm**. These upgrades can be performed while the cluster is still active. In this scenario, we ran the command that is shown in Example 5-27.

Example 5-27 Upgrading by using nimadm on standby node jordan

```
nimsrv# nimadm -j nimadmv -c jordan -s AIX7302spot -l AIX7302lpp -d hdisk9 -Y
```

2. Validate that the bootlist is set to the newly created cloned and upgraded alternative rootvg, as shown in Example 5-28.

Example 5-28 Bootlist verification on standby node jordan

```
jordan# bootlist -om normal
hdisk9 b1v=hd5 pathid=0
hdisk9 b1v=hd5 pathid=1
hdisk9 b1v=hd5 pathid=2
hdisk9 b1v=hd5 pathid=3
```

3. Stop the cluster services on the first node, which in this scenario is the standby node **jordan**, by running **smitty clstop**, and choose to bring the RGs offline. You can also do this task by running **clmgr stop node jordan**.
4. Restart from the newly cloned and upgraded rootvg disk by running **shutdown -Fr**.
5. Verify that **caavg_private** is active by running **1spv** or **1scluster -i**.
6. Verify that **c1comd** is active by running **1ssrc -s c1comd**. If it is not active, activate it by running **startsrc -s c1comd**.
7. Install all the PowerHA 7.2.x file sets by running **smitty update_all**. Make sure to accept the new license agreements.
8. Verify that all the cluster file sets are at the new level by running **1s1pp -l cluster.***, as shown in Example 5-1 on page 164.

9. Verify that halevel is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
10. Verify that no errors are reported in /tmp/c1convert.log, as shown in Example 5-3 on page 167.
11. Restart the cluster services by running **smitty clstart** or **clmgr start node jordan**.
12. Make sure that the cluster stabilizes.
13. If they were not already performed in parallel with the first node, perform AIX upgrades by running **nimadm** on the second node, which in this scenario is the primary node jessica. This process can be performed while the cluster is still active. In this scenario, we ran the command that is shown in Example 5-29.

Example 5-29 Upgrading by running nimadm on the primary node jessica

```
nimsvr# nimadm -j nimadmv -c jessica -s AIX7302spot -l AIX73021pp -d hdisk5 -Y
```

14. Validate that the bootlist is now set to the newly created cloned and upgraded alternative rootvg, as shown in Example 5-30.

Example 5-30 Bootlist verification on primary node jessica

```
jessica# bootlist -om normal
hdisk5 b1v=hd5 pathid=0
hdisk5 b1v=hd5 pathid=1
hdisk5 b1v=hd5 pathid=2
hdisk5 b1v=hd5 pathid=3
```

15. On the second node jessica, stop the cluster services by using the Move Resource Groups option (**smitty clstop**). This command results in the resource becoming active on node jordan, as shown in Example 5-5 on page 168.
16. Restart from the newly cloned and upgraded rootvg disk by running **shutdown -Fr**.
17. Verify that caavg_private is active by running **1spv** or **1scluster -i**.
18. Verify that **c1cmd** is active by running **1ssrc -s c1cmd**. If it is not active, activate it by running **startsrc -s c1cmd**.
19. Install all the PowerHA 7.2.x file sets by running **smitty update_all**. Make sure to accept the new license agreements.
20. Verify that all the cluster file sets are at the new level by running **1s1pp -l cluster.***, as shown in Example 5-1 on page 164.
21. Verify that the halevel is correct by running **halevel -s**, as shown in Example 5-2 on page 167.
22. Verify that no errors are reported in /tmp/c1convert.log, as shown in Example 5-3 on page 167.
23. Restart the cluster services by running **smitty clstart** or **clmgr start node jessica**.
24. Verify that the cluster completed migration on both nodes, as shown in Example 5-21 on page 179.
25. Verify that the cluster is stable and the RG is still online, as shown in Example 5-22 on page 179.
26. Perform cluster validation testing.

5.4.3 Live Update

Starting with AIX 7.2, the AIX operating system provides the AIX Live Update (formerly Live Kernel Update) function. This capability eliminates the workload downtime that is associated with the AIX system restart that was required by previous AIX releases when fixes to the AIX kernel were deployed. The workloads on the system are not stopped in a Live Update operation, but the workloads can use the interim fixes after the Live Update operation. For more information, see [Live Update](#).

Important: Although this procedure can work, it is *as-is*. It is not an officially supported method for performing a PowerHA upgrade, so if you have any problems, do not expect any help from IBM Support.

In this scenario, a Live Update of AIX is performed on each node, and then either a non-disruptive update or `c1_ezupdate` is performed to complete the PowerHA migration.

Both the AIX and PowerHA levels must have Live Update support. PowerHA added integrated support for Live Update in Version 7.2.0. However, even though by default Live Update is enabled on new installations, it might not be on upgrades, so it should be verified that it is enabled before using it.

Demonstration: For a demonstration of Live Update of a PowerHA node, see this [YouTube video](#).

Complete the following steps:

1. Make sure that you have a backup of the environment to be upgraded.
2. Verify that Live Update support is enabled on both or all cluster nodes, as shown in Example 5-31.

Example 5-31 Checking that Live Update is enabled on all cluster nodes

```
# clmgr -a ENABLE_LIVE_UPDATE view node jordan  
ENABLE_LIVE_UPDATE="false"  
  
# clmgr -a ENABLE_LIVE_UPDATE view node jessica  
ENABLE_LIVE_UPDATE="false"
```

If Live Update is not enabled, enable it by setting the value to true by running the following command:

```
clmgr modify node jordan ENABLE_LIVE_UPDATE=true
```

Repeat this command for each node as necessary and synchronize the cluster.

3. Pick any node, but *only* one node, and perform a Live Update.
During the live update, the node becomes unmanaged immediately before the upgrade, and then becomes auto-managed post upgrade. This process is normal, and is the function that PowerHA provides during a Live Update operation.
4. On a successful AIX upgrade, verify that `caavg_private` is active by running `1spv` or `1scluster -i`.
5. Verify that `c1cmd` is active by running `1ssrc -s c1cmd`. If it is not active, activate it by running `startsrc -s c1cmd`.
6. Repeat steps 3 - 5 as needed for each node.

7. On a successful AIX upgrade of all nodes, perform the process that is described in 5.3.6, “Migration by using cl_ezupdate” on page 179 for PowerHA.
8. Verify that all the cluster file sets are at the new level by running `lslpp -l cluster.*`, as shown in Example 5-1 on page 164.
9. Verify that the `halevel` is correct by running `halevel -s`, as shown in Example 5-2 on page 167.
10. Verify that no errors are reported in `/tmp/clconvert.log`, as shown in Example 5-3 on page 167.
11. Verify that the cluster completed migration on both nodes, as shown in Example 5-21 on page 179.
12. Verify that the cluster is stable and that the RG is still online as shown in Example 5-22 on page 179.
13. Perform cluster validation testing.

5.5 Common migration errors

You can correct some common migration problems that you might encounter.

5.5.1 Stuck in migration

When migration is completed, you might not progress to the update of the ODM entries until the `node_up` event is run on the last node of the cluster. If you have this problem, start the node to see whether this action completes the migration protocol and updates the version numbers correctly. For PowerHA 7.2.7, the version number must be 23 in the `HACMPcluster` class. You can verify this number by running the `odmget` command, as shown in Example 5-32. For PowerHA 7.2.7, if the version number is less than 23, you are still stuck in migration.

Note: Usually a complete stop, sync, verify, and restart of the cluster completes the migration, but a sync might not be possible. You may modify the ODM manually, but the preferred action is to contact IBM Support.

Example 5-32 Running odmget to check the version

```
# odmget HACMPcluster|grep version
    cluster_version = 22
```



Part 3

Cluster administration

In this part, we present cluster administrative tasks and scenarios for modifying and maintaining a PowerHA cluster.

This part contains the following chapters:

- ▶ Chapter 6, “Cluster maintenance” on page 195
- ▶ Chapter 7, “Cluster management” on page 251
- ▶ Chapter 8, “Cluster security” on page 347



Cluster maintenance

This chapter provides basic guidelines to follow while you are planning and performing maintenance operations in a PowerHA cluster. The goal is to keep the cluster applications as highly available as possible. We use functions within PowerHA and AIX to perform these operations. These scenarios are not exhaustive.

In this chapter, AIX best practices for troubleshooting, including monitoring the error log, are assumed. However, we do not cover how to determine what problem exists, whether dealing with problems either after they are discovered or as preventive maintenance.

This chapter contains the following topics:

- ▶ Change control and testing
- ▶ Starting and stopping the cluster
- ▶ Resource group and application management
- ▶ Scenarios
- ▶ Updating multipath drivers
- ▶ Repository disk replacement
- ▶ Critical volume groups
- ▶ Cluster Test Tool

6.1 Change control and testing

Change control is imperative to provide high availability (HA) in any system, but it is more crucial in a clustered environment. One of the top and avoidable issues with things not working after a failover is making changes on a single system without making those changes across the cluster.

6.1.1 Scope

Change control is not within the scope of the documented procedures in this book. It encompasses several aspects and is *not* optional. Change control includes, but is not limited to, these items:

- ▶ Limited root access
- ▶ Thoroughly documented and *tested* procedures
- ▶ Proper planning and approval of all changes

6.1.2 Test cluster

A *test cluster* is important both for maintaining proper change control and for the overall success of the production cluster. Test clusters allow thorough testing of administrative and maintenance procedures to find problems before the problem reaches the production cluster. Test clusters should not be considered a luxury, but a *must-have*.

Although many PowerHA customers have a test cluster, or at least begin with a test cluster, over time these cluster nodes become used within the company in some form. Using these systems requires a scheduled maintenance window much like the production cluster. If that is the case, do not be fooled because it truly is not a test cluster.

A test cluster, ideally, is at least the same AIX, PowerHA, and application level as the production cluster. The hardware should also be as similar as possible. Usually, fully mirroring the production environment is not practical, especially when there are multiple production clusters. Several approaches exist to maximize a test cluster when multiple clusters have varying levels of software.

Using logical partitioning, Virtual I/O Servers (VIOSs), and multiple various rootvg images through `alt_disk_install` or `multibos` are a common practice. Virtualization allows a test cluster to be created with few physical resources, and can even be within the same physical machine. With the multi-boot option, you can easily change cluster environments by starting the partition from another image. This scenario also allows testing of many software procedures, such as the following ones:

- ▶ Applying AIX maintenance
- ▶ Applying PowerHA fixes
- ▶ Applying application maintenance

This type of test cluster requires at least one disk per image per logical partition (LPAR). For example, if the test cluster has two nodes and three different rootvg images, it requires a minimum of six hard disks. This approach is still more efficient than having six separate nodes in three separate test clusters.

A test cluster also allows testing of hardware maintenance procedures. These procedures include, but are not limited to, the following updates and replacement:

- ▶ System firmware updates
- ▶ Adapter firmware updates
- ▶ Adapter replacement
- ▶ Disk replacement

More testing can be accomplished by using the Cluster Test Tool and error log emulation. For more information, see 6.8, “Cluster Test Tool” on page 226.

6.2 Starting and stopping the cluster

Starting the cluster services refers to the process of starting the Reliable Scalable Cluster Technology (RSCT) subsystems that are required by PowerHA, and then starting the PowerHA daemons that enable the coordination that is required between nodes in a cluster. During startup, the cluster manager runs the **node_up event**, and resource groups (RGs) are acquired. Stopping the cluster services refers to stopping the same daemons on a node, which might cause extra PowerHA scripts to run, depending on the type of shutdown that you perform.

After PowerHA is installed, the cluster manager process (**c1strmgrES**) is always running, regardless of whether the cluster is online. It can be in one of the following states (as displayed by running the **lssrc -ls c1strmgrES** command):

NOT_CONFIGURED	The cluster is not configured or the node is not synchronized.
ST_INIT	The cluster is configured but not active on this node.
ST_STABLE	The cluster services are running with resources online.
ST_JOINING	The cluster node is joining the cluster.
ST_VOTING	The cluster nodes are voting to decide event execution.
ST_RP_RUNNING	The cluster is running a recovery program.
RP_FAILED	A recovery program event script failed.
ST_BARRIER	The c1strmgr process is between events, waiting at the barrier.
ST_CBARRIER	The c1strmgr process is exiting a recovery program.
ST_UNSTABLE	The cluster is unstable, usually due to an event error.

Changes in the state of the cluster are referred to as *cluster events*. The Cluster Manager monitors local hardware and software subsystems on each node for events, such as an *application failure* event. In response to such events, the Cluster Manager runs one or more event scripts, such as a *restart application* script. Cluster Managers running on all nodes exchange messages to coordinate the required actions in response to an event.

During maintenance periods, you might need to stop and start cluster services. But before you do this action, make sure that you understand the node interactions that it causes and the impact on your system’s availability. The cluster must be synchronized and verification should detect no errors.

Section 6.2.1, “Cluster services” on page 198 briefly describes the processes themselves and then the processing that is involved in starting or shutting down these services. In 6.2.2, “Starting cluster services” on page 198, we describe the procedures that are necessary to start cluster services on a node. For shutting down services, see 6.2.3, “Stopping cluster services” on page 201.

6.2.1 Cluster services

The main PowerHA, RSCT, and Cluster Aware AIX (CAA) daemons are as follows:

- ▶ Cluster Manager daemon (**c1strmgrES**)

This daemon is the main PowerHA one. It maintains a global view of the cluster topology and resources and runs event scripts in response to changes in the state of nodes, interfaces, or resources (or when the user makes a request).

The Cluster Manager receives information about the state of interfaces from Topology Services. The Cluster Manager maintains updated information about the location, and the status of all RGs. The Cluster Manager is a client of group services, and uses the latter for reliable inter-daemon communication.

- ▶ Cluster Communications daemon (**c1cmd**)

This daemon, which is part of CAA, provides secure communication between cluster nodes for all cluster utilities, such as verification and synchronization, and system management (Cluster Single Point of Control (C-SPOC)). The **c1cmd** daemon, like **c1strmgrES**, is started automatically at start time.

- ▶ Cluster Information Program (**c1infoES**)

This daemon provides status information about the cluster-to-cluster nodes and clients and calls the `/usr/es/sbin/cluster/etc/c1info.rc` script in response to a cluster event. The **c1info** daemon is optional on cluster nodes and clients.

- ▶ Cluster Group Services Subsystem (**cthagsd**)

This RSCT subsystem provides the reliable communication and protocols that are required for cluster operation. Clients are distributed daemons, such as the PowerHA Cluster Manager and the Enhanced Concurrent Logical Volume Manager. All cluster nodes must run the **cthagsd** daemon.

- ▶ Cluster configuration daemon (**c1confd**)

This daemon keeps CAA cluster configuration information in sync. It wakes up every 10 minutes to synchronize any necessary cluster changes.

- ▶ Cluster Event Management daemon (**clevmgrdES**)

This daemon is a primary “go-between” for the event reports, such as ahafs, snmpd, and AIX error report, that categorizes the reports into specific event categories and passes it up to the cluster manager.

- ▶ Resource Monitoring and Control (RMC) subsystem (**rmcd**)

This RSCT subsystem acts as a resource monitor for the event management subsystem and provides information about the operating system characteristics and utilization. The RMC subsystem must be running on each node in the cluster. By default, the **rmcd** daemon is set up to start from **inittab** when it is installed. The **rc.cluster** script ensures that the RMC subsystem is running.

6.2.2 Starting cluster services

In this section, we describe the startup options that are available for starting cluster services on any single node, multiple nodes, or even all nodes. As is the case with most operations, these options can be performed from the CLI, SystemMirror User Interface (SMUI), or System Management Interface Tool (SMIT).

Starting cluster services by using SMIT

As the root user, complete the following steps to start the cluster services on a node:

1. Run the SMIT fast path **smitty c1start** and press Enter. The Start Cluster Services panel opens, as shown in Figure 6-1.

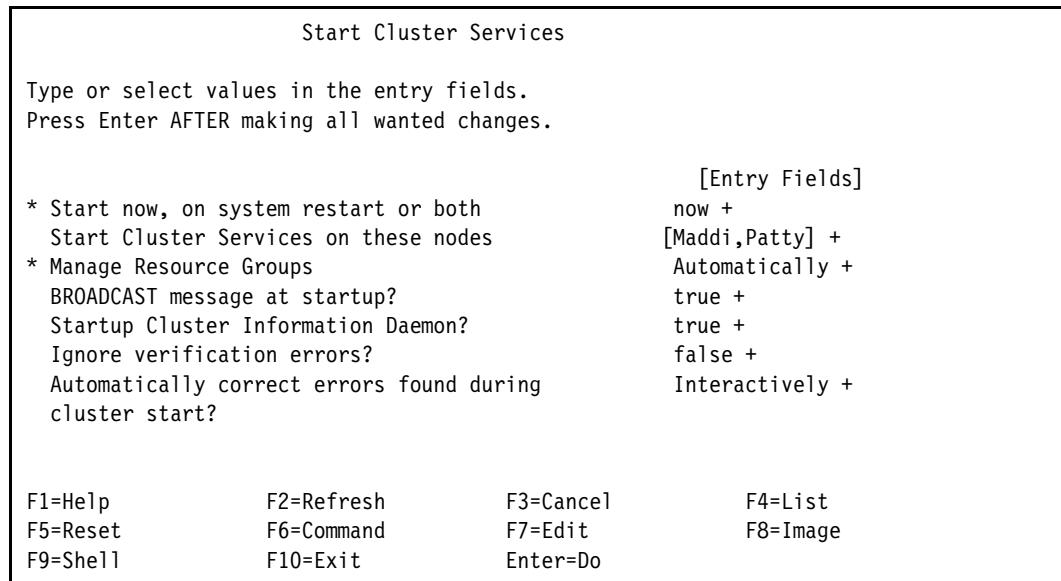


Figure 6-1 Start Cluster Services menu

2. Enter the field values as follows:

- Start now, on system restart, or both

Indicate whether you want to start cluster services and the **c1infoES** when you commit the values on this panel by pressing Enter (now), when the operating system restarts (on system restart), or on both occasions.

- Start Cluster Services on these nodes

Enter the name of one or more nodes on which you want to start cluster services. Alternatively, you can select nodes from a list. When entering multiple nodes manually, separate the names with a comma, as shown in Figure 6-1.

- Manage Resource Groups

Use one of the following options:

- Automatically, which is the default, brings RGs online according to the configuration settings of the RGs and the current cluster state, and then starts managing the RGs and applications for availability.
- Manually does not activate RGs while the cluster services on the selected node are started. After you start cluster services, you can bring any RGs online or offline, as needed.

- BROADCAST message at startup?

Indicate whether you want to send a broadcast message to all nodes when the cluster services start.

- Startup Cluster Information Daemon?

Indicate whether you want to start the **c1info** daemon. If your application uses **c1info**, and if you use the **c1stat** monitor or you want to run event emulation, set this field to true. Otherwise, set it to false.

Note: In a production environment, having PowerHA services start automatically on system restart is *not* considered a best practice.

The reason is directly related to what happens after system failure. If the RG owner system crashes, and AIX is set to restart after crash, it can restart cluster services in the middle of a current takeover. Depending on the cluster configuration, this restart might cause RG contention, RG processing errors, or even a fallback to occur, all of which can extend an outage.

However, during test and maintenance periods, and even on dedicated standby nodes, using this option might be convenient.

- Ignore Verification Errors?

Set this value to true only if the verification reports an error and this error does not put the overall cluster functions at risk. Set this value to false to stop all selected nodes from starting cluster services if the verification finds errors on any node.

- Automatically correct errors found during cluster start?

The options are Yes, No, and Interactively. This option is also known as *auto corrective actions*. Choosing Yes corrects errors automatically without prompting. Choosing No does not correct them, and prevents the cluster services from starting if errors are encountered. The Interactively option prompts you during startup about what errors are found and requests a reply to fix or not to fix.

Note: There are situations when choosing Interactively corrects some errors. For more information, see 7.6.6, “Running automatic corrective actions during verification” on page 310.

After you complete the fields and press Enter, the system starts the cluster services on the nodes that are specified, and activates the cluster configuration that you defined. The time that it takes the commands and scripts to run depends on your configuration (that is, the number of disks, the number of interfaces to configure, the number of file systems to mount, and the number of applications being started).

During the node_up event, the RGs are acquired. The time that it takes to run each node_up event depends on the resource processing during the event. The node_up events for the joining nodes are processed sequentially.

When the command completes and the PowerHA cluster services are started on all the specified nodes, SMIT displays a command status window. When the SMIT panel indicates the completion of the cluster startup, event processing usually has not yet completed. To verify that the nodes are running, run `c1stat` or even `tail` on the hacmp.out file on any node. For more information, see 7.7.1, “Cluster status checking utilities” on page 313.

Starting cluster services by using SMUI

For a demonstration about how to start a cluster by using the PowerHA SMUI, see this [YouTube video](#).

Starting cluster services by using clmgr

The `c1mgr` command can be used to start the cluster from the CLI, as shown in Example 6-1 on page 201. This base command inherits the settings from the SMIT panel unless the attributes are explicitly stated.

Example 6-1 Starting the cluster by running clmgr

```
# clmgr start cluster

Warning: "WHEN" must be specified. Since it was not, a default of "now" will be
used.

Warning: "MANAGE" must be specified. Since it was not, a default of "auto" will
be used.

Verifying cluster configuration before starting cluster services
Starting Event Manager (clevmgrdES) subsystem on jordan

jordan: 0513-059 The clevmgrdES Subsystem has been started. Subsystem PID is
11338064.
jordan: PowerHA: Cluster services started on Sat Nov 26 14:15:26 CST 2022
jordan: event serial number 25911

The cluster is now online.
```

6.2.3 Stopping cluster services

In this section, we describe the startup options of cluster services on any single node, multiple nodes, or even all nodes. As is the case with most operations, this task can be performed from the CLI, SMUI, or SMIT.

Stopping cluster services by using SMIT

The following steps describe the procedure for stopping cluster services on a single node, multiple nodes, or all nodes in a cluster by using the C-SPOC utility on one of the cluster nodes. C-SPOC stops the nodes sequentially, not in parallel. If any node that is selected to be stopped is inactive, the shutdown operation stops on that node. As with starting services, SMIT must always be used to stop cluster services. The SMIT panel to stop cluster services is shown in Figure 6-2.

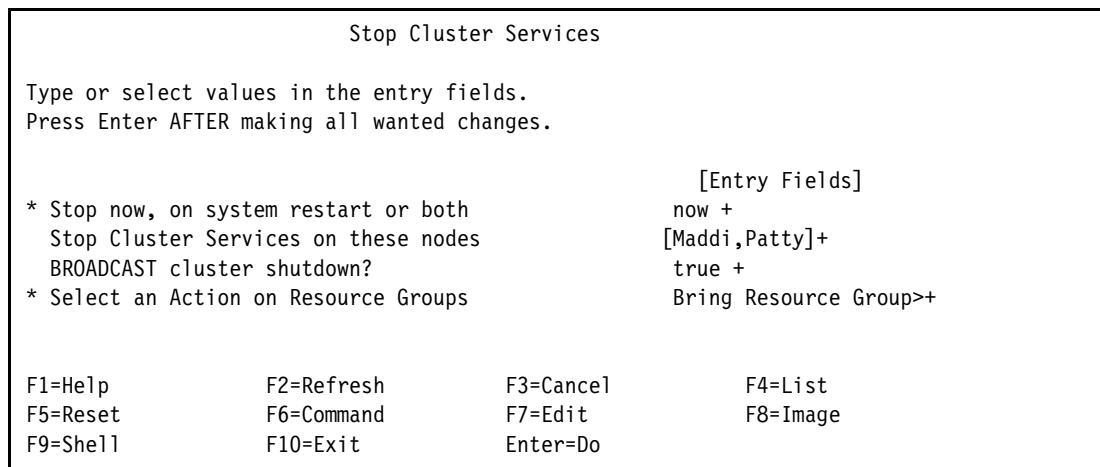


Figure 6-2 Stop Cluster Services menu

To stop cluster services, complete these steps:

1. Enter the fast path **smitty clstop** and press Enter.
2. Enter the field values in the SMIT panel as follows:

- Stop now, on system restart, or both

Indicate whether you want the cluster services to stop now, at restart (when the operating system restarts), or on both occasions. If you select restart or both, the entry in the /etc/inittab file that starts cluster services is removed. Cluster services will no longer display automatically after a restart.

- BROADCAST cluster shutdown?

Indicate whether you want to send a broadcast message to users before the cluster services stop. If you specify true, a message is broadcast on all cluster nodes.

- Shutdown mode (Select an Action on Resource Groups)

Indicate the type of shutdown:

- Bring Resource Group Offline

Stops PowerHA and releases the RGs running on the node (if any). Other cluster nodes do not take over the resources of the stopped node. In previous versions, this type of shutdown was known as *graceful*.

- Move Resource Groups

Stops PowerHA and releases the RGs that are present on the node. The next priority node takes over the resources of the stopped node. In previous versions, this type of shutdown was known as *graceful with takeover*.

- Unmanage Resource Groups

PowerHA stops on the node immediately. The node retains control of all its resources. You can use this option to bring down a node while you perform maintenance. This option is a newer one that is similar to the older **forced** option. However, because cluster services are stopped, the applications are no longer highly available. If a failure occurs, no recovery for them will be provided. This feature is used when performing nondisruptive updates and upgrades to PowerHA.

3. Press Enter to stop the cluster.

Stopping cluster services by using the SMUI

For a demonstration about how to stop a cluster by using the PowerHA SMUI, see this [YouTube video](#).

Stopping cluster services by using clmgr

The **clmgr** command can be used to stop the cluster from the CLI, as shown in Example 6-2. This base command inherits the settings from the SMIT panel unless the attributes are explicitly stated.

Example 6-2 Stopping cluster services by using clmgr

```
# clmgr stop cluster
```

Warning: "WHEN" must be specified. Since it was not, a default of "now" will be used.

Warning: "MANAGE" must be specified. Since it was not, a default of "offline" will be used.

```
INFO: As TIMEOUT is not specified, clmgr cognitive mechanism has predicted  
TIMEOUT=120.
```

```
PowerHA SystemMirror on jordan shutting down. Please exit any cluster  
applications...  
jordan: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.  
jordan: 0513-044 The clevmgrdES Subsystem was requested to stop.  
...
```

The cluster is now offline.

6.3 Resource group and application management

This section describes how to do the following actions:

- ▶ Bring an RG offline.
- ▶ Bring an RG online.
- ▶ Move an RG to another node or site.
- ▶ Suspend and resume application monitoring.

Understanding each of these actions is important, along with stopping and starting cluster services because they are often used during maintenance periods.

In the following topics, we assume that cluster services are running, the RGs are online, the applications are running, and the cluster is stable. If the cluster is not in the stable state, then the operations that are related to RG are not possible.

All three RG options that we describe can be done by using the **c1RGmove** command. However, in our examples, we use C-SPOC. They also all have similar SMIT panels and lists. To streamline this documentation, we show only one SMIT panel in each of the following sections.

6.3.1 Bringing a resource group offline

In this section, we describe the options to bring an RG offline. As is the case with most operations, this operation can be performed from the CLI, SMUI, or SMIT.

Bringing an RG offline by using SMIT

To bring an RG offline, complete the following steps:

1. Run `smitty cspoc` and then select **Resource Group and Applications** → **Bring a Resource Group Offline**.

The list is displayed, as shown in Figure 6-3. It lists only the RGs that are online or in the ERROR state on all nodes in the cluster.

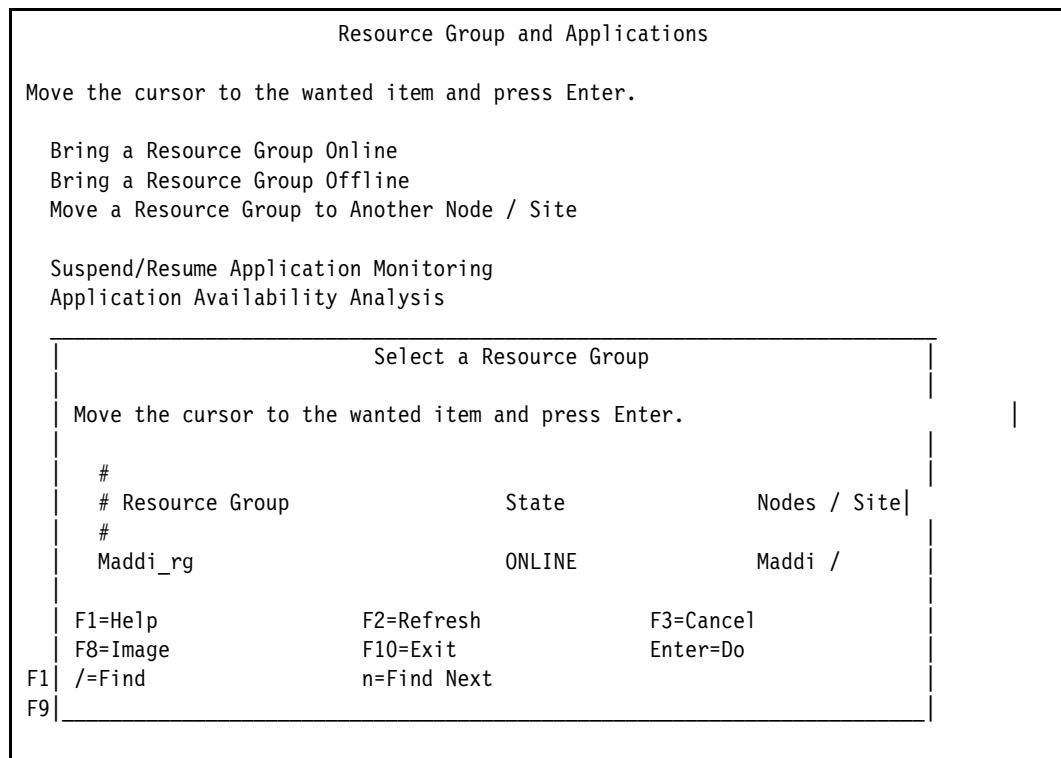


Figure 6-3 Resource Group list

2. Select an RG from the list and press Enter. Another list is displayed (Select an Online Node). The list contains only the nodes that are active in the cluster and that currently are hosting the previously selected RG.
3. Select an online node from the list and press Enter.
4. The final SMIT menu opens with the information that was selected in the previous lists, as shown in Figure 6-4. Verify the entries that you previously specified and then press Enter to start the processing of the RG to be brought offline.

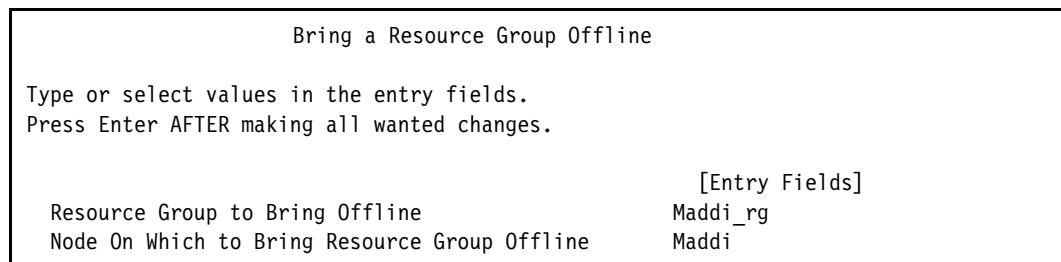


Figure 6-4 Bring a Resource Group Offline

After processing is complete, the RG is offline, but cluster services remain active on the node. The standby node will not acquire the RG.

This option is also available by using either the **c1RGinfo** or **c1mgr** command. For more information about these commands, see the [man pages](#).

Bringing an RG offline by using SMUI

For a demonstration about how to bring an RG offline by using the PowerHA SMUI, see this [YouTube video](#).

Bringing an RG offline by using clmgr

The **c1mgr** command can be used to bring an RG offline from the CLI, as shown in Example 6-3.

Example 6-3 Bringing a resource group offline by using clmgr

```
# clmgr offline rg redbookrg
Attempting to bring group redbookrg offline on node jessica.

Waiting for the cluster to stabilize.....

Resource group movement is successful.
Resource group redbookrg is offline on node jessica.

Cluster Name: jessica_cluster

Resource Group Name: redbookrg
Primary instances:
The instance is temporarily offline upon user requested rg_move performed on
Sat Nov 26 15:08:02 2022

Node                               Group State
-----
jessica                            OFFLINE
jordan                             OFFLINE
```

6.3.2 Bringing a resource group online

In this section, we describe the options to bring an RG online. As is the case with most operations, this operation can be performed from the CLI, SMUI, or SMIT.

Bringing a resource group online by using SMIT

To bring an RG online, complete the following steps:

1. Run **smitty cspoc** and then select **Resource Group and Applications** → **Bring a Resource Group Online**.
2. Select an RG from the list.

3. Select a destination node from the list, as shown in Figure 6-5.

The final SMIT menu opens, with the information that was selected in the previous lists.

4. Verify the entries that were previously specified and then press Enter to start moving the RG.

On successful completion, PowerHA displays a message and the status, location, and a type of location of the RG that was successfully started on the specified node.

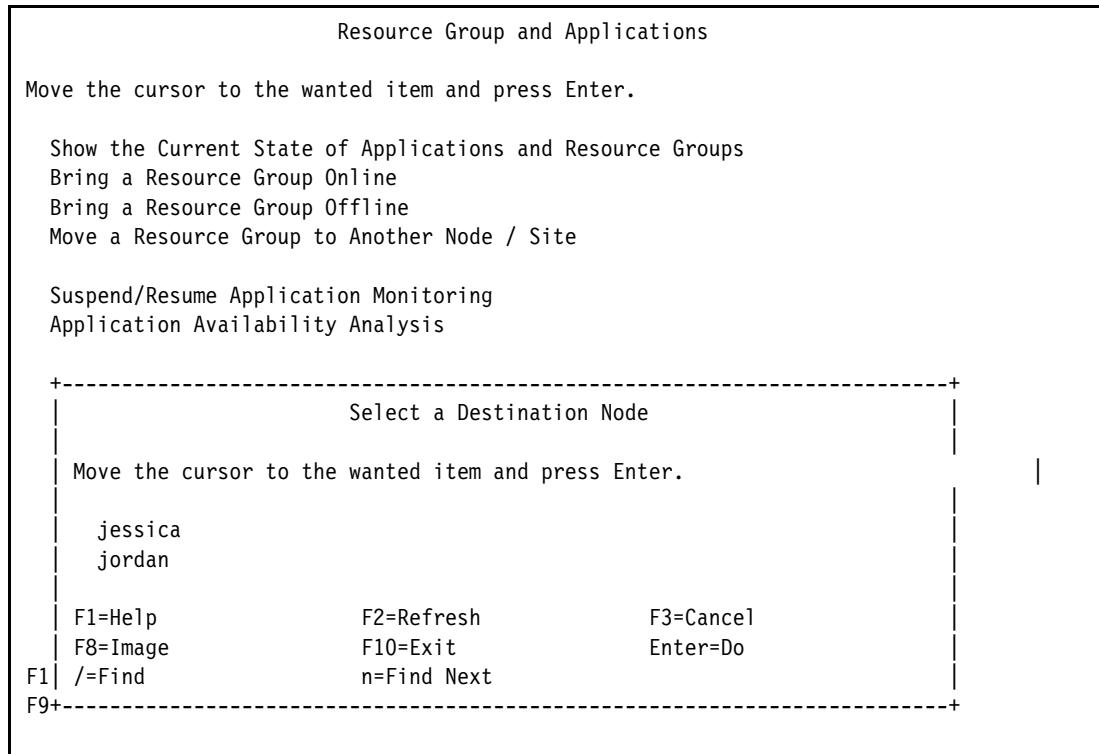


Figure 6-5 Destination node list

Bringing a resource group online by using SMUI

For a demonstration about how to bring an RG online by using the PowerHA SMUI, see this [YouTube video](#).

Bringing a resource group online by using clmgr

The `clmgr` command can be used to bring an RG online from the CLI, as shown in Example 6-4.

Example 6-4 Bringing a resource group online by using clmgr

```
# clmgr online rg redbookrg
Attempting to bring group redbookrg online on node jessica.

Waiting for the cluster to stabilize.....

Resource group movement is successful.
Resource group redbookrg is online on node jessica.

Cluster Name: jessica_cluster
```

Resource Group Name: redbookrg		Group State
Node		
jessica		ONLINE
jordan		OFFLINE

6.3.3 Moving a resource group

Moving an RG consists of releasing the resources on the current owner node and then processing the normal RG startup procedures on the destination node. This process results in a short period in which the application is not available.

PowerHA can move an RG to another site. The concept is the same as moving it between local nodes. For our example, we use the option to move to another node rather than to another site. As is the case with most operations, this operation can be performed from the CLI, SMUI, or SMIT.

Moving a resource group by using SMIT

To move an RG, complete the following steps:

1. Run **smitty cspoc** and then select **Resource Group and Applications** → **Move a Resource Group to Another Node/Site** → **Move Resource Groups to Another Node**. A list opens. It lists only the RGs that are online in the ERROR or UNMANAGED states on all nodes in the cluster.
 2. Select the appropriate RG from the list and press Enter.
- Another list opens (Select a Destination Node). The list contains only those nodes that are active in the cluster and are participating nodes in the previously selected RG.
3. Select a destination node from the list.

The final SMIT menu opens with the information that was selected in the previous lists (see Figure 6-6).

Figure 6-6 Move Resource Groups to Another Node panel

4. Verify the entries that were previously specified and then press Enter to start moving the RG.

On successful completion, PowerHA displays a message and the status, location, and a type of location of the RG that was successfully stopped on the specified node, as shown in Figure 6-7.

```
COMMAND STATUS

Command: OK          stdout: yes          stderr: no

Before command completion, additional instructions might appear below.

[MORE...7]
Resource group xsiteGLVMRG is online on node jordan.

Cluster Name: PHASEtoEE

Resource Group Name: xsiteGLVMRG
Node           Primary State   Secondary State
-----
jessica@dallas      OFFLINE       ONLINE SECONDARY
jordan@fortworth    ONLINE        OFFLINE

[BOTTOM]

F1=Help          F2=Refresh       F3=Cancel       F6=Command
F8=Image          F9=Shell         F10=Exit        /=Find
n=Find Next
```

Figure 6-7 Resource Group Status post-move

This option is also available by using either the `c1RGinfo` or `c1mgr` command. For more information about these commands, see the [man pages](#).

Anytime that an RG is moved to another node, application monitoring for the applications is suspended during the application stop. After the application restarts on the destination node, application monitoring resumes. For more information, see 6.3.4, “Suspending and resuming application monitoring” on page 209.

Moving a resource group by using the SMUI

For a demonstration about how to move an RG by using the PowerHA SMUI, see this [YouTube video](#).

Moving a resource group by using `clmgr`

The `c1mgr` command can be used to move an RG by using the CLI, as shown in Example 6-5.

Example 6-5 Moving a resource group by using `clmgr`

```
# clmgr move rg redbookrg node=jordan
Attempting to move the resource group redbookrg to node jordan.

Waiting for the cluster to stabilize...

Resource group movement is successful.
Resource group redbookrg is online on node jordan

Cluster Name: jessica_cluster

Resource Group Name: redbookrg
```

Node	Group State
jessica	OFFLINE
jordan	ONLINE

6.3.4 Suspending and resuming application monitoring

During application maintenance periods, taking the application offline only is often what you want instead of stopping cluster services. If application monitoring is being used, you must suspend application monitoring before stopping the application. Otherwise, PowerHA takes the predefined recovery procedures when it detects that the application is down, which is not what you want during maintenance. Defining application monitors is explained in 7.7.6, “Using the clanalyze log analysis tool” on page 330.

To suspend application monitoring, complete the following steps:

1. Run **smitty cspoc** and then select **Resource Group and Applications** → **Suspend/Resume Application Monitoring** → **Suspend Application Monitoring**. Press Enter.
2. You are prompted to select the application server for which this monitor is configured. If you have multiple application monitors, they are all suspended until you choose to resume them or until a cluster event occurs to resume them automatically.
3. For the Resource Group field, press F4 and choose the appropriate RG, as shown in Figure 6-8.
4. Press Enter twice to choose and run the process.

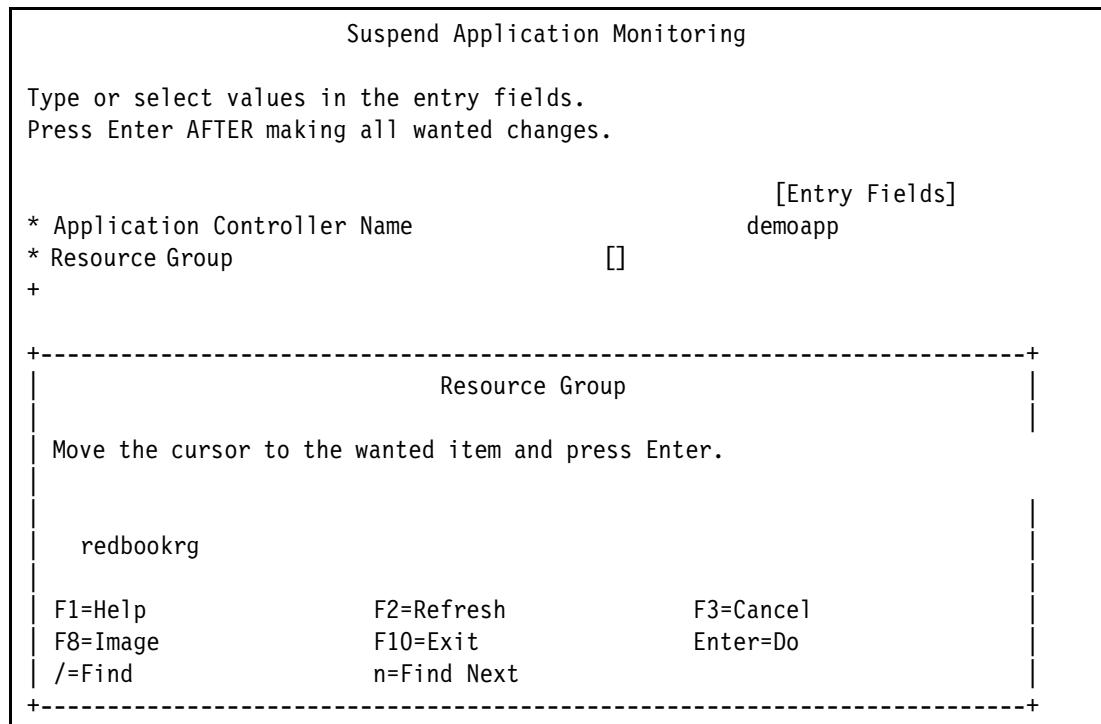


Figure 6-8 Suspend Application Monitoring

The monitoring remains suspended until either it is manually resumed or until the RG is stopped and restarted.

Suspending application monitoring by using the SMUI

Unlike RGs, manipulating the application monitoring *cannot* be performed by using the SMUI.

Suspending application monitoring by using clmgr

The `clmgr` command can be used to suspend application monitoring by using the CLI, as shown in Example 6-6. The syntax is as follows:

```
clmgr manage application suspend <application> rg=<resource_group>
```

Example 6-6 Suspending application monitoring by using clmgr

```
# clmgr manage application suspend demoapp rg=redbookrg
```

```
Beginning the process of suspending monitoring. Waiting up to 60 seconds for  
the state change to take effect...
```

```
Monitoring for application "demoapp", running in resource group "redbookrg",  
has been successfully suspended
```

Resuming application monitoring by using SMIT

To resume application monitoring, complete the following steps:

1. Run `smitty cspoc` and then select **Resource Group and Applications** → **Suspend/Resume Application Monitoring** → **Resume Application Monitoring**. Press Enter.
2. Choose the appropriate application server that is associated with the application monitor that you want to resume from the list.
3. For the Resource Group field, press F4 and choose the appropriate RG, as shown in Figure 6-8 on page 209.
4. Press Enter twice to choose and run the process.

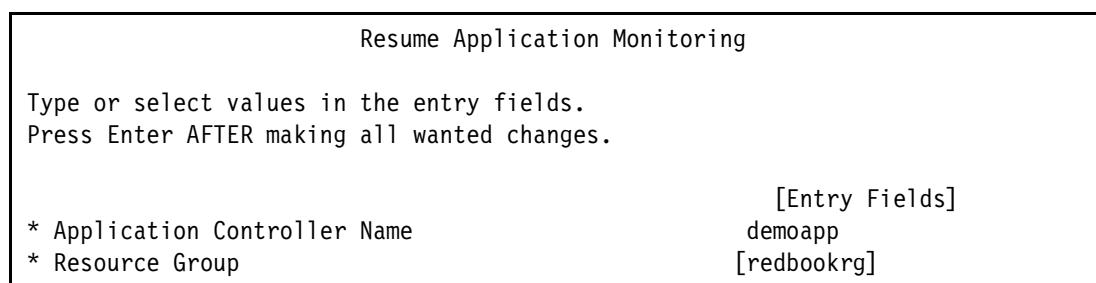


Figure 6-9 Resuming application monitoring by using SMIT

Application monitoring continues to stay active until either manually suspended or until the RG is brought offline.

Resuming application monitoring by using the SMUI

Unlike RGs, manipulating the application monitoring *cannot* be performed by using the SMUI.

Resuming application monitoring by using clmgr

The `clmgr` command can be used to resume application monitoring from the CLI, as shown in Example 6-7 on page 211. The syntax is as follows:

```
clmgr manage application resume <application> rg=<resource_group>
```

Example 6-7 Resuming application monitoring by using clmgr

```
# clmgr manage application resume demoapp rg=redbookrg
Beginning the process of resuming monitoring. Waiting up to 60 seconds for the
state change to take effect...
```

```
Monitoring for application "demoapp", running in resource group "redbookrg",
has resumed successfully.
```

6.4 Scenarios

In this section, we cover the following common scenarios:

- ▶ PCI hot-plug replacement of a network interface card (NIC)
- ▶ Installing AIX and PowerHA fixes
- ▶ Replacing a Logical Volume Manager (LVM) mirrored disk
- ▶ Application maintenance

6.4.1 PCI hot-plug replacement of a NIC

This section describes the process of replacing a PCI hot-pluggable NIC by using the C-SPOC PCI Hot Plug Replace a Network Interface Card facility.

Note: Although PowerHA continues to provide this facility with virtualization primarily being used, this procedure is rarely used.

Special considerations

Consider the following factors before you replace a PCI hot-pluggable NIC:

- ▶ Manually record the IP address settings of the network interface that is being replaced to prepare for unplanned failures.
- ▶ If a network interface that you are hot-replacing is the only available keepalive path on the node where it is, you *must* shut down PowerHA on this node to prevent a partitioned cluster while the interface is being replaced.

This situation is avoidable by having a working non-IP network between the cluster nodes.

- ▶ With SMIT, you may do a graceful shutdown on this node. From this point, you can manually hot-replace the NIC.
- ▶ Hot-replacement of Ethernet NICs is supported.
- ▶ Do not attempt to change any configuration settings while hot-replacement is in progress.

The SMIT interface simplifies the process of replacing a PCI hot-pluggable NIC. PowerHA supports only one PCI hot-pluggable NIC replacement by using SMIT at one time per node.

Note: If the network interface was alive before the replacement process began, then between the initiation and completion of the hot-replacement, the interface being replaced is in a maintenance mode. During this time, network connectivity monitoring is suspended on the interface during the replacement process.

Scenario 1: Live NICs only

Use this procedure when hot-replacing the following interfaces:

- ▶ A live PCI network service interface in an RG and with an available non-service interface.
- ▶ A live PCI network service interface that is not in an RG and with an available non-service interface.
- ▶ A live PCI network boot interface with an available non-service interface.

Go to the node on which you want to replace a hot-pluggable PCI NIC and complete the following steps:

1. Run **smitty cspoc**, and then select **Communication Interfaces → PCI Hot Plug Replace a Network Interface Card**. Press Enter.

Tip: You can also get to this panel by running **smitty cl_pcihp**.

SMIT displays a list of available PCI network interfaces that are hot-pluggable.

2. Select the network interface that you want to hot-replace. Press Enter. The service address of the PCI interface is moved to the available non-service interface.
3. SMIT prompts you to physically replace the NIC. After you replace the card, confirm that the replacement occurred.
 - If you select Yes, the service address is moved back to the network interface that was hot-replaced. On aliased networks, the service address does not move back to the original network interface, but remains as an alias on the same network interface. The hot-replacement is complete.
 - If you select No, you must manually reconfigure the interface settings to their original values:
 - i. Run the **drslot** command to take the PCI slot out of the removed state.
 - ii. Run **mkdev** on the physical interface.
 - iii. Use **ifconfig** manually (rather than **smitty chinet**, **cfgmgr**, or **mkdev**) to avoid configuring duplicate IP addresses or an unwanted boot address.

Scenario 2: Live NICs only

Follow this procedure when hot-replacing a live PCI network service interface on an RG but with no available non-service interface. Steps 1 - 3 on page 213 are the same as in “Scenario 1: Live NICs only”, so in this scenario we start from the SMIT fast path of **smitty cl_pcihp**:

1. Select the network interface that you want to hot-replace and press Enter.
2. SMIT prompts you to choose whether to move the RG to another node during the replacement process to ensure its availability.
 - If you choose to move the RG, SMIT gives you the option of moving the RG back to the node on which the hot-replacement took place after completing the replacement process.
 - If you choose not to move the RG to another node, it will be offline during the replacement process.

3. SMIT prompts you to physically replace the NIC. After you replace the card, confirm that the replacement occurred.
 - If you select Yes, the hot-replacement is complete.
 - If you select No, you must manually reconfigure the interface settings to their original values:
 - i. Run the **drsSlot** command to take the PCI slot out of the removed state.
 - ii. Run **mkdev** on the physical interface.
 - iii. Use **ifconfig** manually (rather than **smitty chinet**, **cfgmgr**, or **mkdev**) to avoid configuring duplicate IP addresses or an unwanted boot address.
 - iv. If applicable, move the RG back to the node from which you moved it in Step 2 on page 212.

Scenario 3: Non-alive NICs only

Use this procedure when hot-replacing the following interfaces:

- ▶ A non-alive PCI network service interface in an RG and with an available non-service interface.
- ▶ A non-alive PCI network service interface that is *not* in an RG and with an available non-service interface.
- ▶ A non-alive PCI network boot interface with an available non-service interface.

We begin again from the fast path of **smitty cl_pcihp** as in the previous scenario:

Select the network interface that you want to hot-replace and press Enter. SMIT prompts you to physically replace the NIC. After you replace it, confirm that the replacement occurred.

- ▶ If you select Yes, the hot-replacement is complete.
- ▶ If you select No, you must manually reconfigure the interface settings to their original values:
 - a. Run the **drsSlot** command to take the PCI slot out of the removed state.
 - b. Run **mkdev** on the physical interface.
 - c. Use **ifconfig** manually (rather than **smitty chinet**, **cfgmgr**, or **mkdev**) to avoid configuring duplicate IP addresses or an unwanted boot address.

6.4.2 Service Packs

This section relates to installing fixes, previously referred to as APARs or PTFs. AIX now has maintenance updates that are known as Technology Levels (TLs), and Service Packs (SPs) for the TLs. PowerHA adopted the AIX process of creating SPs.

Some AIX fixes can be loaded dynamically without a restart. Kernel and device driver updates often require a restart because installing updates to them runs **bosboot**. One way to determine whether a restart is required is to check the .toc file that is created. To do so, run the **inutoc** command before installing the fixes. The .toc file contains file set information similar to Example 6-8.

Example 6-8 Checking the .toc file before installing fixes

bos.64bit	7.2.3.0	I b usr,root
# Base Operating System 64-bit Runtime		
bos.acct	7.2.3.0	I N usr,root
# Accounting Services		

In the example, the file set `bos.64bit` requires a restart, as indicated by the `b` character in the fourth column. The `N` character indicates that a restart is not required.

Applying PowerHA fixes is similar to applying AIX fixes, but restarting after installing the base file sets is not required. However, other base prerequisites like RSCT might require a restart. Always check with IBM Support if you are unsure about the effects of loading certain fixes.

When you update AIX or PowerHA software, make sure to perform the following tasks:

- ▶ Take a cluster snapshot and save it somewhere off the cluster.
- ▶ Back up the operating system and data before performing any upgrade. Prepare a backout plan in case you encounter problems with the upgrade.
- ▶ *Always* perform procedures on a test cluster before running them in production.
- ▶ Use `alt_disk` update or Live Update if possible.

Note: Follow this same general rule for fixes to the application. Follow specific instructions for the application.

The general procedure for applying AIX fixes that require a restart is as follows:

1. Stop cluster services on the standby node.
2. *Apply, but* do not commit, the TL or SP to the standby node (and restart as needed).
3. Start cluster services on the standby node.
4. Stop cluster services on the production node by using Move Resource Group option to the standby machine.
5. *Apply the* TL or SP to the primary node (and restart as needed).
6. Start the cluster services on the primary node.

If you install either AIX or PowerHA fixes that do not require a restart, using the Unmanage Resource Groups option is now possible when stopping cluster services, as described in 6.2.3, “Stopping cluster services” on page 201. The general procedure for using this option for a two-node, hot-standby cluster is as follows:

1. Stop cluster services on the standby cluster by using the Unmanage option.
2. *Apply,* but do not commit, the SP to the standby node.
3. Start cluster services on the standby node.
4. Stop cluster services on the production node by using the Unmanage option.
5. *Apply the* SP to the primary node.
6. Start cluster services on the primary node.

Important: Never *unmanage* more than one node at a time. Complete the procedures thoroughly on one node before beginning on another node. Make sure to test these procedures in a test environment before attempting them in production.

Demonstration: For a demonstration about a nondisruptive update, see this [YouTube video](#).

Similarly, you can use the `cl_ezupdate` tool to perform nondisruptive updates for both AIX and PowerHA in a semi-automated fashion. For more information, see 5.3.6, “Migration by using `cl_ezupdate`” on page 179.

6.4.3 Storage

Most shared storage environments today use some level of redundant array of independent disks (RAID) for data protection and redundancy. In those cases, individual disk failures normally do not require AIX LVM maintenance to be performed. Any procedures that are required are often external to cluster nodes and do not affect the cluster itself. However, if protection is provided by using LVM mirroring, then LVM maintenance procedures are required.

C-SPOC provides the Cluster Disk Replacement facility to help in the replacement of a failed LVM mirrored disk. This facility does all the necessary LVM operations to replace an LVM mirrored disk. To use this facility, make sure that the following conditions are met:

- ▶ You have root privilege.
- ▶ The affected disk, and preferably the entire volume group (VG), is mirrored.
- ▶ The replacement disk that you want is available to each node, and a physical volume identifier (PVID) is already assigned to it. You can see the PVID of each node by running the **1spv** command.

To physically replace an existing disk, remove the old disk and replace the new one in its place. (Assuming that the drive is hot-plug replaceable, which is common.)

Important: C-SPOC cannot be used to replace a disk in Geographic Logical Volume Manager (GLVM) configurations.

To replace a mirrored disk by using C-SPOC, complete the following steps:

1. Locate the failed disk and note the PVID on the disk and the VG to which it belongs.
2. Run **smitty cspoc**, select **Storage → Physical Volumes → Cluster Disk Replacement**, and then press Enter.

SMIT displays a list of disks that are members of VGs that are contained in cluster RGs. There must be two or more disks in the VG where the failed disk is. The list includes the VG, the hdisk, the disk PVID, and the reference cluster node. (This node is usually the cluster node that has the VG varied on.)

3. Select the disk for disk replacement (*source disk*) and press Enter.

SMIT displays a list of those available shared disk candidates that have a PVID that is assigned to them to use for replacement. (Only a disk that has the same capacity or larger than the failed disk is suitable to replace the failed disk.)

4. Select a replacement disk (*destination disk*) and press Enter.

SMIT displays your selections from the two previous panels.

5. Press Enter to continue or Cancel to terminate the disk replacement process.

A warning message appears and states that continuing will delete any information that you might have stored on the destination disk.

6. Press Enter to continue or select Cancel to terminate.

SMIT displays a command status panel, and informs you about the **replacepv** command recovery directory. If disk configuration fails and you want to proceed with disk replacement, you must manually configure the destination disk. If you terminate the procedure at this point, be the destination disk can be configured on more than one node in the cluster.

The **replacepv** command updates the VG in use in the disk replacement process (on the reference node only).

Note: During command execution, SMIT provides the name of the recovery directory to use if `replacepv` fails. Note this information because it is required in the recovery process.

Configuration of the destination disk on all nodes in the RG occurs now.

If a node in the RG fails to import the updated VG, you can use the C-SPOC Import a Shared VG facility, as shown in “Importing volume groups by using C-SPOC” on page 278.

C-SPOC does not remove the failed disk device information from the cluster nodes. Remove it manually by running the `rmdev -dl <devicename>` command.

6.4.4 Applications

Most application maintenance requires the application to be brought offline, which can be done in several ways. The most appropriate method for an environment depends on the overall cluster configuration.

In a multitier environment where an application server depends on a database, and maintenance is performed on the database, usually both the database and the application server must be stopped, or at least the database is in the cluster must be stopped. When you use RG dependencies, the application server can be part of the same cluster.

It is common to help minimize the overall downtime of the application by performing the application maintenance first on nonproduction nodes for that application, which usually are standby nodes. However, it is not common that a backup or failover node is only a standby node. If the node is not a true standby node, then any work load or applications running on that node must be accounted for to minimize any adverse effects of installing maintenance. (Maintenance should have all been tested previously in a test cluster.)

Usually, stopping cluster services is not needed. You can bring the RG offline as described in 6.3.1, “Bringing a resource group offline” on page 203. If the shared VG must be online during maintenance, you can suspend application monitoring and start the application stop-server script to bring the application offline. However, this approach keeps the service IP address online, which might not be wanted.

In a multiple RG or multiple application environment that is running on the same node, stopping the cluster services on the local node might not be feasible. Be aware of the possible effects that might be caused by not stopping cluster services on the node in which application maintenance is being performed.

If during the maintenance period the system encounters a unrecoverable error that results in a crash, a failover occurs. This failover might be unwanted if the maintenance was not performed on the failover candidates first and the maintenance is incomplete on the local node. Although a failover might be a rare occurrence, the possibility exists.

Another possibility is that if another production node fails during the maintenance period, a failover can occur successfully on the local node without adverse effects. If this failover is not an acceptable result and there are multiple RGs, then you might want to move the other RGs to another node first and stop the cluster services on the local node.

If you use persistent addresses and you stop cluster services, local adapter swap protection is no longer provided. Although again rare, there is a possibility that when you use the persistent address to do maintenance and the hosting NIC fails, your connection will be dropped.

After application maintenance, *always* test the cluster again. Depending on what actions you selected to stop the application, you must reverse the process, for example, restart cluster services, bring the RG back online through C-SPOC, or manually run the application start server script and resume application monitoring as needed.

6.5 Updating multipath drivers

In most environments, some form of multipath device drivers is used for storage access. Eventually, you must update those drivers. Usually, when you do these updates, the devices must be offline. However, in CAA and the repository disk, the device is always active. This section describes how to stop services on the repository disk so that you can update the most common multipath devices.

Beginning in PowerHA 7.1.3 SP1, **c1mgr** was updated to allow CAA to be stopped on either a node, cluster, or site level. These steps can be done on one node at a time, or on the entire cluster. Our scenarios show how to do it either way.

6.5.1 Cluster-wide update

This scenario covers performing a cluster-wide update, which results in the entire cluster going offline. Complete the following steps:

1. Stop the cluster by running the following command on any node in the cluster. The results are shown in Example 6-9.

```
c1mgr offline cluster WHEN=now MANAGE=offline STOP_CAA=yes
```

CAA is inactive, and the CAA cluster and caavg_private no longer exist. This result is the same for all nodes in the cluster.

Example 6-9 Stopping all cluster services cluster-wide

```
[cassidy:root] / # c1mgr offline cluster WHEN=now MANAGE=offline STOP_CAA=yes
jessica: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.
jessica: 0513-044 The clevmgrdES Subsystem was requested to stop.
```

Broadcast message from root@cassidy (tty) at 12:26:05 ...

```
PowerHA SystemMirror on cassidy shutting down. Please exit any cluster
applications...
cassidy: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.
cassidy: 0513-044 The clevmgrdES Subsystem was requested to stop.
.....
```

The cluster is now offline.

```
jessica: Nov 22 2022 12:25:59 /usr/es/sbin/cluster/utilities/clstop: called with
flags -N -g
cassidy: Nov 22 2022 12:26:05 /usr/es/sbin/cluster/utilities/clstop: called with
flags -N -g
```

```
[cassidy:root] / # lspv
hdisk0          00f70c99013e28ca           rootvg      active
hdisk1          00f6f5d015a4310b          None
hdisk2          00f6f5d015a44307          None
```

hdisk3	00f6f5d01660fb01	None
hdisk4	00f6f5d0166106fa	xsitevg
hdisk5	00f6f5d0166114f3	xsitevg
hdisk6	00f6f5d029906df4	xsitevg
hdisk7	00f6f5d0596beebf	xsitevg
hdisk8	00f70c995a1bc94a	None

-
2. Perform the multipath driver update according to the vendor instructions.

Note: Sometimes, this step might change the device numbering. This change does not cause a problem because PowerHA and CAA know the repository disk by the PVID. However, check the disk device attributes (such as `reserve_policy`, `queue_depth`, and others) to make sure that they are still what you want.

3. Start cluster services by running the following command on any node in the cluster:

```
c1mgr online cluster WHEN=now MANAGE=auto START_CAA=yes
```

Important: If you use third-party storage multipathing device drivers, contact the vendor for support assistance. Consult IBM *only* if you use native AIX multipath I/O (MPIO).

4. After you perform maintenance, restart the cluster services as shown in Example 6-10. The CAA cluster and caavg_private are back and active.

Example 6-10 Starting cluster services cluster-wide after maintenance is performed

```
[cassidy:root] / # c1mgr online cluster WHEN=now MANAGE=auto START_CAA=yes

jessica: start_cluster: Starting PowerHA SystemMirror
jessica: 4391046 - 0:00 syslogd
jessica: Setting routerevalidate to 1
jessica: 0513-059 The clevmgrdES Subsystem has been started. Subsystem PID is
11665620.
cassidy: start_cluster: Starting PowerHA SystemMirror
cassidy: 3014870 - 0:00 syslogd
cassidy: Setting routerevalidate to 1
cassidy: 0513-059 The clevmgrdES Subsystem has been started. Subsystem PID is
17104906.

Broadcast message from root@cassidy (tty) at 12:31:36 ...

Starting Event Manager (clevmgrdES) subsystem on cassidy

The cluster is now online.

Starting Cluster Services on node: jessica
This may take a few minutes. Please wait...
jessica: Nov 22 2022 12:31:26 Starting execution of
/usr/es/sbin/cluster/etc/rc.cluster
jessica: with parameters: -boot -N -A -b -P cl_rc_cluster
jessica:
jessica: Nov 22 2022 12:31:26 Checking for srcmstr active...
jessica: Nov 22 2022 12:31:26 complete.
jessica: Nov 22 2022 12:31:27
```

```

jessica: /usr/es/sbin/cluster/utilities/clstart: called with flags -m -G -b -P
cl_rc_cluster -B -A
jessica:
jessica:      Nov 26 2022 17:16:42
jessica: Completed execution of /usr/es/sbin/cluster/etc/rc.cluster
jessica: with parameters: -boot -N -A -b -P cl_rc_cluster.
jessica: Exit status = 0
jessica:

Starting Cluster Services on node: cassidy
This may take a few minutes. Please wait...
cassidy: Nov 22 2022 12:31:34 Starting execution of
/usr/es/sbin/cluster/etc/rc.cluster
cassidy: with parameters: -boot -N -A -b -P cl_rc_cluster
cassidy:
cassidy: Nov 22 2022 12:31:34 Checking for srcmstr active...
cassidy: Nov 22 2022 12:31:34 complete.
cassidy: Nov 22 2022 12:31:35
cassidy: /usr/es/sbin/cluster/utilities/clstart: called with flags -m -G -b -P
cl_rc_cluster -B -A
cassidy:
cassidy:      Nov 22 2022 12:31:36
cassidy: Completed execution of /usr/es/sbin/cluster/etc/rc.cluster
cassidy: with parameters: -boot -N -A -b -P cl_rc_cluster.
cassidy: Exit status = 0
cassidy:

[cassidy:root] / # lspv
hdisk0      00f70c99013e28ca          rootvg      active
hdisk1    00f6f5d015a4310b          caavg_private  active
hdisk2      00f6f5d015a44307         None
hdisk3      00f6f5d01660fdb1         None
hdisk4      00f6f5d0166106fa        xsitevg     concurrent
hdisk5      00f6f5d0166114f3        xsitevg     concurrent
hdisk6      00f6f5d029906df4        xsitevg     concurrent
hdisk7      00f6f5d0596beebf        xsitevg     concurrent
hdisk8      00f70c995a1bc94a        None

```

6.5.2 Individual node update

This scenario updates an individual node, which can be done one node at a time. Complete the following steps:

1. Stop the cluster services on the selected node by running the following command. The results are shown in Example 6-11 on page 220.
`c1mgr offline node <nodename> WHEN=now MANAGE=offline STOP_CAA=yes`
If you intend to move the RG, set `MANAGE=move` instead of `offline`.
2. Perform the multipath driver update according to the vendor instructions.

Note: Sometimes, this step might change the device numbering. This change does not cause a problem because PowerHA and CAA know the repository disk by the PVID. However, check the disk device attributes (such as `reserve_policy`, `queue_depth`, and others) to make sure that they are still what you want.

3. Start the cluster services on the selected node by running the following command:

```
c1mgr online node <nodename> WHEN=now MANAGE=auto START_CAA=yes
```

Important: If you use third-party storage multipathing device drivers, contact the vendor for support assistance. Consult IBM *only* if you use native AIX multipath I/O (MPIO).

4. Repeat these steps as needed on one node at a time.

The results of step 1 on page 219 are shown in Example 6-11. CAA is inactive, but the CAA cluster and caavg_private no longer exist on node cassidy. This result applies only to the individual node in this case. Also, the cluster exists and is still active on node jessica.

Example 6-11 Stopping all cluster services on an individual node

```
[cassidy:root] / # c1mgr stop node cassidy WHEN=now MANAGE=offline STOP_CAA=yes
Broadcast message from root@cassidy (tty) at 16:24:19 ...
```

```
PowerHA SystemMirror on cassidy shutting down. Please exit any cluster
applications...
cassidy: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.
cassidy: 0513-044 The clevmgrdES Subsystem was requested to stop.
.
"cassidy" is now offline.
```

```
cassidy: Nov 22 2022 16:24:19 /usr/es/sbin/cluster/utilities/clstop: called with
flags -N -g
[cassidy:root] / # lsv
hdisk0      00f70c99013e28ca          rootvg      active
hdisk1      00f6f5d015a4310b
hdisk2      00f6f5d015a44307         None
hdisk3      00f6f5d01660fb01         None
hdisk4      00f6f5d0166106fa        xsitevg
hdisk5      00f6f5d0166114f3        xsitevg
hdisk6      00f6f5d029906df4        xsitevg
hdisk7      00f6f5d0596beebf        xsitevg
hdisk8      00f70c995a1bc94a         None
```

```
[jessica:root] / # lsv
hdisk0      00f6f5d00146570c          rootvg      active
hdisk1      00f6f5d015a4310b
hdisk2      00f6f5d01660fb01         amyvg
hdisk3      00f6f5d015a44307         amyvg
hdisk4      00f6f5d0166106fa        xsitevg      concurrent
hdisk5      00f6f5d0166114f3        xsitevg      concurrent
hdisk6      00f6f5d029906df4        xsitevg      concurrent
hdisk7      00f6f5d0596beebf        xsitevg      concurrent
```

- After performing maintenance, restart the cluster services on node cassidy, as shown in Example 6-12. The CAA cluster and caavg_private are back and active.

Example 6-12 Starting the cluster services on an individual node after maintenance

```
[cassidy:root] / # clmgr start node cassidy WHEN=now MANAGE=auto START_CAA=yes
```

```
cassidy: start_cluster: Starting PowerHA SystemMirror
cassidy: 3014870      - 0:00 syslogd
cassidy: Setting routerevalidate to 1
cassidy: 0513-059 The clevmgrdES Subsystem has been started. Subsystem PID is
17039420.
```

```
Broadcast message from root@cassidy (tty) at 17:18:19 ...
```

```
Starting Event Manager (clevmgrdES) subsystem on cassidy
```

```
....  
"cassidy" is now online.
```

```
Starting Cluster Services on node: cassidy
This may take a few minutes. Please wait...
cassidy: Nov 22 2022 17:18:17 Starting execution of
/usr/es/sbin/cluster/etc/rc.cluster
cassidy: with parameters: -boot -N -A -b -P cl_rc_cluster
cassidy:
cassidy: Nov 22 2022 17:18:17 Checking for srcmstr active...
cassidy: Nov 22 2022 17:18:17 complete.
cassidy: Nov 22 2022 17:18:18
cassidy: /usr/es/sbin/cluster/utilities/clstart: called with flags -m -G -b -P
cl_rc_cluster -B -A
cassidy:
cassidy: Nov 22 2022 17:18:19
cassidy: Completed execution of /usr/es/sbin/cluster/etc/rc.cluster
cassidy: with parameters: -boot -N -A -b -P cl_rc_cluster.
cassidy: Exit status = 0
cassidy:
```

[cassidy:root] / # lspv			
hdisk0	00f70c99013e28ca	rootvg	active
hdisk1	00f6f5d015a4310b	caavg_private	active
hdisk2	00f6f5d015a44307	None	
hdisk3	00f6f5d01660fb01	None	
hdisk4	00f6f5d0166106fa	xsitevg	concurrent
hdisk5	00f6f5d0166114f3	xsitevg	concurrent
hdisk6	00f6f5d029906df4	xsitevg	concurrent
hdisk7	00f6f5d0596beebf	xsitevg	concurrent
hdisk8	00f70c995a1bc94a	None	

6.5.3 Steps for maintenance on PowerHA before Version 7.1.3 SP1

The steps that are listed in 6.5, “Updating multipath drivers” on page 217 were added in PowerHA 7.1.3 SP1. For more information about the steps for maintenance on earlier PowerHA 7 releases, contact IBM Support.

6.6 Repository disk replacement

If you encounter a hardware error on the repository disk or migrate storage subsystems, you must swap or replace the repository disk.

6.6.1 Automatic Repository Update

Beginning in PowerHA 7.2.0, Automatic Repository Update (ARU) can automatically swap a failed repository disk with a backup repository disk. A maximum of six repository disks per site can be defined in a cluster. The backup disks are polled once a minute by `c1confd` to verify that they are still viable for an ARU operation. The configuration of the ARU is automatic when you configure a backup repository disk for PowerHA. Essentially, the only step that is required is to define a backup or add a repository disk, as described in 6.6.2, “Manual repository swap” on page 222.

Requirements for Automatic Repository Update

The requirements for the Automatic Repository Update are as follows:

- ▶ AIX 7.1.4 or AIX 7.2.0 or later.
- ▶ PowerHA 7.2.0 or later.
- ▶ The storage that is used for the backup repository disk must meet the same requirements that the primary repository disk does. For more information about the PowerHA repository disk requirements, see [PowerHA SystemMirror 7.2 for AIX](#).

Demonstration: For a demonstration of automatic repository replacement, see this [YouTube video](#).

6.6.2 Manual repository swap

To perform a manual swap of the repository disk for PowerHA 7.2.7, complete the following steps:

1. Ensure that you have a new shared disk that is accessible by all cluster nodes. The shared disk's PVID must be known to each node in the cluster.
2. Add the new shared disk as a repository disk by running `smitty cm_add_repository_disk`. Choose the disk either by pressing F4 and selecting it from the list, or manually enter it, as shown in Figure 6-10 on page 223.

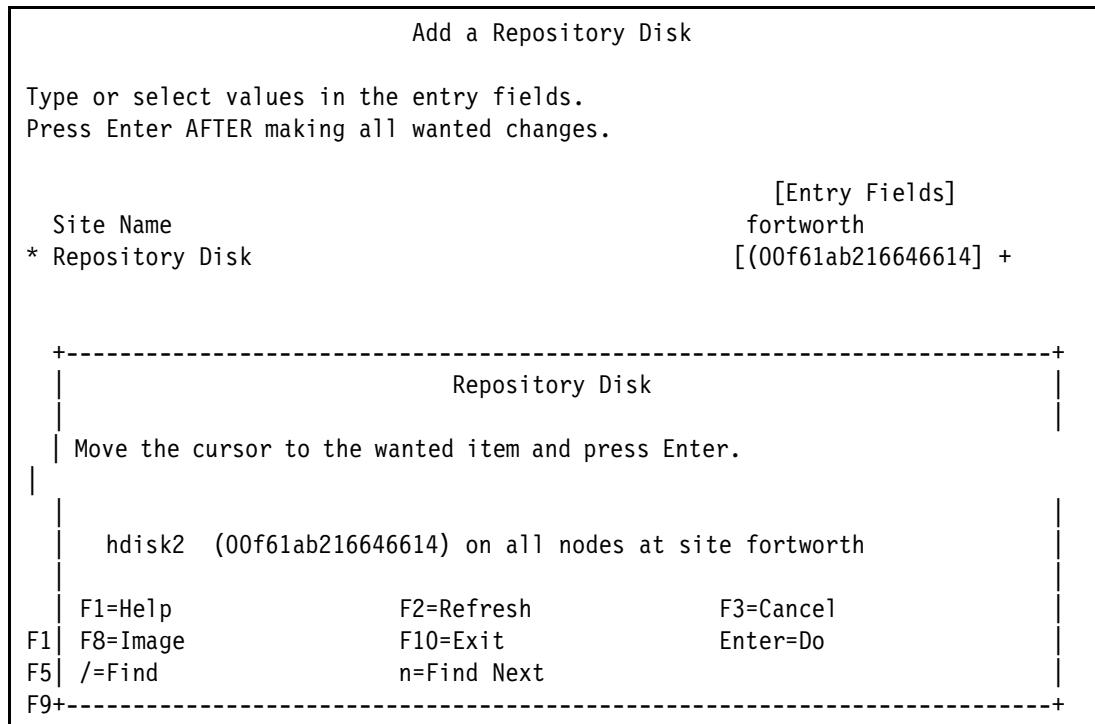


Figure 6-10 Add a Repository Disk

3. Run **smitty sysmirror**, select **Problem Determination Tools → Replace the Primary Repository Disk**, and then press Enter.
4. If your sites are defined, you can select a site from the list. Otherwise, you are directed to the last SMIT menu.
5. Select the new repository disk by pressing F4. See Figure 6-11.

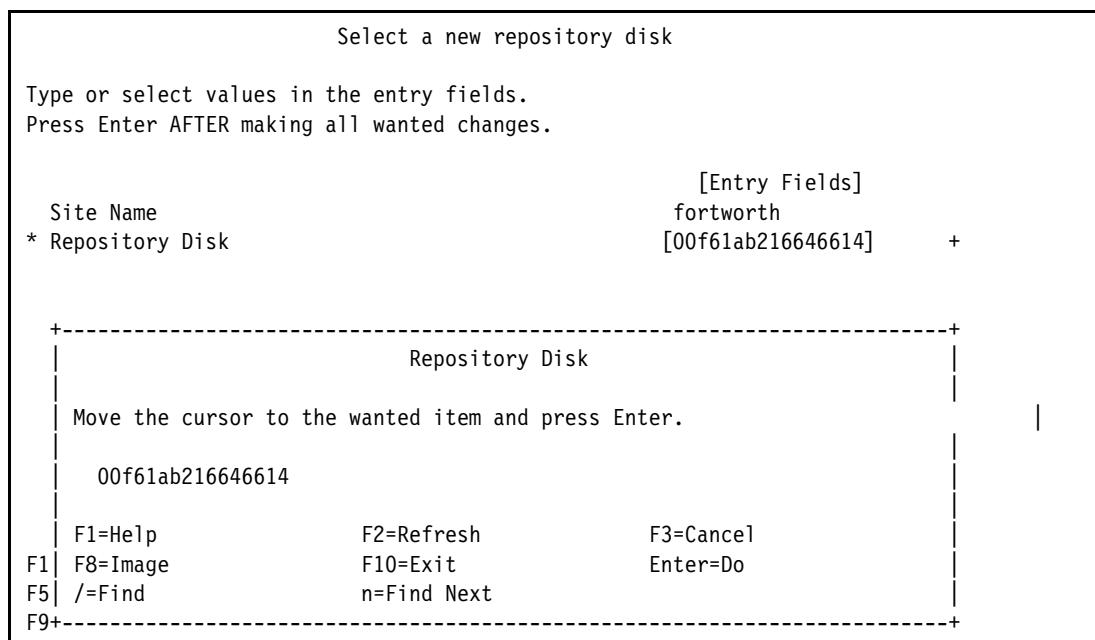


Figure 6-11 Replacing a repository disk

6. Synchronize the cluster.

This procedure of replacing a repository disk can also be accomplished by using the `clmgr` command, as shown in 6.7, “Critical volume groups” on page 224. If you are not using sites, you can exclude the site option from the syntax.

Example 6-13 The clmgr replace repository process

```
[jordan:root] /cluster # clmgr replace repository hdisk2 SITE=fortworth
```

```
***Warning: this operation destroys any information that is currently stored on  
"hdisk2". Are you sure you want to proceed? (y/n)
```

The configuration must be synchronized to make this change known across the cluster.

Then, synchronize the cluster as stated in the output by using `clmgr sync cluster`.

6.7 Critical volume groups

The PowerHA SystemMirror critical VG feature provides multiple node disk heartbeat functions for the Oracle Real Application Cluster (RAC) voting disk. This feature was introduced in PowerHA 7.1, and it is a replacement for the Multi-Node Disk Heart Beat technology.

Critical VGs safeguard the Oracle RAC voting disks. PowerHA continuously monitors the read/write accessibility of the voting disks. You can set up one of the following recovery actions if you lose access to a VG:

- ▶ Notify only.
- ▶ Halt the node.
- ▶ Fence the node so that the node remains up but cannot access the Oracle database.
- ▶ Shut down cluster services and bring all RGs offline.

Important: Critical VGs and Multi-Node Disk Heart Beat do not replace the storage area network (SAN)-based disk heartbeat. These technologies are used for separate purposes.

Do not use critical VGs instead of the SAN-based heartbeat.

If you have Oracle RAC, you must have at least one designated VG for voting. To configure a critical VG, complete the following steps:

1. Set up a concurrent RG for two or more nodes by defining the following items:
 - Startup policy: Online on all available nodes
 - Failover policy: Bring offline
 - Fallback policy: Never fallback
2. Create an ECVG that is accessible for all nodes in the RG. This VG stores the Oracle RAC voting files.
3. Add the VG to the concurrent RG.
4. Synchronize your cluster.
5. Start `smitty cspoc` and select **Storage → Volume Groups → Manage Critical Volume Groups → Mark a Volume Group as Critical**.
6. Select the VG from the list, as shown in Figure 6-12 on page 225.

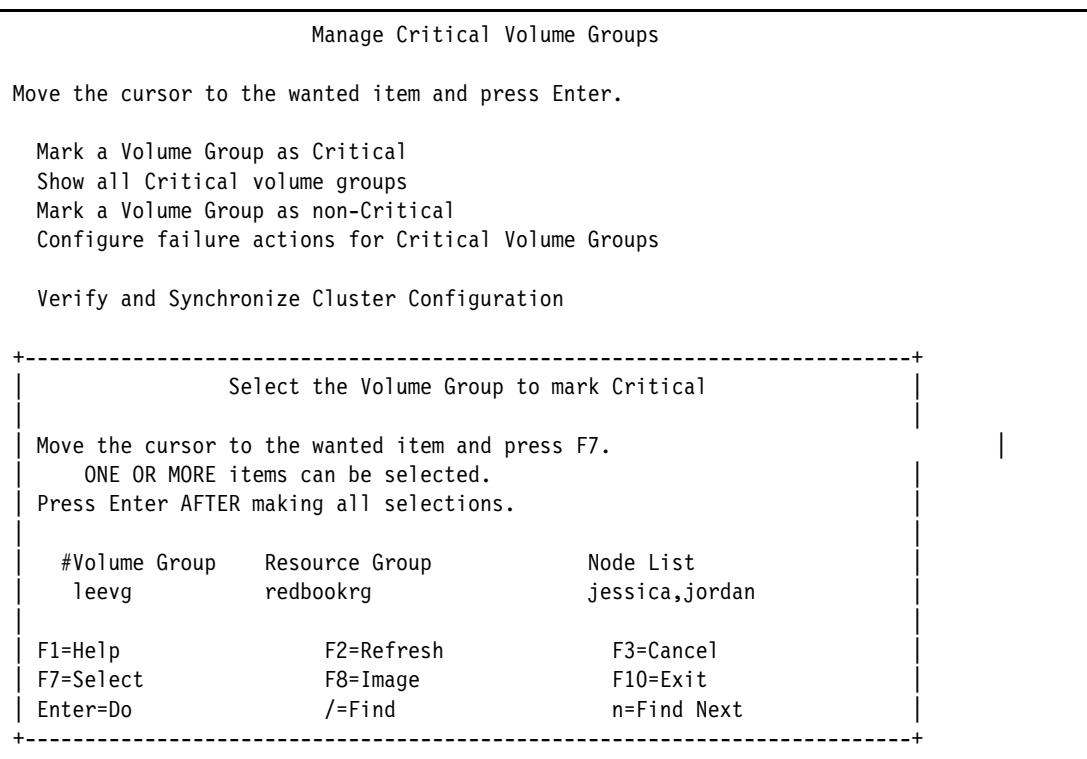


Figure 6-12 Marking a volume group as critical

7. Press Enter. You are prompted with a reminder to sync the cluster and what the default action is, as shown in Figure 6-13.

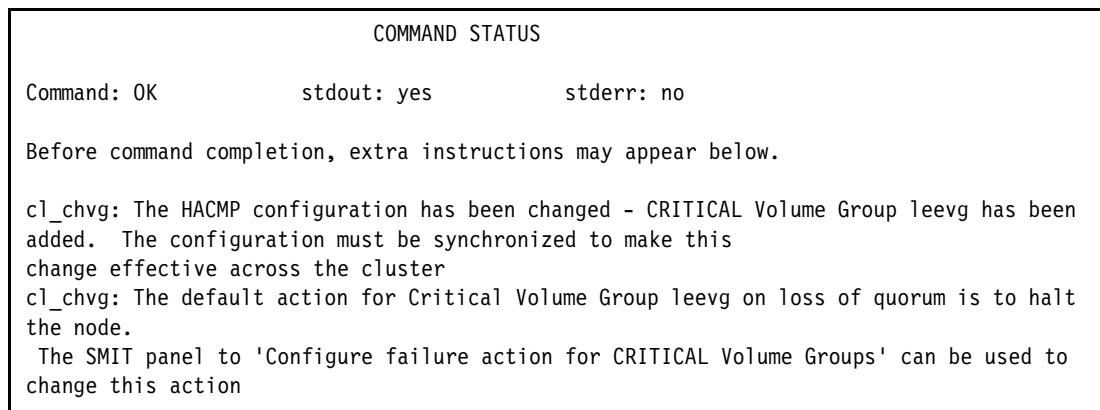


Figure 6-13 Output from marking a critical volume group

8. Configure the failure action by starting **smitty cspoc** and selecting **Storage → Volume Groups → Manage Critical Volume Groups → Configure failure actions for Critical Volume Groups**.
9. Select the VG from the list.
10. Select the recovery action that occurs if you lose disk access:
 - Notify Only
 - Halt the node
 - Fence the node
 - Shutdown Cluster Services and bring all Resource Groups Offline

11. Optionally, specify a notification method, as shown in Figure 6-14.

12. Synchronize the cluster by running `clmgr sync cluster`.

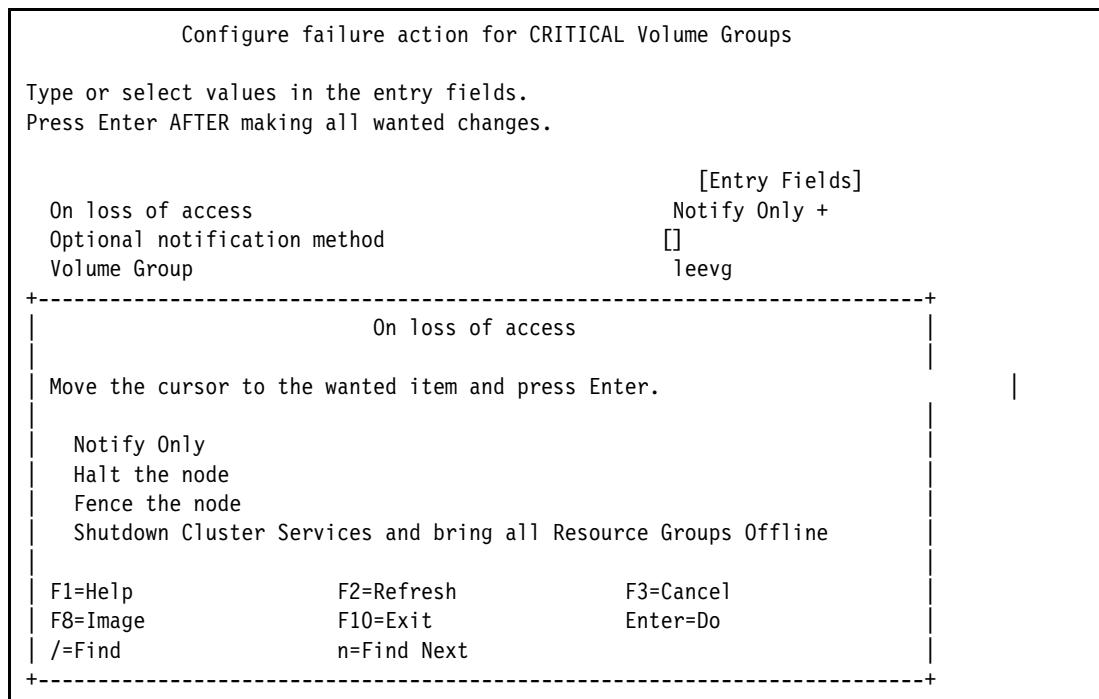


Figure 6-14 Configuring a failure action for critical volume groups

6.8 Cluster Test Tool

Use the Cluster Test Tool utility to test a PowerHA cluster configuration and to evaluate how a cluster operates under a set of specified circumstances, such as when cluster services on a node fail or when a node loses connectivity to a cluster network.

You can start a test, let it run unattended, and return later to evaluate the results of your testing. Run the utility under both low load and high load conditions to observe how system load affects your PowerHA cluster.

You run the Cluster Test Tool from SMIT on one node in a PowerHA cluster. For testing purposes, this node is referred to as the *control node*. From the control node, the tool runs a series of specified tests (some on other cluster nodes), gathers information about the success or failure of the tests, and stores this information in the Cluster Test Tool log file for evaluation or future reference.

Important: If you uninstall PowerHA, the program removes any files that you might have customized for the Cluster Test Tool. If you want to retain these files, copy them before you uninstall PowerHA.

6.8.1 Test duration

Running automated testing on a basic, two-node cluster that has a simple cluster configuration can take 30 - 60 minutes to complete. Usually, the most significant amount of time that is spent is restarting the applications. In our test, without a real application, the entire series of automated test completed in about 7 minutes.

Individual tests can take approximately 3 minutes to run. The following conditions affect the length of time to run the tests:

- ▶ Cluster complexity.
- ▶ Testing in complex environments takes considerably longer.
- ▶ Network latency.
- ▶ Cluster testing relies on network communication between the nodes. Any degradation in network performance slows the performance of the Cluster Test Tool.
- ▶ Use of verbose logging for the tool.
- ▶ Custom, user-defined resources or events.
- ▶ If you customize verbose logging to run more commands to capture output, testing takes longer to complete. In general, the more commands that you add for verbose logging, the longer a test procedure takes to complete.
- ▶ Manual intervention on the control node.
- ▶ At some points in the test, you might need to intervene.
- ▶ Running custom tests.
- ▶ If you run a custom test plan, the number of tests that are run also affects the time that is required to run the test procedure. If you run a long list of tests, or if any of the tests require a substantial amount of time to complete, then the time to process the test plan increases.

6.8.2 Considerations

The Cluster Test Tool has several considerations. It does not support testing of the following PowerHA cluster-related components:

- ▶ RGs with dependencies
- ▶ Replicated resources

You can perform general cluster testing for clusters that support sites, but not testing that is specific to PowerHA sites or any of the PowerHA Enterprise Edition products. Here are some situations regarding cluster testing:

- ▶ Replicated resources

You can perform general cluster testing for clusters that include replicated resources, but not testing that is specific to replicated resources, including GLVM, or any of the PowerHA Enterprise Edition products.

- ▶ Dynamic cluster reconfiguration

You cannot run dynamic reconfiguration while the tool is running.

- ▶ Pre-events and post-events

These events run in the usual way, but the tool does not verify that they were run or that the correct action was taken.

In addition, the Cluster Test Tool might not recover from the following situations:

- ▶ A node that fails unexpectedly, that is, a failure that is not initiated by testing.
- ▶ The cluster does not stabilize.

6.8.3 Automated testing

Use the automated test procedure (a predefined set of tests that is supplied with the tool) to perform basic cluster testing on any cluster. No setup is required. You can run the test from SMIT and view test results from SMIT and the Cluster Test Tool log file.

The automated test procedure runs a predefined set of tests on a node that the tool randomly selects. The tool ensures that the node that is selected for testing varies from one test to another one. You can run the automated test procedure on any PowerHA cluster that is not in service.

Running automated tests

You can run the automated test procedure on any PowerHA cluster that is not in service because the beginning of the test includes starting the cluster. The Cluster Test Tool runs a specified set of tests and randomly selects the nodes, networks, RGs, and other items for testing. The tool tests various cluster components during the testing. For a list of the tests that are run, see “Understanding automated testing” on page 228.

Before running the automated test, consider the following information:

- ▶ Ensure that the cluster is not in service in a production environment.
- ▶ Optional: Stop the PowerHA cluster services (optional).

Some of the tests include starting the cluster services on all nodes in the cluster. If the Cluster Manager is already running when you run those tests, the results will differ from what was expected, but the Cluster Test Tool continues to run.

- ▶ Cluster nodes are attached to two IP networks.

One network is used to test a network that becomes unavailable and then available. The second network provides network connectivity for the Cluster Test Tool. Both networks are tested, one at a time.

Understanding automated testing

These topics list the sequence that the Cluster Test Tool uses for the automated testing, and describes the syntax of the tests that run during automated testing.

The automated test procedure runs sets of predefined tests in this order:

1. General topology tests
2. RG tests on non-concurrent RGs
3. RG tests on concurrent RGs
4. IP-type network tests for each network
5. Non-IP network tests for each network
6. VG tests for each RG
7. Site-specific tests
8. Catastrophic failure test

The Cluster Test Tool discovers information about the cluster configuration and randomly selects cluster components, such as nodes and networks, to be used in the testing.

Which nodes are used in testing varies from one test to another one. The Cluster Test Tool can select some nodes for the initial battery of tests, and then for subsequent tests it can intentionally select the same nodes or choose from nodes on which no tests were run previously. In general, the logic in the automated test sequence ensures that all components are sufficiently tested in all necessary combinations.

Testing follows these rules:

- ▶ Tests the operation of a concurrent RG on one randomly selected node, but not all nodes in the RG.
- ▶ Tests only those RGs that include monitored application servers or VGs.
- ▶ Requires at least two active IP networks in the cluster to test non-concurrent RGs.

The automated test procedure runs a node_up event at the beginning of the test to ensure that all cluster nodes are running and available for testing.

Starting the Cluster Test Tool

You can use the Cluster Test Tool to run an automated test procedure by completing the following steps:

1. Run **smitty sysmirror**, select **Problem Determination Tools → Cluster Test Tool → Execute Automated Test Procedure**, and then press Enter.

The final SMIT panel opens with the following options:

Verbose Logging	When you select Yes (the default), extra information is included in the log file to help you judge the success or failure of some tests. Select No to decrease the amount of information that is logged by the Cluster Test Tool.
Cycle Log File	When you select Yes (the default), a new log file is used to store output from the Cluster Test Tool. Select No to append messages to the current log file.
Abort on Error	When you select No (the default), the Cluster Test Tool continues to run tests after some tests fail. This situation might cause subsequent tests to fail because the cluster state differs from the one that is expected by one of those tests. Select Yes to stop processing after the first test fails.

Select an option and then press Enter. A panel opens and prompts you to confirm your changes:

Are you sure?

2. Press Enter again, and the automated test plan begins.
3. Evaluate the test results.

General topology tests

The Cluster Test Tool runs the general topology tests in the following specific order:

1. Brings up a node and start the cluster services on all available nodes.
2. Stops the cluster services on a node and bring RGs offline.
3. Restarts the cluster services on the node that was stopped.
4. Stops the cluster services and move RGs to another node.
5. Restarts the cluster services on the node that was stopped.
6. Stops the cluster services on another node and place the RGs in an UNMANAGED state.

7. Restarts the cluster services on the node that was stopped.
8. Performs a selective failover on VG loss.
9. Performs a hard failure of a node to cause a failover.

The Cluster Test Tool uses legacy terminology for stopping cluster services. When the automated test procedure starts, the tool runs each of the following tests in this order:

1. Starts the cluster services on all available nodes:
NODE_UP, ALL
2. Stops the cluster services gracefully on a node:
NODE_DOWN_GRACEFUL, node1
3. Restarts the cluster services on the node that was stopped:
NODE_UP, node1
4. Stops the cluster services with a takeover on a node:
NODE_DOWN_TAKEOVER, node2
5. Restarts the cluster services on the node that was stopped:
NODE_UP, node2
6. Force stops the cluster services on a node:
NODE_DOWN_FORCED, node2
7. Restarts the cluster services on the node that was stopped:
NODE_UP, node3

Resource group tests

Two groups of RG tests can be run. Which group of tests to run depends on the startup policy for the RG: non-concurrent or concurrent. If a resource of the specified type does not exist in the RG, the tool logs an error in the Cluster Test Tool log file, which is in /var/hacmp/log/cl_testtool.log.

Resource group starts on a specified node

The following tests run if the cluster includes one or more RGs that have a startup management policy *other* than Online on All Available Nodes. That is, the cluster includes one or more non-concurrent RGs.

The Cluster Test Tool runs each of the following tests, in the order that is listed here, for each RG:

1. Brings an RG offline and online on a node:
RG_OFFLINE, RG_ONLINE
2. Brings down a local network on a node to produce an RG failover:
NETWORK_DOWN_LOCAL, rg_owner, svc1_net, Selective failover on local network down
3. Recovers the previously failed network:
NETWORK_UP_LOCAL, prev_rg_owner, svc1_net, Recover previously failed network

4. Moves an RG to another node:

RG_MOVE

5. Brings down an application server and recovers from the application failure:

SERVER_DOWN, ANY, app1, /app/stop/script, Recover from application failure

Resource group starts on all available nodes

If the cluster includes one or more RGs that have a startup management policy of Online on All Available Nodes (meaning that the cluster has concurrent RGs), the tool runs one test that brings down an application server and recovers from the application failure.

The tool runs the following test:

RG_OFFLINE, RG_ONLINE SERVER_DOWN, ANY, app1, /app/stop/script, Recover from application failure

Network tests

The tool runs tests for IP networks and for non-IP networks. For each IP network, the tool runs these tests:

- ▶ Brings a network down and up:
NETWORK_DOWN_GLOBAL, NETWORK_UP_GLOBAL
- ▶ Fails a network interface, then joins a network interface. This test is run for the service interface on the network. If no service interface is configured, the test uses a random interface that is defined on the network.
FAIL_LABEL, JOIN_LABEL

For each IP network, the tool runs the bring down and up a network test:

NETWORK_DOWN_GLOBAL, NETWORK_UP_GLOBAL

Volume group tests

The tool also runs tests for VGs. For each RG in the cluster, the tool runs tests that fail a VG in the VG_DOWN RG.

Site-specific tests

If sites are present in the cluster, the tool runs tests for them. The automated testing sequence that the Cluster Test Tool uses contains two site-specific tests:

- ▶ **auto_site**: This sequence of tests runs if you have any cluster configuration with sites. For example, this sequence is used for clusters with cross-site LVM mirroring configured that does not use XD_data networks. The tests in this sequence include the following ones:

SITE_DOWN_GRACEFUL Stops the cluster services on all nodes in a site while taking resources offline.

SITE_UP Restarts the cluster services on the nodes in a site.

SITE_DOWN_TAKEOVER Stops the cluster services on all nodes in a site and moves the resources to nodes at another site.

SITE_UP Restarts the cluster services on the nodes at a site.

RG_MOVE_SITE Moves an RG to a node at another site.

- ▶ **auto_site_isolation**: This sequence of tests runs only if you configured sites and an XD-type network. The tests in this sequence include the following ones:

SITE_ISOLATION	Isolates sites by failing XD_data networks.
SITE_MERGE	Merges sites by bringing up XD_data networks.

Catastrophic failure test

As a final test, the tool stops the Cluster Manager on a randomly selected node (see the following command) that has at least one active RG:

```
CLSTRMGR_KILL, node1, Kill the cluster manager on a node
```

When the tool terminates the Cluster Manager on the control node, you most likely will need to reactivate the node.

6.8.4 Custom testing

If you are an experienced PowerHA administrator and want to tailor cluster testing to your environment, you can create custom tests that can be run from SMIT.

You create a custom test plan (a file that lists a series of tests to be run) to meet requirements that are specific to your environment and apply that test plan to any number of clusters. You specify the order in which the tests run and the specific components to be tested. After you set up your custom test environment, you run the test procedure from SMIT and view test results in SMIT and in the Cluster Test Tool log file.

Planning a test procedure

Before you create a custom test procedure, make sure that you are familiar with the PowerHA clusters on which you plan to run the test. List the following components in your cluster and have this list available when setting up a test:

- ▶ Nodes
- ▶ IP networks
- ▶ Non-IP networks
- ▶ XD-type networks
- ▶ VGs
- ▶ RGs
- ▶ Application servers
- ▶ Sites

Your test procedure should bring each component offline and then online, or cause an RG failover to ensure that the cluster recovers from each failure. Start your test by running a node_up event on each cluster node to ensure that all cluster nodes are running and available for testing.

Creating a custom test procedure

A test plan is a text file that lists cluster tests to run in the order in which they are listed in the file. In a test plan, specify one test per line. You can set values for test parameters in the test plan or use variables to set parameter values.

Note: The Cluster Test Tool uses existing terminology for stopping cluster services as follows:

Graceful = Bring Resource Groups Offline
Takeover = Move Resource Groups
Forced = Unmanage Resource Groups

The tool supports the following tests:

FAIL_LABEL	Brings down the interface that is associated with the specified label on the specified node.
JOIN_LABEL	Brings up the interface that is associated with the specified label on the specified node.
NETWORK_UP_GLOBAL	Brings up a specified network (IP network or non-IP network) on all nodes that have interfaces on the network.
NETWORK_DOWN_GLOBAL	Brings down a specified network (IP network or non-IP network) on all nodes that have interfaces on the network.
NETWORK_UP_LOCAL	Brings up a network on a node.
NETWORK_DOWN_LOCAL	Brings down a network on a node.
NETWORK_UP_NONIP	Brings up a non-IP network.
NETWORK_DOWN_NONIP	Brings down a non-IP network.
NODE_UP	Starts the cluster services on the specified node.
NODE_DOWN_GRACEFUL	Stops the cluster services and brings the RGs offline on the specified node.
NODE_DOWN_TAKEOVER	Stops the cluster services with the resources that are acquired by another node.
NODE_DOWN_FORCED	Stops the cluster services on the specified node with the Unmanage Resource Group option.
CLSTRMGR_KILL	Terminates the Cluster Manager on the specified node.
RG_MOVE	Moves an RG that is already online to a specific node.
RG_MOVE_SITE	Moves an RG that is already online to an available node at a specific site.
RG_OFFLINE	Brings an RG offline that is already online.
RG_ONLINE	Brings an RG online that is already offline.
SERVER_DOWN	Brings down a monitored application server.
SITE_ISOLATION	Brings down all XD_data networks in the cluster at which the tool is running, which causes site isolation.
SITE_MERGE	Brings up all XD_data networks in the cluster at which the tool is running, which simulates a site merge. Run the SITE_MERGE test after running the SITE_ISOLATION test.
SITE_UP	Starts the cluster services on all nodes at the specified site that are stopped.
SITE_DOWN_TAKEOVER	Stops the cluster services on all nodes at the specified site and moves the resources to nodes at another site by starting automatic rg_move events.

SITE_DOWN_GRACEFUL	Stops the cluster services on all nodes at the specified site and takes the resources offline.
VG_DOWN	Emulates an error condition for a specified disk that contains a VG in an RG.
WAIT	Generates a wait period for the Cluster Test Tool.

Specifying parameters for tests

You can specify parameters for the tests in the test plan. Parameters can be specified in one of the following ways:

- ▶ By using a variables file. A variables file defines values for variables that are assigned to parameters in a test plan.
- ▶ By setting values for test parameters as environment variables.
- ▶ By identifying values for parameters in the test plan.

When the Cluster Test Tool starts:

- ▶ It uses a variables file if you specified the location of one in SMIT.
- ▶ If it does not locate a variables file, it uses values that are set in an environment variable.
- ▶ If a value is not specified in an environment variable, it uses the value in the test plan.
- ▶ If the value that is set in the test plan is not valid, the tool displays an error message.

Using a variables file

The variables file is a text file that defines the values for test parameters. By setting parameter values in a separate variables file, you can use your test plan to test more than one cluster.

The entries in the file have this syntax:

```
parameter_name = value
```

For example, the following entry specifies a node as node_lee:

```
node=node_lee
```

To provide more flexibility, you can do these tasks:

1. Set the name for a parameter in the test plan.
2. Assign the name to another value in the variables file. For example, you can specify the value for node as node1 in the test plan:

```
NODE_UP,node1
```

Bring up node1 in the variables file, and then set the value of node1 to node_lee:

```
node1=node_lee
```

Here is an example of a variables file:

```
node1=node_lee
node2=node_ashley
node3=node_briley
node4=node_keeley
```

Using a test plan

If you want to run a test plan on only one cluster, you can define test parameters in the test plan. The associated test can be run only on the cluster that includes those specified cluster attributes.

Description of the tests

The test plan supports the tests that are listed in this section. The description of each test includes information about the test parameters and the success indicators for a test.

Note: One of the success indicators for each test is that the cluster becomes stable.

Test syntax

The syntax for a test is as follows:

`TEST_NAME, parameter1, parametern|PARAMETER, comments`

- ▶ The test name is in uppercase letters.
- ▶ Parameters follow the test name.
- ▶ Italic text indicates parameters that are expressed as variables.
- ▶ Commas separate the test name from the parameters and the parameters from each other. A space around the commas is also supported.
- ▶ The syntax line shows parameters as *parameter1* and *parameter_n*, with *n* representing the next parameter. Tests typically have 2 - 4 parameters.
- ▶ The vertical bar, or pipe character (|), indicates parameters that are mutually exclusive alternatives.
- ▶ Optional: The comments part of the syntax is user-defined text that appears at the end of the line. The Cluster Test Tool displays the text string when the Cluster Test Tool runs.

In the test plan, the tool ignores these items:

- ▶ Lines that start with a number sign (#)
- ▶ Blank lines

Node tests

The node tests start and stop the cluster services on specified nodes:

- ▶ The following command starts the cluster services on a specified node that is offline or on all nodes that are offline:

`NODE_UP, node | ALL, comments`

The command options are as follows:

`node` The name of a node on which cluster services start.

`ALL` Any nodes that are offline will have their cluster services start.

`comments` User-defined text to describe the configured test.

- Example:

`NODE_UP, node1, Bring up node1`

- Entrance criteria

Any node that will be started is inactive.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services successfully start on all specified nodes.
- No RG enters the ERROR state.
- No RG moves from online to offline.

- ▶ The following command stops cluster services on a specified node and brings RGs offline:

`NODE_DOWN_GRACEFUL, node | ALL, comments`

The command options are as follows:

node The name of a node on which the cluster services stop.

ALL All nodes that are online that will have the cluster services stop. At least one node in the cluster must be online.

comments User-defined text to describe the configured test.

- Example:

`NODE_DOWN_GRACEFUL, node3, Bring down node3 gracefully`

- Entrance criteria

Any node that will be stopped is active.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services stop on the specified nodes.
- The cluster services continue to run on other nodes if ALL is not specified.
- RGs on the specified node go offline, and do not move to other nodes.
- RGs on other nodes remain in the same state.

- ▶ The following command stops the cluster services on a specified node with an RG that is acquired by another node as configured, depending on resource availability:

`NODE_DOWN_TAKEOVER, node, comments`

The command options are as follows:

node The name of a node on which to stop the cluster services.

comments User-defined text to describe the configured test.

- Example:

`NODE_DOWN_TAKEOVER, node4, Bring down node4 by moving the resource groups.`

- Entrance criteria

The specified node is active.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services stop on the specified node.
- The cluster services continue to run on other nodes.
- All RGs remain in the same state.

- ▶ The following command stops the cluster services on a specified node and unmanages the RGs. Resources on the node remain online (they are *not* released):

`NODE_DOWN_FORCED, node, comments`

The command options are as follows:

`node` The name of a node on which to stop the cluster services.

`comments` User-defined text to describe the configured test.

- Example:

`NODE_DOWN_FORCED, node2, Bring down node2 via unmanaged.`

- Entrance criteria

The cluster services on another node have not already been stopped with its RGs placed in an UNMANAGED state. The specified node is active.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The RGs on the node change to the UNMANAGED state.
- The cluster services stop on the specified node.
- The cluster services continue to run on other nodes.
- All RGs remain in the same state.

Network tests for an IP network

The Cluster Test Tool requires two IP networks to run any of the tests that are described in this section. The second network provides network connectivity for the tool to run. The Cluster Test Tool verifies that two IP networks are configured before running the test.

- ▶ The following command brings up a specified network on a specified node:

`NETWORK_UP_LOCAL, node, network, comments`

The command options are as follows:

`node` The name of a node on which to start the network.

`network` The name of the network to which the interface is connected.

`comments` User-defined text to describe the configured test.

- Example:

`NETWORK_UP_LOCAL, node6, hanet1, Bring up hanet1 on node 6`

- Entrance criteria

The specified node is active and has at least one inactive interface on the specified network.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services continue to run on the cluster nodes where they were active before the test.
- RGs that are in the ERROR state on the specified node and that have a service IP label available on the network can go online, but should not enter the ERROR state.
- RGs on other nodes remain in the same state.

- ▶ The following command brings down a specified network on a specified node:

`NETWORK_DOWN_LOCAL, node, network, comments`

The command options are as follows:

<i>node</i>	The name of a node on which to bring down the network.
<i>network</i>	The name of the network to which the interface is connected.
<i>comments</i>	User-defined text to describe the configured test.

Important: If one IP network is already unavailable on a node, the cluster might become partitioned. The Cluster Test Tool does not account for this situation when determining success.

- Entrance criteria

The specified node is active and has at least one active interface on the specified network.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services continue to run on the cluster nodes where they were active before the test.
- RGs on other nodes remain in the same state, but some might be hosted on a different node.
- If the node hosts an RG for which the recovery method is set to notify, the RG does not move.

- ▶ The following command brings up the specified network on all nodes that have interfaces on the network. The specified network can be an IP network or a serial network.

`NETWORK_UP_GLOBAL, network, comments`

The command options are as follows:

<i>network</i>	The name of the network to which the interface is connected.
<i>comments</i>	User-defined text to describe the configured test.

- Example

`NETWORK_UP_GLOBAL, hanet1, Bring up hanet1 on node 6`

- Entrance criteria

The specified network is active on at least one node.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services continue to run on the cluster nodes where they were active before the test.
- RGs that are in the ERROR state on the specified node and that have a service IP label that is available on the network can go online, but should not enter the ERROR state.
- RGs on other nodes remain in the same state.

- ▶ The following command brings down the specified network on all nodes that have interfaces on the network. The network that is specified can be an IP network or a serial network.

`NETWORK_DOWN_GLOBAL, network, comments`

The command options are as follows:

network The name of the network to which the interface is connected.

comments User-defined text to describe the configured test.

- Example

`NETWORK_DOWN_GLOBAL, hanet1, Bring down hanet1 on node 6`

- Entrance criteria

The specified network is inactive on at least one node.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services continue to run on the cluster nodes where they were active before the test.
- RGs on other nodes remain in the same state.

Network interface tests for IP networks

These tests bring up or down the network interfaces on an IP network:

- ▶ The following command brings up a network interface that is associated with the specified IP label on a specified node:

`JOIN_LABEL iplabel, comments`

The command options are as follows:

iplabel The IP label of the interface

comments User-defined text to describe the configured test

The only time that you can have an RG online and the service label hosted on an inactive interface is when the service interface fails but there was no place to move the RG, in which case it stays online.

- Example

`JOIN_LABEL, app_serv_address, Bring up app_serv_address on node 2`

- Entrance criteria

The specified interface is active on the specified node.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The specified interface comes up on the specified node.
- The cluster services continue to run on the cluster nodes where they were active before the test.

- RGs that are in the ERROR state on the specified node and that have a service IP label that is available on the network can go online, but should not enter the ERROR state.
 - RGs on other nodes remain in the same state.
- The following command brings down a network interface that is associated with a specified label on a specified node:

`FAIL_LABEL, iplabel, comments`

The command options are as follows:

<i>iplabel</i>	The IP label of the interface
<i>comments</i>	User-defined text to describe the configured test

– Example

`FAIL_LABEL, app_serv_label, Bring down app_serv_label, on node 2`

– Entrance criteria

The specified interface is currently inactive on the specified node.

– Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- Any service labels that were hosted by the interface are recovered.
- RGs that are in the ERROR state on the specified node and that have a service IP label that is available in the network can go online, but should not enter the ERROR state.
- RGs remain in the same state; however, the RG can be hosted by another node.

Network tests for a non-IP network

The testing for non-IP networks is part of the following test procedures:

- `NETWORK_UP_GLOBAL`
- `NETWORK_DOWN_GLOBAL`
- `NETWORK_UP_LOCAL`
- `NETWORK_DOWN_LOCAL`

Resource group tests

These tests are for RGs:

- The following command brings an RG online in a running cluster:

`RG_ONLINE, rg, node | ALL | ANY | RESTORE, comments`

The command options are as follows:

<i>rg</i>	The name of the RG to bring online.
<i>node</i>	The name of the node where the RG comes online.
<i>ALL</i>	Use ALL for concurrent RGs <i>only</i> . When ALL is specified, the RG is brought online on all nodes in the RG. If you use ALL for non-concurrent groups, the Test Tool interprets it as ANY.
<i>ANY</i>	Use ANY for non-concurrent RGs to pick a node where the RG is offline. For concurrent RGs, use ANY to pick a random node where the RG will be brought online.

RESTORE	Use RESTORE for non-concurrent RGs to bring the RGs online on the highest priority available node. For concurrent RGs, the RG is brought online on all nodes in the nodelist.
<i>comments</i>	User-defined text to describe the configured test.
– Example	<code>RG_ONLINE, rg_1, node2, Bring rg_1 online on node 2.</code>
– Entrance criteria	The specified RG is offline, there are available resources, and the RG can meet all dependencies.
– Success indicators	The following conditions indicate success for this test: <ul style="list-style-type: none"> • The cluster becomes stable. • The RG is brought online successfully on the specified node. • No RGs go offline or into the ERROR state.
► The following command brings an RG offline in a running cluster:	
	<code>RG_OFFLINE, rg, node ALL ANY, comments</code>
	The command options are as follows:
<i>rg</i>	The name of the RG to bring online.
<i>node</i>	The name of the node where the RG comes online.
ALL	Use ALL for concurrent RGs <i>only</i> . When ALL is specified, the RG is brought online on all nodes in the RG. If you use ALL for non-concurrent groups, the Test Tool interprets it as ANY.
ANY	Use ANY for non-concurrent RGs to pick a node where the RG is offline. For concurrent RGs, use ANY to pick a random node where the RG will be brought online.
<i>comments</i>	User-defined text to describe the configured test.
– Example	<code>RG_OFFLINE, rg_1, node2, Bring rg_1 offline from node2</code>
– Entrance criteria	The specified RG is online on the specified node.
– Success indicators	The following conditions indicate success for this test: <ul style="list-style-type: none"> • The cluster becomes stable. • The RG, which was online on the specified node, is brought offline successfully. • Other RGs remain in the same state.
► The following command moves an RG that is already online in a running cluster to a specific or any available node:	
	<code>RG_MOVE, rg, node ANY RESTORE, comments</code>
	The command options are as follows:
<i>rg</i>	The name of the RG to bring offline.
<i>node</i>	The target node, that is, the name of the node to which the RG moves.

ANY	Use ANY to let the Cluster Test Tool pick a random available node to which to move the RG.
RESTORE	Enable the RG to move to the highest priority node available.
<i>comments</i>	User-defined text to describe the configured test.
– Example	<code>RG_MOVE, rg_1, ANY, Move rg_1 to any available node.</code>
– Entrance criteria	The specified RG must be non-concurrent and must be online on a node other than the target node.
– Success indicators	The following conditions indicate success for this test:
	<ul style="list-style-type: none"> • The cluster becomes stable. • The RG is moved to the target node successfully. • Other RGs remain in the same state.
▶	The following command moves an RG that is already online in a running cluster to an available node at a specific site:
	<code>RG_MOVE_SITE, rg, site OTHER, comments</code>
	The command options are as follows:
<i>rg</i>	The name of the RG to bring offline.
<i>site</i>	The site where the RG moves.
OTHER	Use OTHER to have the Cluster Test Tool pick the <i>other</i> site as the RG destination. For example, if the RG is online on siteA, it moves to siteB, and conversely if the RG is online on siteB, it moves to siteA.
<i>comments</i>	User-defined text to describe the configured test.
– Example	<code>RG_MOVE_SITE, rg_1, site_2, Move rg_1 to site_2.</code>
– Entrance criteria	The specified RG is online on a node, other than the node in the target site.
– Success indicators	The following conditions indicate success for this test:
	<ul style="list-style-type: none"> • The cluster becomes stable. • The RG is moved to the target site successfully. • Other RGs remain in the same state.

Volume group test

This test is for the VGs. The following command forces an error for a disk that contains a VG in an RG:

```
VG_DOWN, vg, node | ALL | ANY, comments
```

The command options are as follows:

<i>vg</i>	The VG on the disk to fail.
<i>node</i>	The name of the node where the RG that contains the specified VG is online.
ALL	Use ALL for concurrent RGs. When ALL is specified, the Cluster Test Tool fails the VG on all nodes in the RG where the RG is online. If ALL is used for non-concurrent RGs, the tool performs this test for any RG.
ANY	Use ANY to have the Cluster Test Tool select the node as follows: For a non-concurrent RG, the Cluster Test Tool selects the node where the RG is online. For a concurrent RG, the Cluster Test Tool selects a random node from the concurrent RG node list, where the RG is online.
<i>comments</i>	User-defined text to describe the configured test.

► Example

```
VG_DOWN, sharedvg, ANY, Fail the disk where sharedvg resides
```

► Entrance criteria

The RG containing the specified VGs is online on the specified node.

► Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The RG containing the specified VG successfully moves to another node, or if it is a concurrent RGs, it goes into an ERROR state.
- RGs can change state to meet dependencies.

Site tests

These tests are for the site:

► The following command fails all the XD_data networks, causing the site_isolation event:

```
SITE_ISOLATION, comments
```

The command options are as follows:

<i>comments</i>	User-defined text to describe the configured test.
-----------------	--

– Example

```
SITE_ISOLATION, Fail all the XD_data networks
```

– Entrance criteria

At least one XD_data network is configured and is up on any node in the cluster.

– Success indicators

The following conditions indicate success for this test:

- The XD_data network fails, and no RGs change state.
- The cluster becomes stable.

- ▶ The following command runs when at least one XD_data network is up to restore connections between the sites, and remove site isolation. Run this test after running the **SITE_ISOLATION** test.

SITE_MERGE, *comments*

The command option is as follows:

comments User-defined text to describe the configured test.

- Example

SITE_MERGE, Heal the XD_data networks

- Entrance criteria

At least one node must be online.

- Success indicators

The following conditions indicate success for this test:

- No RGs change state.
- The cluster becomes stable.

- ▶ The following command stops the cluster services and moves the RGs to other nodes on all nodes at the specified site:

SITE_DOWN_TAKEOVER, *site*, *comments*:

The command options are as follows:

site The site that contains the nodes on which cluster services will be stopped.

comments User-defined text to describe the configured test.

- Example

SITE_DOWN_TAKEOVER, site_1, Stop cluster services on all nodes at site_1, bringing the resource groups offline and moving the resource groups.

- Entrance criteria

At least one node at the site must be online.

- Success indicators

The following conditions indicate success for this test:

- The cluster services are stopped on all nodes at the specified site. All primary instance RGs move to another site.
- All secondary instance RGs go offline.
- The cluster becomes stable.

- ▶ The following command starts the cluster services on all nodes at the specified site:

SITE_UP, *site*, *comments*

The command options are as follows:

site The site that contains the nodes on which cluster services will be started.

comments User-defined text to describe the configured test.

- Example
SITE_UP, site_1, Start cluster services on all nodes at site_1.
- Entrance criteria
At least one node at the site must be offline.
- Success indicators
The following conditions indicate success for this test:
 - The cluster services are started on all nodes at the specified site.
 - The RGs remain in the same state.
 - The cluster becomes stable.

General tests

Other tests that are available to use in PowerHA cluster testing are as follows:

- ▶ Bring down an application server.
- ▶ Terminate the Cluster Manager on a node.
- ▶ Add a wait time for test processing.

The commands for these tests are listed here:

- ▶ The following command runs the specified command to stop an application server. This test is useful when testing application availability. In the automated test, the test uses the stop script to turn off the application.

`SERVER_DOWN, node | ANY, appserv, command, comments`

The command options are as follows:

<i>node</i>	The name of a node on which the specified application sever will become unavailable.
<i>ANY</i>	Any available node that participates in this RG can have the application server become unavailable. The Cluster Test Tool tries to simulate server failure on any available cluster node. This test is equivalent to failure on the node that owns the RG, if the server is in an RG that has policies other than Startup (Online on all available nodes) or Failover (Bring offline on error node only).
<i>appserv</i>	The name of the application server that is associated with the specified node.
<i>command</i>	The command to run to stop the application server.
<i>comments</i>	User-defined text to describe the configured test.

- Example
`SERVER_DOWN,node1,db_app /apps/stop_db.p1, Kill the db app`
- Entrance criteria

The RG is online on the specified node.

- Success indicators
The following conditions indicate success for this test:
 - The cluster becomes stable.
 - The cluster nodes remain in the same state.

- The RG that contains the application server is online; however, the RG can be hosted by another node, unless it is a concurrent RG, in which case the group goes into the ERROR state.
- ▶ The following command runs the **kill** command to terminate the Cluster Manager on a specified node:

CLSTRMGR_KILL, *node*, *comments*

The command options are as follows:

<i>node</i>	The name of the node on which to terminate the Cluster Manager.
<i>comments</i>	User-defined text to describe the configured test.

Note: If **CLSTRMGR_KILL** is run on the local node, you might need to restart the node. On startup, the Cluster Test Tool automatically starts again. You can avoid manual intervention to restart the control node during testing by doing these tasks:

- ▶ Editing the /etc/cluster/hacmp.term file to change the default action after an abnormal exit. The **clexit.rc** script checks for the presence of this file and, if the file is executable, the script calls it instead of halting the system automatically.
- ▶ Configuring the node to automatic initial program load (IPL) before running the Cluster Test Tool (it stops).

For the Cluster Test Tool to accurately assess the success or failure of a **CLSTRMGR_KILL** test, do not do other activities in the cluster while the Cluster Test Tool is running.

- Example

CLSTRMGR_KILL, *node5*, Bring down *node5* hard

- Entrance criteria

The specified node is active.

- Success indicators

The following conditions indicate success for this test:

- The cluster becomes stable.
- The cluster services stop on the specified node.
- The cluster services continue to run on other nodes.
- The RGs that were online on the node where the Cluster Manager fails move to other nodes.
- All RGs on other nodes remain in the same state.

- ▶ The following command generates a wait period for the Cluster Test Tool for a specified number of seconds:

WAIT, *seconds*, *comments*

The command options are as follows:

<i>seconds</i>	The number of seconds that the Cluster Test Tool waits before proceeding with processing.
<i>comments</i>	User-defined text to describe the configured test.

- Example
WAIT, 300, We need to wait for five minutes before the next test
- Entrance criteria
Not applicable
- Success indicators
Not applicable

Example test plan

The excerpt in Example 6-14 is from a sample test plan and includes the following tests:

- ▶ **NODE_UP**
- ▶ **NODE_DOWN_GRACEFUL**

It also includes a WAIT interval. The comment text at the end of the line describes the action that the test will do.

Example 6-14 Excerpt from a sample test plan

```
NODE_UP,ALL,starts cluster services on all nodes
NODE_DOWN_GRACEFUL,brianna,stops cluster services gracefully on node brianna
WAIT,20
NODE_UP,brianna,starts cluster services on node waltham
```

Running a custom test procedure

Before you start running custom tests, make sure that these factors are met:

- ▶ Your test plan is configured correctly. For more information about setting up a test plan, see “Creating a custom test procedure” on page 232.
- ▶ You specified values for the test parameters.
- ▶ You configured logging for the tool to capture information that you want to examine for your cluster.
- ▶ The cluster is not in service in a production environment.

To run custom testing, complete the following steps:

1. Run **smitty sysmirror**, and then select **Problem Determination Tools** → **Cluster Test Tool** → **Execute Custom Test Procedure**.
2. In the Execute Custom Test Procedure panel (Figure 6-15 on page 248), enter the following field values:

Test Plan	This <i>required</i> field contains the full path to the test plan file for the Cluster Test Tool. This file specifies the tests for the tool to run.
Variables File	Optional. This field contains the full path to the variables file for the Cluster Test Tool. This file specifies the variable definitions that are used in processing the test plan.
Verbose Logging	When set to Yes (default), the log file includes extra information that might help you to judge the success or failure of some tests. Select No to decrease the amount of information that is logged by the Cluster Test Tool.

Cycle Log File	When set to Yes (default), uses a new log file to store output from the Cluster Test Tool. Select No to append messages to the current log file.
Abort On Error	When set to No (default), the Cluster Test Tool continues to run tests after some of the tests fail. This setting might cause subsequent tests to fail because the cluster state differs from the state that is expected by one of those tests. Select Yes to stop processing after the first test fails.

Note: The tool stops running and issues an error if a test fails and Abort On Error is set to Yes.

3. Press Enter to start running the custom tests.
4. Evaluate the test results.

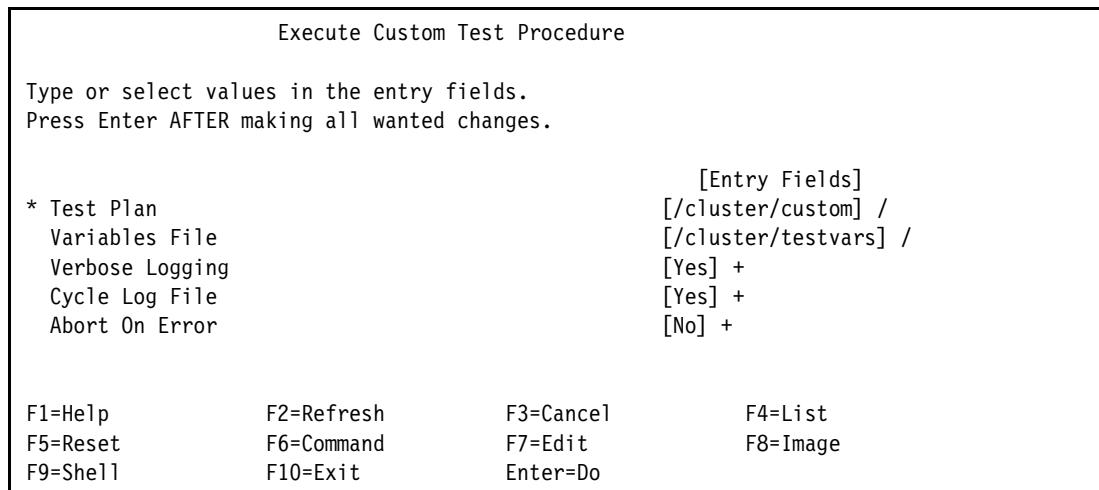


Figure 6-15 Custom test SMIT menu

Important: If you uninstall PowerHA, the program removes any files that you might have customized for the Cluster Test Tool. If you want to retain these files, copy them before you uninstall PowerHA.

Log files

If a test fails, the Cluster Test Tool collects information in the automatically created log files. You evaluate the success or failure of tests by reviewing the contents of the Cluster Test Tool log file `/var/hacmp/log/cl_testtoo1.log`. PowerHA never deletes the files in this directory.

For each test plan that has any failures, the tool creates a new directory under `/var/hacmp/log/`. If the test plan has no failures, the tool does not create a log directory. The directory name is unique and consists of the name of the Cluster Test Tool plan file, and the timestamp when the test plan was run.

Note: Detailed output from an automated cluster test is shown in Appendix B, “Cluster Test Tool log” on page 621.

Log file rotation

The Cluster Test Tool saves up to three log files and numbers them so that you can compare the results of different cluster tests:

```
/var/hacmp/log/cl_testtool.log  
/var/hacmp/log/cl_testtool.log.1  
/var/hacmp/log/cl_testtool.log.2
```

The tool also rotates the files, where the oldest file is overwritten. If you do not want the tool to rotate the log files, you can disable this feature from SMIT.



Cluster management

In this chapter, we describe PowerHA cluster management and administration, including helpful tips when you use these features.

This chapter contains the following topics:

- ▶ Cluster Single Point of Control
- ▶ File collections
- ▶ User administration
- ▶ Shared storage management
- ▶ Time synchronization
- ▶ Cluster verification and synchronization
- ▶ Monitoring PowerHA

7.1 Cluster Single Point of Control

Cluster Single Point of Control (C-SPOC) is a useful tool to help you manage the entire cluster from any single point. It provides facilities for performing common cluster-wide administration tasks from any active node within the cluster. The downtime that is caused by cluster administration is reduced by using C-SPOC.

Highly available environments require special consideration when you plan changes to the environment. Ensure that you follow a strict change management discipline.

Before we describe cluster management in more detail, we emphasize the following general best practices for cluster administration:

- ▶ Where possible, use the PowerHA C-SPOC facility when you make changes to the cluster.
- ▶ Document routine operational procedures (for example, shutdown, startup, and increasing the size of a file system).
- ▶ Restrict access to the root password to trained PowerHA administrators. Use the SystemMirror User Interface (SMUI) and create users that may perform specific tasks. For a demonstration of this process, see this [YouTube video](#).
- ▶ Always take a snapshot of your existing configuration before making any changes. If you perform live changes (known as dynamic automatic reconfiguration (DARE)), a cluster snapshot of the active configuration is automatically created.
- ▶ Monitor your cluster regularly. For more information about monitoring, see 7.7, “Monitoring PowerHA” on page 312.

The C-SPOC function is provided by its own set of cluster administration commands, which are accessible through System Management Interface Tool (SMIT) menus. The commands are in the /usr/es/sbin/cluster/cspoc directory. C-SPOC uses the Cluster Communications daemon (**c1comdES**) to run commands on remote nodes. If this daemon is not running, the command might not be run and the C-SPOC operation might fail.

Note: After PowerHA is installed, **c1strmgrES** is started from `inittab`, so it is always running whether cluster services are started or not.

C-SPOC operations fail if any target node is down at the time of execution or if the selected resource is not available. C-SPOC requires a correctly configured cluster in the sense that all nodes within the cluster can communicate.

If node failure occurs during a C-SPOC operation, an error is displayed on the SMIT panel and the error output is recorded in the C-SPOC log file (`cspoc.log`). Check this log if any C-SPOC problem occurs. For more information about PowerHA logs, see 7.7.5, “Log files” on page 324.

7.1.1 The C-SPOC SMIT menu

C-SPOC SMIT menus are accessible by running `smitty sysmirror` and then selecting System Management (C-SPOC), or by using the fast path `smitty cspoc`. The main C-SPOC functions or submenus are listed as they are listed in the SMIT C-SPOC menu, in the same order as they are listed in the main C-SPOC menu:

- ▶ Storage
This option contains utilities to assist with the cluster-wide administration of shared volume groups (VGs), logical volumes (LVs), file systems, physical volumes (PVs), and mirror pools. For more information about this topic, see 7.4.4, “C-SPOC Storage menu” on page 282.
- ▶ PowerHA SystemMirror Services
This option contains utilities to start and stop cluster services on selected nodes and also the function to show running cluster services on the local node. For more information, see 6.2, “Starting and stopping the cluster” on page 197 and “Checking the cluster subsystem status” on page 316.
- ▶ Communication Interfaces
This option contains utilities to manage the configuration of communication interfaces to AIX and update PowerHA with these settings.
- ▶ Resource Groups and Applications
This option contains utilities to manipulate resource groups (RGs) in addition to application monitoring and application availability measurement tools. For more information about application monitoring, see 7.7.9, “Application monitoring” on page 333. For more information about the application availability analysis tool, see 7.7.10, “Measuring application availability” on page 344.
- ▶ PowerHA SystemMirror logs
This option contains utilities to display the contents of some log files and change the debug level and format of log files (standard HTML). You can also change the location of cluster log files in this menu. For more information about these topics, see 7.7.5, “Log files” on page 324.
- ▶ File Collections
This option contains utilities to assist with file synchronization throughout the cluster. A *file collection* is a user-defined set of files. For more information about file collections, see 7.2, “File collections” on page 254.
- ▶ Security and Users
This option contains menus and utilities for various security settings and also users, groups, and password management within a cluster. For more information about security, see 8.1, “Cluster security” on page 348. For more information about user management, see 7.3, “User administration” on page 261.
- ▶ Lightweight Directory Access Protocol (LDAP)
This option is used to configure an LDAP server and client for a PowerHA SystemMirror cluster environment. This LDAP is used as a central repository for implementing most of the security features.
- ▶ Open a SMIT Session on a Node
Use this option to run a remote SMIT session from another node in the cluster. However, you use the standard AIX default SMIT menu and not a PowerHA or C-SPOC specific menu.

7.2 File collections

PowerHA provides cluster-wide file synchronization capabilities by using C-SPOC file collections. A *file collection* is a user-defined set of files. You can add files to a file collection or remove files from it, and you can specify the frequency at which PowerHA synchronizes these files.

PowerHA provides three ways to propagate your files:

- ▶ Manually: You can synchronize your files manually at any time. The files on the local system are copied from the local node to the remote nodes in the cluster.
- ▶ Automatically during cluster verification and synchronization: The files are copied from the local node where you initiate the verification operation.
- ▶ Automatically when changes are detected: PowerHA periodically checks the file collection on all nodes. If a file changed, it synchronizes this file across the cluster. You can set a timer to determine how frequently PowerHA checks your file collections.

PowerHA retains the permissions, ownership, and timestamp of the file on the local node and propagates them to the remote nodes. You can specify ordinary files for a file collection. You can also specify a directory and wildcard file names. You cannot add the following items:

- ▶ Symbolic links
- ▶ Wildcard directory names
- ▶ Pipes
- ▶ Sockets
- ▶ Device files (/dev/*)
- ▶ Files from the /proc directory
- ▶ Object Data Manager (ODM) files from /etc/objrepos/* and /etc/es/objrepos/*

Always use full path names. Each file can be added to only one file collection, except those files that are automatically added to the HACMP_Files collection. The files should not exist on the remote nodes, and PowerHA creates them during the first synchronization. Any zero length or nonexistent files are not propagated from the local node.

PowerHA creates a backup copy of the modified files during synchronization on all nodes. These backups are stored in the /var/hacmp/filebackup directory. Only one previous version is retained, and you can only manually restore them.

The file collection logs are stored in the /var/hacmp/log/clutils.log file.

Important: You are responsible for ensuring that the files on the local node (where you start the propagation) are the most recent and not corrupted.

7.2.1 Predefined file collections

PowerHA provides two file default collections: Configuration_Files and HACMP_Files. Although neither file is set up for automatic synchronization, you can enable them by setting either of the following options to Yes in the SMIT “Change/Show a file collection” menu:

- ▶ Propagate files during cluster synchronization
- ▶ Propagate files automatically when changes are detected

For more information, see “Modifying a file collection” on page 258.

Configuration_Files

This collection contains the essential AIX configuration files:

- ▶ /etc/hosts
- ▶ /etc/services
- ▶ /etc/snmpd.conf
- ▶ /etc/snmpdv3.conf
- ▶ /etc/rc.net
- ▶ /etc/inetd.conf
- ▶ /usr/es/sbin/cluster/netmon.cf
- ▶ /usr/es/sbin/cluster/etc/clhosts
- ▶ /usr/es/sbin/cluster/etc/rhosts
- ▶ /usr/es/sbin/cluster/etc/clinfo.rc

You can add to or remove files from the file collections. For more information, see “Adding files to a file collection” on page 259.

HACMP_Files

If you add any of the following user-defined files to your cluster configuration, then they are automatically included in the HACMP_Files file collection:

- ▶ Application server start script
- ▶ Application server stop script
- ▶ Event notify script
- ▶ Pre-event script
- ▶ Post-event script
- ▶ Event error recovery script
- ▶ Application monitor notify script
- ▶ Application monitor cleanup script
- ▶ Application monitor restart script
- ▶ Pager text message file
- ▶ High availability (HA) Tape support start script
- ▶ HA Tape support stop script
- ▶ User-defined event recovery program
- ▶ Custom snapshot method script

For an example of how this inclusion works, our cluster has an application server, app_server_1, which has the following three files:

- ▶ A start script: /usr/app_scripts/app_start
- ▶ A stop script: /usr/app_scripts/app_stop
- ▶ A custom post-event script to the PowerHA node_up event:
/usr/app_scripts/post_node_up

These three files were automatically added to the HACMP_Files file collection when we defined them during the PowerHA configuration. You can check that they were included by completing the following steps:

1. Start file collection management by running `smitty cm_filecollection_mgt`, selecting **Change>Show a File Collection** → **HACMP_Files** from the list, and pressing Enter.
2. Select the Collection files field and press F4 (List). Example 7-1 shows that the application start and stop scripts and the post-event command are automatically added to this file collection.

Note: You cannot manually add or remove files from this file collection. Also, the collection cannot be renamed. When using the HACMP_Files collection, make sure that all the scripts work as designed on all nodes.

If you do not want to synchronize all your user-defined scripts or if they are not the same on all nodes, then disable this file collection and create another one, which includes only the required files.

Example 7-1 How to list which files are included in an existing file collection

Change/Show a File Collection

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

File Collection Name New File Collection Name File Collection Description +----- Collection files	<p>[Entry Fields]</p> <p>HACMP_Files [] [User-defined scripts ></p> <p>The value for this entry field must be in the range shown below. Press Enter or Cancel to return to the entry field, and enter the wanted value.</p> <p>/tmp/app_scripts/app_start /tmp/app_scripts/app_stop</p> <p>F1=F1=Help F2=Refresh F3=Cancel F8=Image F10=Exit Enter=Do F5=/Find n=Find Next F9+-----</p>
---	---

7.2.2 Managing file collections

You can create, modify, or remove a file collection.

Creating a file collection

To add a file collection, complete the following steps:

1. Start SMIT, run `smitty cspoc`, and select **File Collections**, or you can start File Collections directly by using the following command:
`smitty cm_filecollection_menu`
2. Select **Manage File Collections → Add a File Collection**.
3. Supply the following information (see Figure 7-1):
 - File Collection Name: A unique name for the file collection.
 - File Collection Description: A short description of this file collection.
 - Propagate files during cluster synchronization?: When set to Yes, then PowerHA propagates this file collection during cluster synchronization. This solution is convenient for cluster-related files, for example, your application start and stop scripts are automatically synchronized after you change the cluster configuration.
 - Propagate files automatically when changes are detected?: If you select Yes, PowerHA checks the files in this collection regularly, and if any of them are changed, then it repropagates them.

If both “Propagate files” options are kept as No, no automatic synchronization occurs.

Add a File Collection

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

<p>* File Collection Name</p> <p>File Collection Description</p> <p>Propagate files during cluster synchronization?</p> <p>Propagate files automatically when changes are detected?</p>	<p>[Entry Fields]</p> <p>[application_files]</p> <p>[Application config fi></p> <p>yes +</p> <p>no +</p>		
<p>F1=Help F5=Reset F9=Shell</p>	<p>F2=Refresh F6=Command F10=Exit</p>	<p>F3=Cancel F7>Edit</p>	<p>F4>List F8=Image Enter=Do</p>

Figure 7-1 Add a File Collection

Modifying a file collection

To change a file collection, complete the following steps:

1. Start PowerHA File Collection Management by entering `smitty cm_filecollection_menu` and selecting **Manage File Collections** → **Change/Show a File Collections**.
2. Select a file collection from the list.
3. Now, you can change the following information (for more information, see Figure 7-2 and “Creating a file collection” on page 257):
 - File Collection Name.
 - File Collection Description.
 - Propagate files during cluster synchronization (Yes or No).
 - Propagate files automatically when changes are detected (Yes or No).
 - Collection files: Press F4 here to see the list of files in this collection.

Change/Show a File Collection			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
File Collection Name New File Collection Name	[Entry Fields] Configuration_Files []		
File Collection Description	[AIX and HACMP configu>		
Propagate files during cluster synchronization?	no +		
Propagate files automatically when changes are detected?	no +		
Collection files	+		
F1=Help F5=Reset F9=Shell	F2=Refresh F6=Command F10=Exit	F3=Cancel F7>Edit Enter=Do	F4>List F8=Image

Figure 7-2 Changing a file collection

Removing a file collection

To remove a file collection, complete the following steps:

1. Start PowerHA File Collection Management by entering `smitty cm_filecollection_menu` and then selecting **Manage File Collections** → **Remove a File Collection**.
2. Select a file collection from the list.
3. Press Enter to confirm the deletion of the file collection.

Changing the automatic update timer

Here you can set the timer for how frequently PowerHA checks the files in the collection for changes. Only one timer can be set for all file collections in the cluster. Complete the following steps:

1. Start PowerHA File Collection Management by entering `smitty cm_filecollection_menu` and selecting **Manage File Collections** → **Change/Show Automatic Update Time**.
2. Supply a value (in minutes) for Automatic File Update Time. The value should be in the range of 10 - 1440 minutes (24 hours).

Adding files to a file collection

To add files to a file collection, complete the following steps:

1. Start PowerHA File Collection Management by entering `smitty cm_filecollection_menu` and selecting **Manage Files in File Collections** → **Add Files to a File Collection**.
2. Select a file collection from the list and press Enter.
3. On the SMIT panel, you can check the current file list or add new files (Figure 7-3):
 - To see a list of current files in this collection, press F4 in the Collection Files field.
 - To add new files, go to the New File field and type the file name that you want to add to the file collection. You can add *only* one file at a time. The file name must start with the forward slash (/). You can specify only ordinary files. You cannot add symbolic links, directory, pipe, socket, or device files (/dev/*), files from /proc directory, or ODM files from /etc/objrepos/* and /etc/es/objrepos/*.

Add Files to a File Collection

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

		[Entry Fields]	
File Collection Name	app_files		
File Collection Description	Application configura>		
Propagate files during cluster synchronization?	no		
Propagate files automatically when changes are detected?	no		
Collection files			
* New File	[/usr/app/config_file]/		
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

Figure 7-3 Add Files to a File Collection

Important: You cannot add files to the HACMP_Files collection.

Removing files from a file collection

To remove files from a file collection, complete the following steps:

1. Start PowerHA File Collection Management by entering `smitty cm_filecollection_menu` and selecting **Manage Files in File Collections** → **Remove Files from a File Collection**.
2. Select a file collection from the list and press Enter.
3. Select one or more files from the list and press Enter (see Figure 7-4).
4. Press Enter again to confirm.

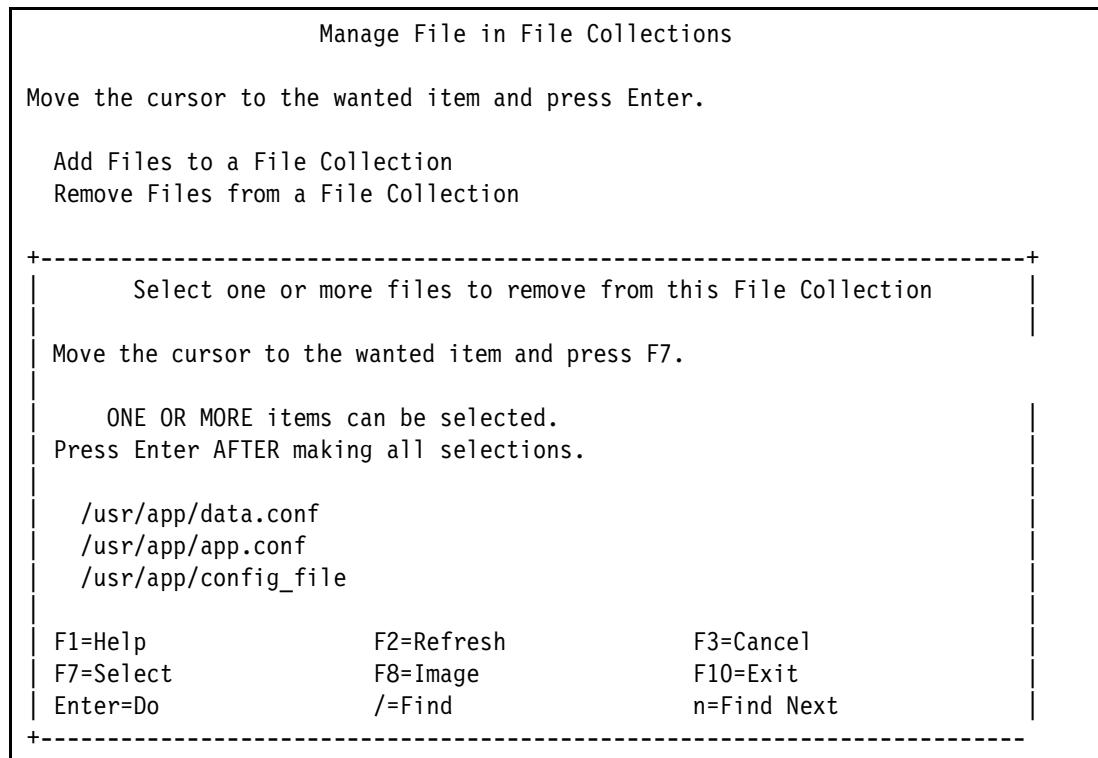


Figure 7-4 Removing files from a file collection

Important: You cannot remove files from the HACMP_Files collection.

Manually propagating files in a file collection

You can manually synchronize file collections (see Figure 7-5 on page 261) by completing the following steps:

1. Start PowerHA File Collection Management by entering `smitty cm_filecollection_menu` and selecting Propagate Files in File Collections.
2. Select a file collection from the list and press Enter.
3. Press Enter again to confirm.

```

COMMAND STATUS

Command: OK          stdout: yes          stderr: no

Before command completion, additional instructions may appear below.

The following file collections will be processed:
app_files
Starting file propagation to remote node buttercup.
Successfully propagated file /usr/app/data.conf to node buttercup.
Successfully propagated file /usr/app/app.conf to node buttercup.
Successfully propagated file /usr/app/config_file to node buttercup.
Total number of files that are propagated to node buttercup: 3

F1=Help          F2=Refresh        F3=Cancel        F6=Command
F8=Image          F9=Shell          F10=Exit         /=Find
n=Find Next

```

Figure 7-5 Manual propagation of a file collection

7.3 User administration

In a PowerHA cluster, user IDs and passwords should be synchronized. If user and group IDs are not the same across your cluster, your application might not work, and users cannot access their files on the shared storage. As a best practice, synchronize passwords so that if there is a failover, users can log in without the delay of having their password reset if they do not know what it is on the failover node.

Here are a couple options to consider for user and password synchronization:

- ▶ Using C-SPOC: PowerHA provides utilities in C-SPOC for user administration. For more information, see 7.3.1, “C-SPOC user and group administration” on page 261.
- ▶ LDAP is the best solution for managing many users in a complex environment. LDAP can be set up to work together with PowerHA. For more information about LDAP, see *Understanding LDAP - Design and Implementation*, SG24-4986.

Note: PowerHA C-SPOC provides SMIT panels for configuring both LDAP servers and clients. The fast path is **smitty cl_ldap**. However, we do not provide more details about that topic.

7.3.1 C-SPOC user and group administration

PowerHA provides C-SPOC tools for cluster-wide user, group, and password administration. The following functions are available with C-SPOC:

- ▶ Add users.
- ▶ List users.
- ▶ Change user attributes.
- ▶ Remove users.
- ▶ Add groups.
- ▶ List groups.
- ▶ Change group attributes.

- ▶ Remove groups.
- ▶ Change user passwords cluster-wide.
- ▶ Manage the list of users that are permitted to change their passwords cluster-wide.

Adding a user

To add a user on all nodes in the cluster, complete the following steps:

1. Start SMIT and run **smitty cspoc**, and then select **Security and Users**, or you can use the fast path by entering **smitty cl_usergroup**.
2. Select Users in a PowerHA SystemMirror cluster.
3. Select Add a User to the Cluster.
4. Select either LOCAL(FILES) or LDAP.
5. Select an RG in which you want to create users. If you leave the Select Nodes by Resource Group field empty, the user is created on *all* nodes in the cluster. If you select an RG here, the user is created only on the subset of nodes on which that RG is configured to run. In a two-node cluster, leave this field blank.

If you have more than two nodes in your cluster, you can create users that are related to specific RGs. If you want to create a user for these nodes only (for example, a user may log in to jordan and jessica, but the user may not log in to harper or athena), select the appropriate RG name from the list (see Figure 7-6).

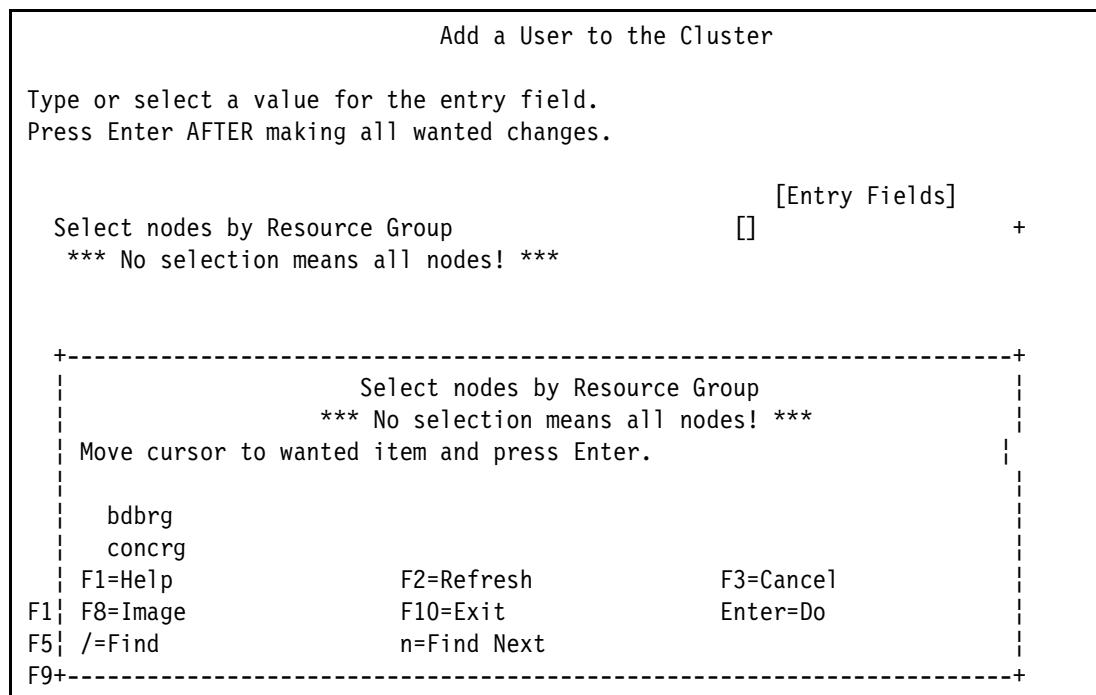


Figure 7-6 Selecting nodes by resource group

6. Create the user. Supply the username and other relevant information as you do when creating any user. You can specify the user ID here, but if the user ID is already on a node, the command fails. If you leave the User ID field blank, the user is created with the first available ID on all nodes (see Figure 7-7 on page 263).

Add a User to the Cluster		
Type or select values in the entry fields.		
Press Enter AFTER making all wanted changes.		
[TOP]	[Entry Fields]	
Select nodes by resource group		
*** No selection means all nodes! ***		
* User NAME	[sbodily]	
User ID	[249] #	
ADMINISTRATIVE USER?	false +	
Primary GROUP	[system] +	
Group SET	[staff] +	
ADMINISTRATIVE GROUPS	[] +	
Another user can SU TO USER?	true +	
SU GROUPS	[ALL] +	
HOME directory	[/home/sbodily]	
Initial PROGRAM	[]	
User INFORMATION	[Mr. Bodily]	
EXPIRATION date (MMDDhhmmYY)	[0]	
Is this user ACCOUNT LOCKED?	false +	
User can LOGIN?	true +	
User can LOGIN REMOTELY?	true +	
Allowed LOGIN TIMES	[]	
Number of FAILED LOGINS before user account is locked	[0] #	
Login AUTHENTICATION GRAMMAR	[compat]	
Valid TTYS	[ALL]	
Days to WARN USER before password expires	[0] #	
Password CHECK METHODS	[]	
Password DICTIONARY FILES	[]	
NUMBER OF PASSWORDS before reuse	[0] #	
WEEKS before password reuse	[0] #	
Weeks between password EXPIRATION and LOCKOUT	[-1]	
Password MAX. AGE	[0] #	
Password MIN. AGE	[0] #	
Password MIN. LENGTH	[0] #	
Password MIN. ALPHA characters	[0] #	
Password MIN. OTHER characters	[0] #	
Password MAX. REPEATED characters	[8] #	
Password MIN. DIFFERENT characters	[0] #	
Password REGISTRY	[]	
Soft FILE size	[-1]	
Soft CPU time	[-1]	
Soft DATA segment	[262144] #	
Soft STACK size	[65536] #	
Soft CORE file size	[2097151] #	
File creation UMASK	[022]	
AUDIT classes	[] +	
TRUSTED PATH?	nosak +	
PRIMARY authentication method	[SYSTEM]	
SECONDARY authentication method	[NONE]	
Keystore Access	[] +	
Adminkeystore Access	[] +	
Initial Keystore Mode	[] +	
Allow user to change Keystore Mode?	[]	
Keystore Encryption Algorithm	[] +	
File Encryption Algorithm	[]	

Figure 7-7 Creating a user on all cluster nodes

Note: When you create a user's home directory and if it will reside on a shared file system, C-SPOC does not check whether the file system is mounted or if the VG is varied. In this case, C-SPOC creates the user home directory under the empty mount point of the shared file system. You can correct this situation by moving the home directory to under the shared file system.

If a user's home directory is on a shared file system, the user can log in only on the node where the file system is mounted.

Listing cluster users

To list users in the cluster, follow these steps:

1. Start C-SPOC Security and Users by entering the **smitty cl_usergroup** fast path.
2. Select Users in an PowerHA SystemMirror Cluster.
3. Select List Users in the Cluster.
4. Select either LOCAL(FILES) or LDAP from the list.
5. Select the nodes for which user lists you want to display. If you leave the Select Nodes by Resource Group field empty, the users for *all* cluster nodes are listed.
If you select an RG here, C-SPOC lists only users from the nodes that belong to the specified RG.
6. Press Enter (see Example 7-2).

Example 7-2 Listing users in the cluster

COMMAND STATUS		
Command: OK	stdout: yes	stderr: no
Before command completion, additional instructions may appear below.		
<pre>jordan root 0 / jordan daemon 1 /etc jordan bin 2 /bin jordan sys 3 /usr/sys jordan adm 4 /var/adm jordan sshd 207 /var/empty jordan sbodily249 /home/sbodily jordan killer 303 /home/killer jordan jerryc 305 /home/jerryc jessica root 0 / jessica daemon 1 /etc jessica bin 2 /bin jessica sys 3 /usr/sys jessica adm 4 /var/adm jessica sshd 207 /var/empty jessica sbodily249 /home/sbodily jessica killer 303 /home/killer jessica jerryc 305 /home/jerryc</pre>		
F1=Help F8=Image n=Find Next	F2=Refresh F9=Shell	F3=Cancel F10=Exit /=Find

Modifying user attributes

To modify user attributes in the cluster, complete the following steps:

1. Start C-SPOC Security and Users by entering **smitty c1_usergroup** and then selecting **Users in an PowerHA SystemMirror Cluster → Change / Show Characteristics of a User in the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes on which you want to modify a user. If you leave the field Select Nodes by Resource Group field empty, the user can be modified on *all* nodes.
If you select an RG here, you can modify a user that belongs to the specified RG.
4. Enter the name of the user that you want to modify or press F4 to select from the list.
5. Now, you can modify the user attributes (see Example 7-3).

Example 7-3 Modifying user attributes

Change / Show Characteristics of a User

Type or select values in the entry fields.

Press Enter AFTER making all wanted changes.

[TOP]	[Entry Fields]
* User NAME	killer
User ID	[303] #
ADMINISTRATIVE USER?	false +
Primary GROUP	[staff] +
Group SET	[staff] +
ADMINISTRATIVE GROUPS	[] +
ROLES	[] +
Another user can SU TO USER?	true +
SU GROUPS	[ALL] +
HOME directory	[/home/killer]
Initial PROGRAM	[/usr/bin/ksh]
User INFORMATION	[Ms Killeen]
EXPIRATION date (MMDDhhmmYY)	[0]
Is this user ACCOUNT LOCKED?	false +
[MORE...36]	

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

Removing a user

To remove a user, complete the following steps:

1. Start C-SPOC Security and Users by entering **smitty c1_usergroup**, and then selecting **Users in an PowerHA SystemMirror Cluster → Remove a User from the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes from which you want to remove a user. If you leave the Select Nodes by Resource Group field empty, any user can be removed from all nodes.
If you select an RG here, C-SPOC removes the user from only the nodes that belong to the specified RG.

4. Enter the username to remove or press F4 to select a user from the list.
5. For Remove AUTHENTICATION information, select Yes (the default) to delete the user password and other authentication information. Select No to leave the user password in the /etc/security/passwd file (see Figure 7-8).

Remove a User from the Cluster

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

[Entry Fields]

Select nodes by resource group
*** No selection means all nodes! ***

* User NAME	[killer] +		
Remove AUTHENTICATION information?	Yes +		
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

Figure 7-8 Remove a User from the Cluster

Adding a group to the cluster

To add a group to the cluster, complete the following steps:

1. Start C-SPOC Security and Users by entering **smitty c1_usergroup** and selecting **Groups in an PowerHA SystemMirror Cluster** → **Add a Group to the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes on which you want to create groups. If you leave the Select Nodes by Resource Group field empty, the group is created on *all* nodes in the cluster.

If you have more than two nodes in your cluster, you can create groups that are related to specific RGs. If you want to create a group for only these nodes, select the appropriate RG name from the list (see Table 7-1).

Table 7-1 Cross-reference of groups, resource groups, and nodes

Resource group	Nodes	Group
bdbrg	jordan and jessica	dbadmin
nonconrg	jessica and jordan	developers
conrg	harper and athena	appusers
apprg	athena, harper, jessica, and jordan	support

Table 7-1 is a cross-reference between RGs, nodes, and groups. It shows the “support” that is present on all nodes (leave the Select Nodes by Resource Group field empty), and groups such as dbadmin are created only on jordan and jessica (select bdbrg in the Select Nodes by Resource Group field).

4. Create the group (see Figure 7-9 on page 267). Supply the group name, user list, and other relevant information as when you create any normal group. Press F4 for a list of the available users to include in the group.

You can specify the group ID here. However, if the ID is already on a node, the command fails. If you leave the Group ID field blank, the group is created with the first available ID on all cluster nodes.

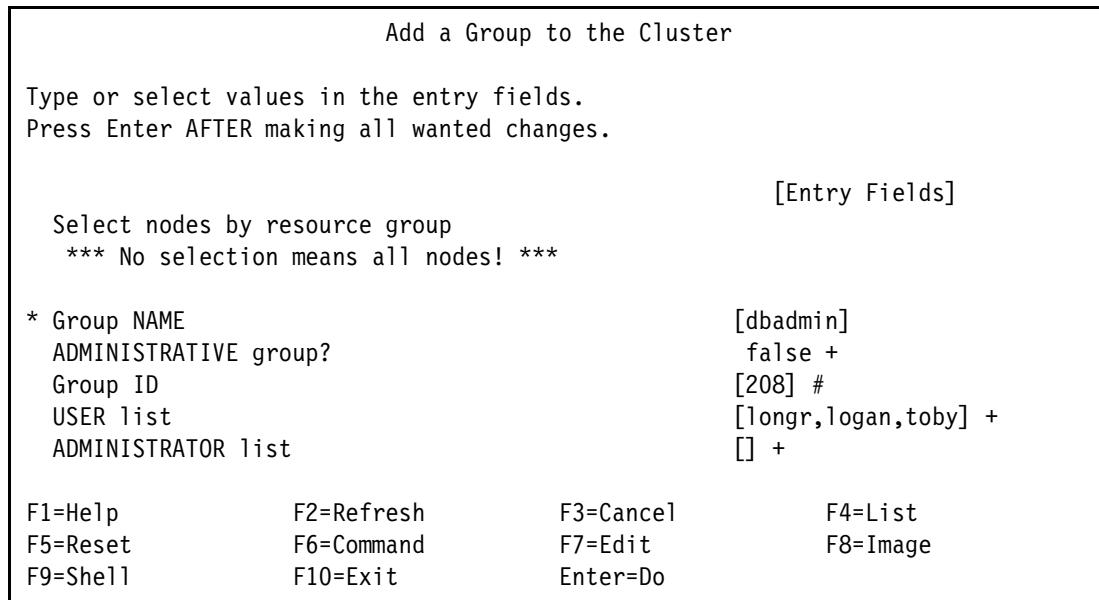


Figure 7-9 Add a Group to the Cluster

Listing groups on the cluster

To list the groups on the cluster, complete the following steps:

1. Start C-SPOC Security and Users by entering **smitty c1_usergroup** and entering **Groups in an PowerHA SystemMirror Cluster → List All Groups in the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes with the groups lists that you want to display. If you leave the Select Nodes by Resource Group field empty, C-SPOC lists all groups on *all* cluster nodes. If you select an RG here, C-SPOC lists groups from only the nodes that belong to the specified RG.

C-SPOC lists the groups and their attributes from the selected nodes, as shown in Example 7-4.

Example 7-4 Listing groups on the cluster

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

```
jordan    system 0      true    root    files
jordan    staff   1      false   sbodily,logan  files
jordan    bin     2      true    root,bin     files
jordan    sys     3      true    root,bin,sys  files
jordan    adm     4      true    bin,adm     files
jordan    security 7      true    root    files
jordan    cron    8      true    root    files
jordan    shutdown 21     true    files
jordan    sshd    205    false   sshd    files
jordan    hacmp   206    false   files
```

jordan	dbgroup208	false	dbadm	root	files
jordan	dbadmin209	false	longr,logan,toby	root	files
jessica	system 0	true	root	files	
jessica	staff 1	false	sbodily,logan		
jessica	bin 2	true	root,bin		files
jessica	sys 3	true	root,bin,sys		files
jessica	adm 4	true	bin,adm		files
jessica	security	7	true	root	files
jessica	cron 8	true	root		files
jessica	shutdown	21	true		files
jessica	sshd 202	false	sshd		files
jessica	hacmp 203	false			files
jessica	dbgroup208	false	dbadm	root	file
jessica	dbadmin209	false	longr,logan,toby	root	files

F1=Help F2=Refresh F3=Cancel F6=Command
 F8=Image F9=Shell F10=Exit /=Find
 n=Find Next

Changing a group in the cluster

To change a group in the cluster, complete the following steps:

1. Start C-SPOC Security and Users by entering `smitty c1_usergroup` and selecting **Groups in an PowerHA SystemMirror Cluster → Change / Show Characteristics of a Group in the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes on which you want to change the groups. If you leave the Select Nodes by Resource Group field empty, you can modify any groups from *all* cluster nodes. If you select an RG here, C-SPOC changes only the groups that are on the nodes that belong to the specified RG.
4. Enter the name of the group that you want to modify or press F4 to select from the list.
5. Change the group attributes (see Figure 7-10).

Change / Show Group Attributes on the Cluster

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

[Entry Fields]

Select nodes by resource group

Group NAME Group ID ADMINISTRATIVE group? USER list ADMINISTRATOR list	dbgroup [208] # false + [dbadm,dbuser] + [root] +
--	---

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

Figure 7-10 Change / Show Group Attributes on the Cluster

Removing a group

To remove a group from a cluster, complete the following steps:

1. Start C-SPOC Security and User by entering `smitty cl_usergroup` and selecting **Groups in a PowerHA SystemMirror Cluster → Remove a Group from the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes whose groups that you want to change. If you leave the Select Nodes by Resource Group field empty, C-SPOC removes the selected group from *all* cluster nodes. If you select an RG here, C-SPOC removes the group from only the nodes that belong to the specified RG. Select the group to remove.
4. Enter the name of the group that you want to modify or press F4 to select from the list.

Considerations for using C-SPOC user and group management

Consider the following information about user and group administration with C-SPOC:

- ▶ C-SPOC user and password management requires the cluster secure communication daemon (`clcomd`) to be running on all cluster nodes. You cannot use C-SPOC if any of your nodes are powered off. In such a case, an error message occurs, similar to the following one:

```
1800-106 An error occurred:  
migcheck[471]: cl_connect() error, nodename=jessica, rc=-1  
migcheck[471]: cl_connect() error, nodename=jessica, rc=-1  
jessica: rshexec: cannot connect to node jessica  
ndu2: cl_rsh had exit code = 1, see cspoc.log or clcomd.log for more  
information.
```

However, you can use C-SPOC regardless of the state of the cluster.

- ▶ Be careful when selecting nodes by RGs. Select exactly the nodes where the user or group you want to modify or remove exists. You cannot modify or remove a user or group if that user or group does not exist on any of the selected nodes.
- ▶ If you encounter an error when using C-SPOC, check `cspoc.log` or `clcomd.log`.

7.3.2 Password management

The PowerHA C-SPOC password management utility is a convenient way for users to change their password on all cluster nodes from a single point of control. If you use this utility, when a user changes their password with the `passwd` command from any cluster node, C-SPOC propagates the new password to all other cluster nodes.

Setting up C-SPOC password management

The C-SPOC password management utilities are disabled by default. To enable them, complete the following steps:

1. Modify the system password utility to use the cluster password utility. On a stand-alone AIX system, the `/usr/bin/passwd` command is used to change a user's password. This command is replaced by the `/usr/es/sbin/cluster/utilities/c1passwd` command when using C-SPOC, which changes the password on all cluster nodes:
 - a. Start C-SPOC Security and Users by entering `smitty c1_usergroup` and selecting **Passwords in an PowerHA SystemMirror cluster → Modify System Password Utility**.
 - b. Press F4 and select Link to Cluster Password Utility from the list (see Figure 7-11).
 - c. Select the nodes where you want to change the password utility. Leave this field blank for all nodes. As a best practice, set up the cluster password utility on all nodes.

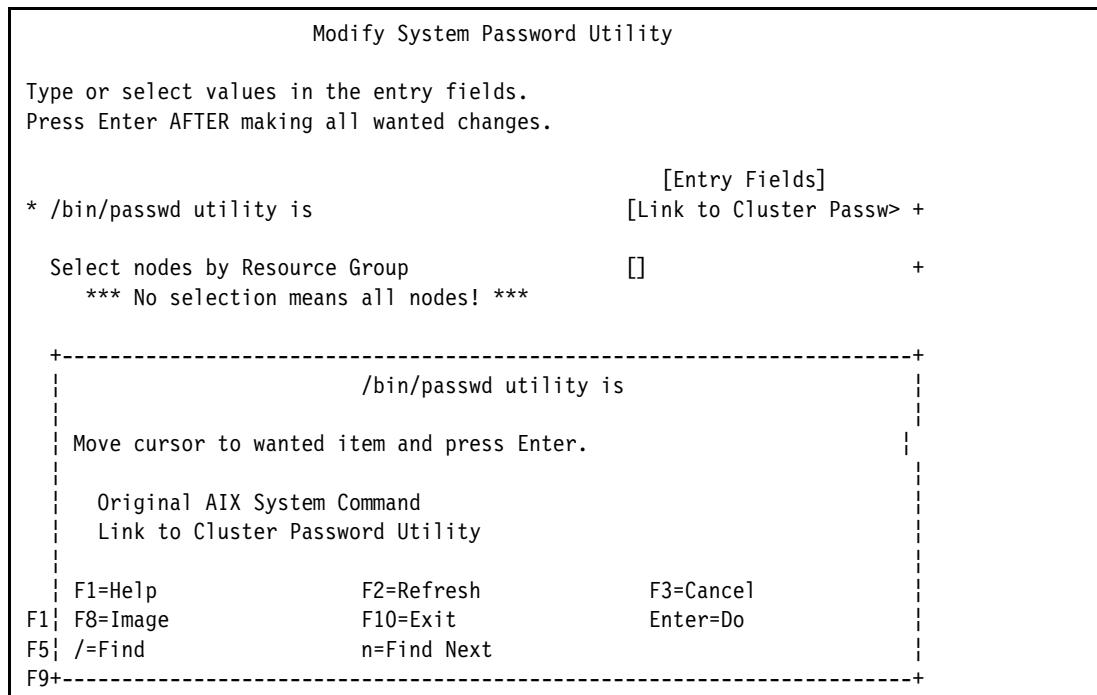


Figure 7-11 Modifying the system password utility

2. Create a list of users who can change their own password from any cluster node:
 - a. Start C-SPOC Security and Users by entering `smitty c1_usergroup` and selecting **Passwords in an PowerHA SystemMirror cluster → Manage List of Users Allowed to Change Password**.
 - b. SMIT shows the users who may change their password cluster-wide (see Figure 7-12 on page 271).

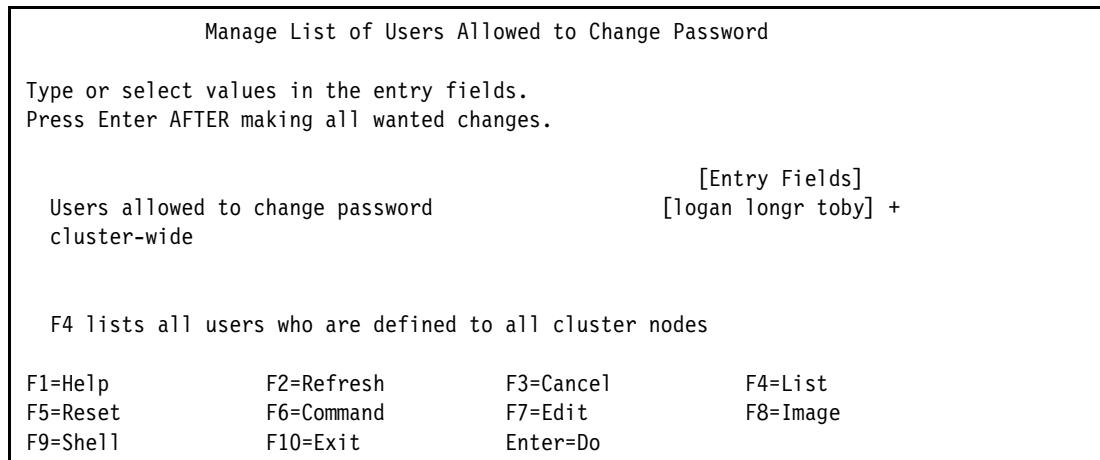


Figure 7-12 Managing the list of users who may change their password cluster-wide

- c. To modify the list of the users who may change their password cluster-wide, press F4 and select the usernames from the list. Choose ALL_USERS to enable all current and future cluster users to use C-SPOC password management (see Figure 7-13).
- As a best practice, include only real named users here, and manually change the password for the technical users.

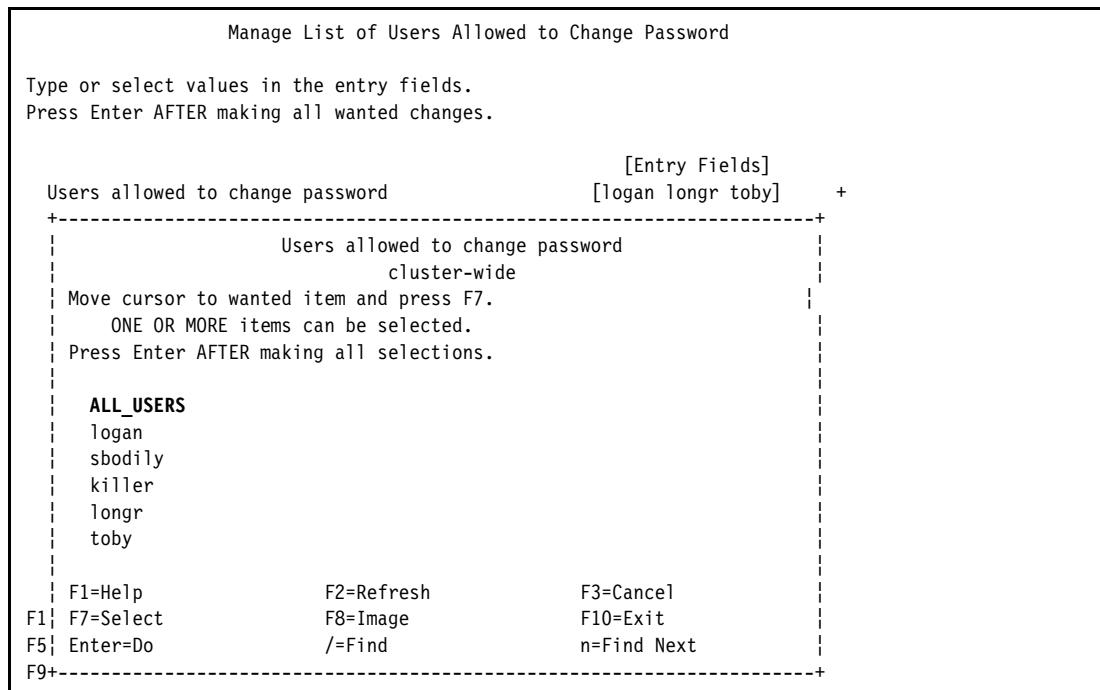


Figure 7-13 Selecting users who may change their password cluster-wide

Note: If you enable C-SPOC password utilities for all users in the cluster, but you have users who exist only on one node, an error message similar to this example occurs:

```
# passwd shane
Changing password for "shane"
shane's New password:
Enter the new password again:
jessica: clpasswdremote: User shane does not exist on node jessica
jessica: cl_rsh had exit code = 1, see cspoc.log or clcomd.log for more
information
```

The password is changed regardless of the error message.

Changing a user password with C-SPOC

To change a user password with C-SPOC, complete the following steps:

1. Start C-SPOC Security and Users by entering **smitty cl_usergroup** and selecting **Passwords in an PowerHA SystemMirror cluster → Change a User's Password in the Cluster**.
2. Select either LOCAL(FILES) or LDAP from the list.
3. Select the nodes on which you want to change the user's password. Leave this field empty for all nodes. If you select an RG here, C-SPOC changes the password only on the nodes that belong to the RG.
4. Type the username or press F4 to select a user from the list.
5. Set User must change password on first login to either true or false (Figure 7-14). The setting depends on what you prefer.

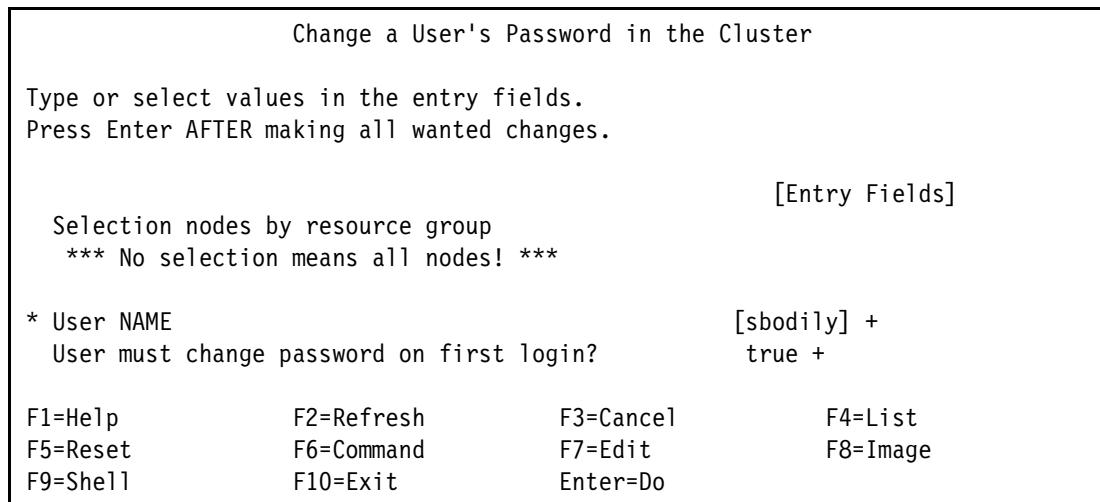


Figure 7-14 Change a User's Password in the Cluster

6. Press Enter and type the new password when prompted.

Tip: You can still use the AIX **passwd** command to change a specific user's password on all nodes if a previously linked password was enabled. Otherwise, you must manually change the password on each node.

Changing your own password

To change your own password, complete the following steps:

1. Start C-SPOC Security and Users by entering `smitty cl_usergroup` and selecting **Passwords in an PowerHA SystemMirror cluster → Change Current Users Password**.
2. Select on which nodes that you want to change your password. Leave this field empty for all nodes. If you select an RG here, C-SPOC changes only the password on nodes that belong to that RG.
3. Your username is shown on the SMIT panel (see Figure 7-15).

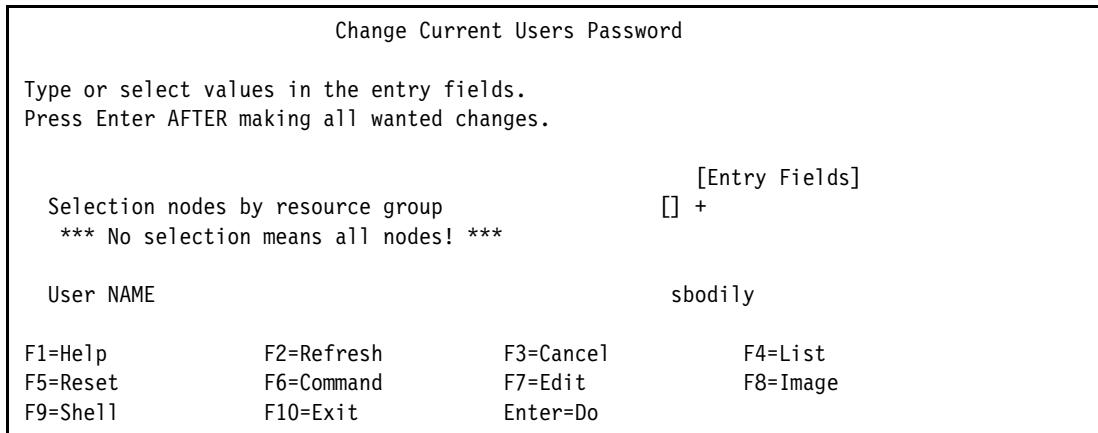


Figure 7-15 Changing your own password

4. Press Enter and change your password when prompted.

Now, your password is changed on all the selected nodes.

Tip: You can use the `passwd` command to change your password on all nodes if the previously linked password was enabled. Otherwise, you must manually change the password on each node.

7.3.3 Encrypted File System management

This section describes how to manage Encrypted File System (EFS) keystores across the cluster.

Enabling the EFS keystore

Complete the following steps:

1. Start C-SPOC Security and Users by entering `smitty cl_usergroup` and selecting **EFS Management** → **Enable EFS Keystore**. Each field is described below:

EFS keystore mode	To see the options, press F4. The options are LDAP and Shared File System.
EFS admin password	A keystore admin password is used to manage EFS. Specify a password to manage keys and permissions.
Volume group for Keystore	This field applies only when the keystore mode is set to Shared File System. Enter the VG where the keystore file system should be. The VG should be type Enhanced Concurrent.
Service IP	This field is for the Service IP to use. This field applies only when the keystore mode is set to Shared File System mode by using a Network File System (NFS).

2. Complete each field as wanted, as shown in Figure 7-16.

Enable EFS Keystore			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
* EFS keystore mode	[Entry Fields]	LDAP	+
EFS admin password		[Redbooks]	+
Volume group for Keystore		[keyvg]	+
Service IP		[ashleysvcip]	+
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

Figure 7-16 Enable EFS Keystore

3. Press Enter to enable the keystore.

Changing or showing the EFS keystore

Complete the following steps:

1. Start C-SPOC Security and Users by entering `smitty cl_usergroup` and selecting **EFS Management** → **Change / Show EFS keystore characteristic**.
2. Change the fields as you want. The password field cannot be changed.
3. Press Enter to make the changes.

Deleting an EFS keystore

Complete the following steps:

1. Start C-SPOC Security and Users by entering `smitty c1_usergroup` and selecting **EFS Management → Delete EFS keystore**.

You see a confirmation dialog box, as shown in Figure 7-16 on page 274.

2. Press Enter to delete the EFS keystore.

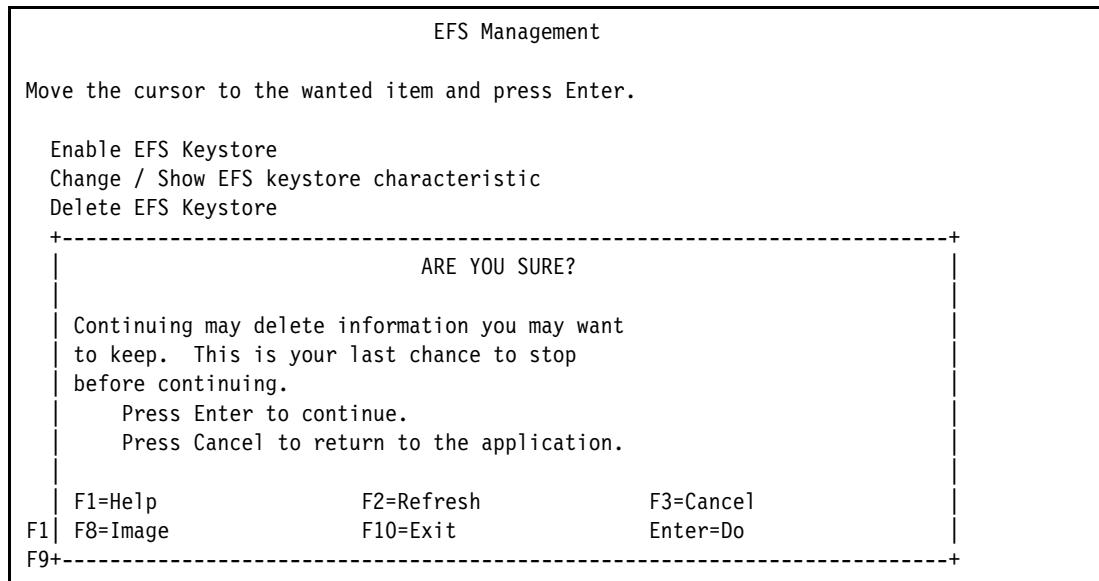


Figure 7-17 Deleting an EFS keystore

7.4 Shared storage management

The PowerHA C-SPOC utility simplifies maintenance of shared Logical Volume Manager (LVM) components in a cluster. C-SPOC commands provide comparable functions in a cluster environment to the standard AIX LVM commands, which can be run on a stand-alone node. By automating repetitive tasks on nodes within the cluster, C-SPOC eliminates a potential source of errors and makes cluster maintenance more efficient. Although you can use the AIX CLI to administer the cluster nodes, it is a best practice to use C-SPOC wherever possible.

7.4.1 Updating LVM components

When changing LVM components manually on nodes within a PowerHA cluster (including VGs, LVs, and file systems), the commands update the AIX ODM on the local node and the Volume Group Descriptor Area (VGDA) on the shared disks in the VG. However, updates to the ODM on all remote nodes require manual propagation to ensure that the cluster operates successfully.

If you use C-SPOC to make LVM changes within a PowerHA cluster, the changes are propagated automatically to all nodes that are selected for the LVM operation.

Importing volume groups manually

The regular AIX based procedure to propagate VG ODM information to other nodes for enhanced concurrent capable VGs is shown in Example 7-5. You can also use the equivalent AIX SMIT command or C-SPOC command-line interface (CLI) command.

Example 7-5 Importing AIX volume groups manually

Tasks performed on the local node (where the volume group is active):

```
node_UK> lsvg -l leevg
leevg:
LV NAME          TYPE    LPs   PPs   PVs  LV STATE    MOUNT POINT
leevglog         jfs2log 1      2      2    open/syncd  N/A
app1lv           jfs2    200   400    4    open/syncd  /appl
node_UK> umount /appl
node_UK> varyoffvg leevg
node_UK> ls -l /dev/leevg
crw-r----- 1 root    system        90,  0 Mar 24 14:50 /dev/leevg
```

Tasks performed on all the other nodes:

```
node_USA> lspv |grep leevg
hdisk1 000685bf8595e225      leevg
hdisk2 000685bf8595e335      leevg
hdisk3 000685bf8595e445      leevg
hdisk4 000685bf8595e559      leevg
node_USA> exportvg leevg
node_USA> importvg -y leevg -n -V 90 hdisk1
node_USA> chvg -a n leevg
node_USA> varyoffvg leevg
```

Note: Ownership and permissions on LV devices are reset when a VG is exported and then reimported. After exporting and importing, a VG is owned by root:system. Some applications that use raw LVs might be affected by this change. Check the ownership and permissions before exporting the VGs and restore them back manually if they are not root:system by default.

Instead of export and import commands, you can use the **importvg -L vgnname hdisk** command on the remote nodes, but be aware that the **-L** option requires that the VG has not been exported on the remote nodes. The **importvg -L** command preserves the LV devices ownership.

Lazy update

In a cluster, PowerHA controls when VGs are activated. PowerHA implements a function that is called *lazy update*. With LVM enhancements adding the learning flag (-L), lazy update became known as *better than lazy update*.

This function examines the VG timestamp, which is maintained in both the VG's VGDA, and the local ODM. AIX updates both these timestamps whenever a change is made to the VG. When PowerHA is going to varyon a VG, it compares the copy of the timestamp in the local ODM with the one in the VGDA. If the values differ, PowerHA refreshes the local ODM information on the VG by using the information in the VGDA.

If a VG under PowerHA control is updated directly (that is, without going through C-SPOC), the information on other nodes on that VG is updated when PowerHA brings the VG online on those nodes, but not before. The actual operations that are performed by PowerHA depend on the state of the VG at the time of activation.

Note: Use C-SPOC to make LVM changes rather than relying on lazy update. C-SPOC imports these changes to all nodes at the time that the C-SPOC operation runs unless a node is powered off. Also, consider using the C-SPOC CLI, as described in 7.4.6, “C-SPOC command-line interface” on page 299.

Importing volume groups automatically

With PowerHA, you may import VGs onto all cluster nodes automatically. This task is done through the Extended Resource Configuration SMIT menu. With automatic import, you may create a VG and then add it to the RG immediately without manually importing it onto each of the destination nodes in the RG.

To use this feature, run **smitty sysmirror** and select **Cluster Applications and Resources → Resource Groups → Change>Show Resources and Attributes for a Resource Group**. Then, select the RG and set its Automatically Import Volume Groups option to true, as shown in Figure 7-18.

Change/Show All Resources and Attributes for a Custom Resource Group		
Type or select values in the entry fields.		
Press Enter AFTER making all wanted changes.		
[MORE...4]	[Entry Fields]	
Startup Policy	Online On Home Node 0>	
Fallover Policy	Fallover To Next Prio>	
Fallback Policy	Never Fallback	
Service IP Labels/Addresses	[ashleysvcip]	+
Application Controllers	[]	+
Volume Groups	[oakleyvg]	+
Use forced varyon of volume groups, if necessary	false	+
Automatically Import Volume Groups	true	+
File systems (empty is ALL for VGs specified)	[]	

Figure 7-18 Automatically importing a volume group

This operation runs after you press Enter. It also automatically switches the setting back to false, which prevents unwanted future imports until you specifically set the option again.

The following guidelines must be met for PowerHA to import available VGs:

- ▶ LVs and file systems must have unique names cluster-wide.
- ▶ All physical disks must be known to AIX and have the appropriate physical volume identifiers (PVIDs) that are assigned.
- ▶ The physical disks on which the VG resides are available to all nodes in the RG.

Importing volume groups by using C-SPOC

With C-SPOC, you can import VGs on all cluster nodes from a single point of control. To do so, complete the following steps:

1. Run `smitty sysmirror` and select **System Management (C-SPOC) → Storage → Volume Groups → Import a Volume Group**.
2. Select a VG to import and the physical disk to use for the import operation. The SMIT panel opens, as shown in Figure 7-19.
3. Complete the fields as wanted and press Enter.

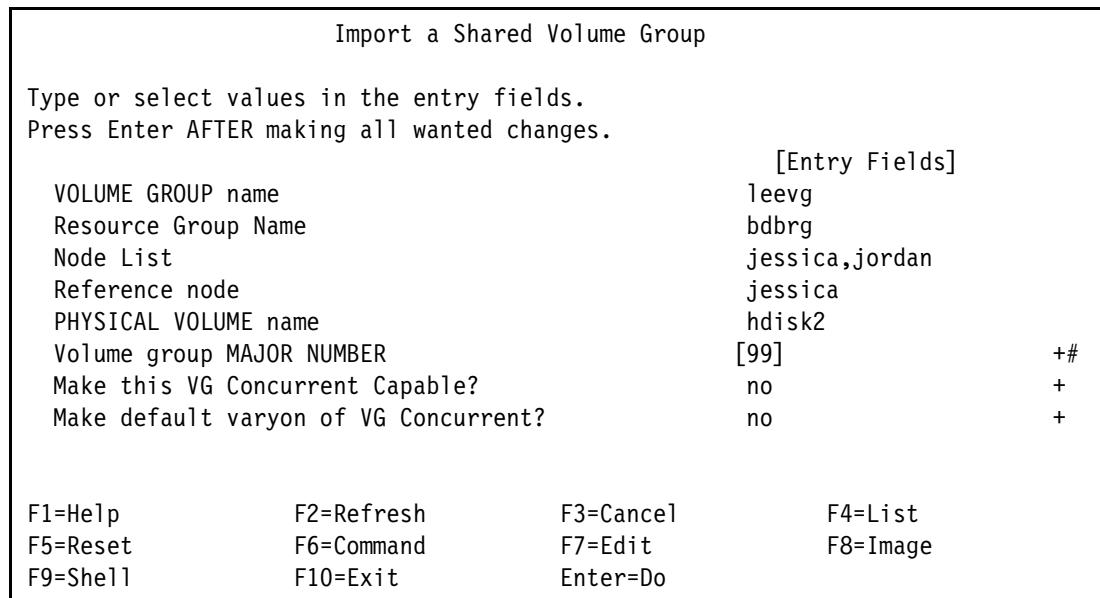


Figure 7-19 C-SPOC importvg panel

7.4.2 Enhanced concurrent volume group LVM limitations

A limitation of using enhanced concurrent volume groups (ECVGs) is the inability to use or change the following LVM VG attributes while a VG is online in concurrent mode:

- ▶ Big VG
- ▶ Hot spare
- ▶ Sync (auto)
- ▶ Bad block relocation

7.4.3 Dynamic volume expansion

C-SPOC does not provide a way to handle dynamic volume expansion (DVE). However, with recent enhancements in AIX, the `chvg -g` command may run on a VG online in concurrent mode.

To increase the size of a shared logical unit number (LUN) that is allocated to your cluster, complete the following steps:

1. Verify that the VG is active in concurrent mode on each node in the cluster.
2. Increase the size of the LUNs.
3. On each node, run the `cfgmgr` command. If you use vSCSI, run `cfgmgr` on each Virtual I/O Server (VIOS) first.

Note: This step might not be required because both VIOS and AIX can automatically detect the change. However, doing this step is a best practice.

4. Verify that the disk size is what you want by running the **bootinfo -s hdisk#** command.
5. Run the **chvg -g vgname** command on *only* the node that has the VG in full active, read/write mode.

DVE example

In this scenario, we have two disks, hdisk6 and hdisk7, that start at 30 GB each, as shown in Example 7-6. They are both members of the 1eevg VG.

Demonstration: For a demonstration of DVE in an active PowerHA 7.1.3 cluster, see this [YouTube video](#).

Example 7-6 Starting disk sizes

```
[jordan:root] /SP1 # clcmd lspv hdisk6 |grep TOTAL
TOTAL PPs:          957 (30624 megabytes)    VG DESCRIPTORS:  1
TOTAL PPs:          957 (30624 megabytes)    VG DESCRIPTORS:  1

[jordan:root] /SP1 # clcmd lspv hdisk7 |grep TOTAL
TOTAL PPs:          957 (30624 megabytes)    VG DESCRIPTORS:  1
TOTAL PPs:          957 (30624 megabytes)    VG DESCRIPTORS:  1

[jordan:root] /SP1 # clcmd bootinfo -s hdisk6
-----
NODE jordan
-----
30624

-----
NODE jessica
-----
30624

[jordan:root] /SP1 # clcmd bootinfo -s hdisk7
-----
NODE jordan
-----
30624

-----
NODE jessica
-----
30624
```

In this example, we begin with the cluster active on both nodes, and the VG is online in concurrent, active-passive mode, as shown in Example 7-7. We parsed out the other irrelevant fields to show the differences after the changes are made. The VG is in active full read/write mode on node jessica. Also, the total VG size is approximately 122 GB.

Example 7-7 Volume group state at the start

```
[jordan:root] /SP1 # clcmd lspv |grep leevg
```

```
-----  
NODE jordan  
-----  
hdisk4      00f6f5d0166106fa      leevg      concurrent  
hdisk5      00f6f5d0166114f3      leevg      concurrent  
hdisk6      00f6f5d029906df4      leevg      concurrent  
hdisk7      00f6f5d0596beebf      leevg      concurrent  
-----  
NODE jessica  
-----  
hdisk4      00f6f5d0166106fa      leevg      concurrent  
hdisk5      00f6f5d0166114f3      leevg      concurrent  
hdisk6      00f6f5d029906df4      leevg      concurrent  
hdisk7      00f6f5d0596beebf      leevg      concurrent
```

```
[jordan:root] /SP1 # clcmd lsvg leevg
```

```
-----  
NODE jordan  
-----  
VOLUME GROUP:    leevg          VG IDENTIFIER: 0f6f5d000004c00000001466765fb16  
VG STATE:       active          PP SIZE:      32 megabytes  
VG PERMISSION:  passive-only   TOTAL PPs:    3828 (122496 megabytes)  
MAX LVs:        256             FREE PPs:    3762 (120384 megabytes)  
Concurrent:     Enhanced-Capable Auto-Concurrent: Disabled  
VG Mode:        Concurrent  
-----  
NODE jessica  
-----  
VOLUME GROUP:    leevg          VG IDENTIFIER: 0f6f5d000004c00000001466765fb16  
VG STATE:       active          PP SIZE:      32 megabytes  
VG PERMISSION:  read/write    TOTAL PPs:    3828 (122496 megabytes)  
MAX LVs:        256             FREE PPs:    3762 (120384 megabytes)  
Concurrent:     Enhanced-Capable Auto-Concurrent: Disabled  
VG Mode:        Concurrent
```

We provision more space onto the disks (LUNs) by adding 9 GB to hdisk6 and 7 GB to hdisk7. Next, we run **cfgmgr** on both nodes. Then, we use **bootinfo -s** to verify that the new sizes are being reported properly, as shown in Example 7-8.

Example 7-8 New disk sizes

```
[jordan:root] /SP1 # clcmd bootinfo -s hdisk6
```

```
-----  
NODE jordan  
-----
```

```
39936
```

```

-----
NODE jessica
-----
39936

[jordan:root] /SP1 # clcmd bootinfo -s hdisk7

-----
NODE jordan
-----
37888

-----
NODE jessica
-----
37888

```

Now, we need to update the VG to be aware of the new space. We do so by running **chvg -g leevg** on node jessica, which has the VG active. Then, we verify the results of the new hdisk size and the new total space to the VG, as shown in Example 7-9. Notice that hdisk6 is now reporting 39 GB, hdisk7 is 37 GB, and the total VG size is now 138 GB.

Example 7-9 Updated volume group information

```

[jessica:root] / # chvg -g leevg

[jordan:root] /SP1 # clcmd lspv hdisk6 |grep TOTAL
TOTAL PPs: 1245 (39840 megabytes) VG DESCRIPTORS: 1
TOTAL PPs: 1245 (39840 megabytes) VG DESCRIPTORS: 1

[jordan:root] /SP1 # clcmd lspv hdisk7 |grep TOTAL
TOTAL PPs: 1181 (37792 megabytes) VG DESCRIPTORS: 1
TOTAL PPs: 1181 (37792 megabytes) VG DESCRIPTORS: 1

[jessica:root] / # clcmd lsvg leevg

-----
NODE jordan
-----
VOLUME GROUP: leevg VG IDENTIFIER: 0f6f5d000004c00000001466765fb16
VG STATE: active PP SIZE: 32 megabytes
VG PERMISSION: passive-only TOTAL PPs: 4340 (138880 megabytes)
MAX LVs: 256 FREE PPs: 4274 (136768 megabytes)
LVs: 2 USED PPs: 66 (2112 megabytes)
Concurrent: Enhanced-Capable Auto-Concurrent: Disabled
VG Mode: Concurrent

-----
NODE jessica
-----
VOLUME GROUP: leevg VG IDENTIFIER: 0f6f5d000004c00000001466765fb16
VG STATE: active PP SIZE: 32 megabytes
VG PERMISSION: read/write TOTAL PPs: 4340 (138880 megabytes)
MAX LVs: 256 FREE PPs: 4274 (136768 megabytes)
LVs: 2 USED PPs: 66 (2112 megabytes)
Concurrent: Enhanced-Capable Auto-Concurrent: Disabled
VG Mode: Concurrent

```

7.4.4 C-SPOC Storage menu

The C-SPOC Storage menu offers the ability to perform LVM commands similar to the ones in the AIX LVM SMIT menus (by running `smitty lvm`). When you use these C-SPOC functions, lists are generated from resources that are available for cluster administration and filtered by resource name. After you select a resource (for example, a VG or a PV) to use, the panels that follow closely resemble the AIX LVM SMIT menus. For AIX administrators who are familiar with AIX LVM SMIT menus, C-SPOC is an simple tool to use.

To select the LVM C-SPOC menu for LV management, run `smitty cspoc` and then select Storage. The following menu options are available:

- ▶ Volume Groups:
 - List All Volume Groups
 - Create a Volume Group
 - Create a Volume Group with Data Path Devices
 - Set Characteristics of a Volume Group
 - Import a Volume Group
 - Mirror a Volume Group
 - Unmirror a Volume Group
 - Manage Critical Volume Groups
 - Synchronize LVM Mirrors
 - Synchronize a Volume Group Definition
 - Remove a Volume Group
 - Manage Mirror Pools for Volume Groups
- ▶ Logical Volumes:
 - List All Logical Volumes by Volume Group
 - Add a Logical Volume
 - Show Characteristics of a Logical Volume
 - Set Characteristics of a Logical Volume
 - Change a Logical Volume
 - Remove a Logical Volume
- ▶ File Systems:
 - List All File Systems by Volume Group
 - Add a File System
 - Change / Show Characteristics of a File System
 - Remove a File System
- ▶ Physical Volumes
 - Remove a Disk From the Cluster
 - Cluster Disk Replacement
 - Cluster Data Path Device Management
 - List all shared Physical Volumes
 - Change>Show Characteristics of a Physical Volume
 - Rename a Physical Volume
 - Show UUID for a Physical Volume
 - Manage Mirror Pools for Volume Groups

For more information about the specific tasks, see 7.4.5, “Examples” on page 283.

7.4.5 Examples

In this section, we present some scenarios for the C-SPOC storage options to administer your cluster. We show the following examples:

- ▶ Adding a scalable ECVG to the existing cluster.
- ▶ Adding a concurrent VG and new concurrent RG to the existing cluster.
- ▶ Creating an LV.
- ▶ Creating a jfs2log LV.
- ▶ Creating a file system.
- ▶ Extending a file system for VGs by using cross-site LVM.
- ▶ Increasing the size of a file system.
- ▶ Mirroring a logical volume.
- ▶ Mirroring a volume group.
- ▶ Synchronizing the LVM mirror.
- ▶ Unmirroring a logical volume.
- ▶ Unmirroring a volume group.
- ▶ Removing a file system.

In our examples, we used two 2-node clusters based on VIOS clients, one Ethernet network that uses IP Address Takeover (IPAT) through aliasing, and a shared repository disk, along with other shared volumes. The storage device is an IBM Storwize V7000 through VIOSs.

Figure 7-20 shows our test cluster setup.

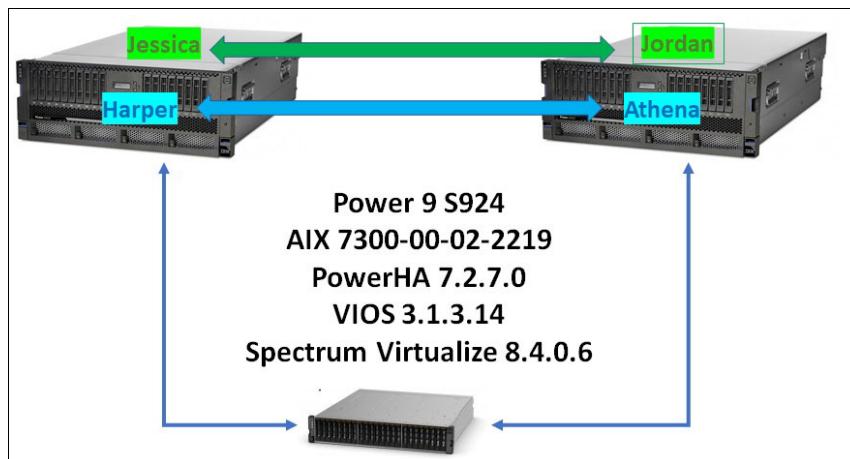


Figure 7-20 C-SPOC LVM testing cluster setup

Adding a scalable enhanced concurrent volume group

The following example shows how to add a new ECVG into the cluster. All shared VGs must be of the enhanced concurrent type. We also add the VG to an existing RG when we create it.

Before creating a shared VG for the cluster by using C-SPOC, check that the following conditions are true:

- ▶ All disk devices are properly configured and in the available state on all cluster nodes.
- ▶ All disks have a PVID.

It was a best practice to manually add PVIDs onto the disks. However, PowerHA can now determine shared disks by universally unique identifier (UUID) and automatically creates the PVIDs that are shown on the list of disks to choose from.

We add the enhanced concurrent capable VG by completing the following steps:

1. Run **smitty cspoc** and then select **Storage → Volume Groups → Create a Volume Group**.
2. Press F7, select the nodes, and press Enter.
3. Press F7, select the disk or disks, and press Enter.
4. Select a VG type from the list.

As a result of the VG type that we chose, we create a scalable VG, as shown in Example 7-10. From here, if we also want to add this VG to an RG, we can either select an existing RG from the list or create an RG. In this example, we add the VG to an existing RG.

Important: When you choose to create an RG from the C-SPOC Logical Volume Management menu, the RG is created with the following default policies. After the group is created, you may change the policies in the Resource Group Configuration.

- ▶ Startup: Online on Home Node Only (OHNO)
- ▶ Failover: Failover To Next Priority Node In The List
- ▶ Fallback: Never Fallback

Example 7-10 Adding a scalable enhanced concurrent VG

Create a Scalable Volume Group

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

[TOP]	[Entry Fields]
Node Names	harper,athena
Resource Group Name	[bdbrg] +
PVID	00c472c006b0b25a
VOLUME GROUP name	[leevg2]
Physical partition SIZE in megabytes	4 +
Volume group MAJOR NUMBER	[67] #
Enable Fast Disk Takeover or Concurrent Access	Fast Disk Takeover +
Volume Group Type	Scalable
CRITICAL volume group?	no +
Max PPs per VG in units of 1024	32 +
Max Logical Volumes	256

After completion, the cluster must be synchronized for it to take effect, as shown by the output in Example 7-11.

Example 7-11 C-SPOC: Creating volume group output

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

```
[TOP]
harper: mkvg: This concurrent capable volume group must be varied on manually.
harper: leevg2
harper: synclvodm: No logical volumes in volume group leevg2.
harper: Volume group leevg2 has been updated.
```

```

athena: synclovdm: No logical volumes in volume group leevg2.
athena: 0516-783 importvg: This imported volume group is concurrently capable.
athena: Therefore, the volume group must be varied on manually.
athena: 0516-1804 chvg: The quorum change takes effect immediately.
athena: Volume group leevg2 has been imported.
c1_mkvg: The PowerHA SystemMirror configuration has been changed - Volume Group
leevg2 has been added. The configuration must be synchronized to make this chan
ge effective across the cluster
[MORE...3]

```

Adding a concurrent VG and a new concurrent RG

The following example shows how to add a new concurrent VG into the cluster. We also show how to add this new VG to a new concurrent RG in the same operation.

Before creating a shared VG for the cluster by using C-SPOC, we check that the following conditions are true:

- ▶ All disk devices are properly configured on all cluster nodes and the devices are listed as available on all nodes.
- ▶ All disks have a PVID.

We add the concurrent VG and RG by completing the following steps:

1. Run **smitty cspoc** and select **Storage → Volume Groups → Create a Volume Group**.
2. Press F7, select the nodes, and then press Enter.
3. Press F7, select the disks, and then press Enter.
4. Select a VG type from the list. As a result of the VG type that we chose, we created a large, concurrent VG, as shown in Example 7-12.

Example 7-12 Creating a concurrent volume group and concurrent resource group

Create a Scalable Volume Group

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

[TOP]	[Entry Fields]	
Node Names	jordan,jessica	
Resource Group Name	[bdbconcrg]	+
PVID	00c472c006b0b25a	
VOLUME GROUP name	[leeconcvg]	
Physical partition SIZE in megabytes	128	+
Volume group MAJOR NUMBER	[67]	#
Enable Fast Disk Takeover or Concurrent Access	Concurrent Access	+
Volume Group Type	Scalable	
CRITICAL volume group?	no	+
Max PPs per VG in units of 1024	32	+
Max Logical Volumes	256	+
Enable Strict Mirror Pools	No	+
Mirror Pool name	[]	
Storage location		+
Enable LVM Encryption	yes	+

Warning:

Changing the volume group major number may result
in the command being unable to run
successfully on a node that does not have the

major number currently available. Please check
for a commonly available major number on all nodes
before changing this setting.

Note: LVM Encryption is enabled by default.

Example 7-13 shows the output from the command that we used to create this VG and RG. The cluster must now be synchronized for the RG changes to take effect. However, the VG information was imported to all cluster nodes that were selected for the operation immediately on creation.

Important: When creating a concurrent RG from the C-SPOC Concurrent Logical Volume Management menu, the RG is created with the following default policies:

- ▶ Startup: Online On All Available Nodes
- ▶ Failover: Bring Offline (On Error Node Only)
- ▶ Fallback: Never Fallback

If the cluster is active at the time of creation, the cluster synchronization performs a DARE and brings the newly created concurrent RG online automatically. If other resources, like an application controller for example, are wanted for the RG, add them manually to the RG before synchronizing the cluster.

Example 7-13 Output from concurrent VG and concurrent RG creation

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

```
[TOP]
jordan: mkvg: This concurrent capable volume group must be varied on manually.
jordan: leeconcrg
jordan: synclvdm: No logical volumes in volume group leeconcrg.
jordan: Volume group leeconcrg has been updated.
jessica: synclvdm: No logical volumes in volume group leeconcrg.
jessica: 0516-783 importvg: This imported volume group is concurrently capable.
jessica: Therefore, the volume group must be varied on manually.
jessica: 0516-1804 chvg: The quorum change takes effect immediately.
jessica: Volume group leeconcrg has been imported.
INFO: The following default policies are used for resource group during volume group
creation.
You can change the policies by using modify resource group policy option.
      Startup Policy as 'Online On All Available Nodes'.
      Failover Policy as 'Bring Offline (On Error Node Only)'.
      Fallback Policy as 'Never Fallback'.
cl_mkvg: The PowerHA SystemMirror configuration has been changed - Resource Group bdbconcrg
has been added. The configuration must be synchronized to make this change effective
across the cluster

cl_mkvg: Discovering Volume Group Configuration...
```

Creating a logical volume

The following example shows how to create an LV in the selected VG, which is already active as part of an RG. This LV can be in any type of RG.

We add jerryclv to the VG that is named leevg by completing the following steps:

1. Run **smitty cspoc** and select **Storage → Logical Volumes → Add a Logical Volume**.
2. Select the VG leeconcv from the list.
3. On the subsequent panel, select devices for allocation, as shown in Example 7-14.

Example 7-14 C-SPOC: Creating a logical volume - 1

Select the Physical Volumes to hold the new Logical Volume		
Move the cursor to the wanted item and press F7. ONE OR MORE items can be selected. Press Enter AFTER making all selections.		
# Reference node jordan	Physical Volume Name hdisk10	
F1=Help F7>Select F1 Enter=Do F9+	F2=Refresh F8=Image / =Find	F3=Cancel F10=Exit n=Find Next

4. Populate the necessary fields, as shown in Example 7-15.

Example 7-15 C-SPOC: Creating a logical volume - 2

Add a Logical Volume		
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.		
[TOP]		
Resource Group Name	[Entry Fields]	bdbconcrgr
VOLUME GROUP name		leeconcv
Node List		jordan,jessica
Reference node		jordan
* Number of LOGICAL PARTITIONS	[9]	#
PHYSICAL VOLUME names	hdisk10	
Logical volume NAME	[jerryclv]	
Logical volume TYPE	[jfs2]	+
POSITION on physical volume	outer_middle	+
RANGE of physical volumes	minimum	+
MAXIMUM NUMBER of PHYSICAL VOLUMES	[]	#
to use for allocation		
Number of COPIES of each logical	1	+
Mirror Write Consistency?	active	+
Allocate each logical partition copy	yes	+
on a SEPARATE physical volume?		
RELOCATE the logical volume during reorganization?	yes	+
Logical volume LABEL	[]	
MAXIMUM NUMBER of LOGICAL PARTITIONS	[512]	#
Enable BAD BLOCK relocation?	yes	+
SCHEDULING POLICY for writing logical	parallel	+
partition copies		
Enable WRITE VERIFY?	no	+

File containing ALLOCATION MAP	[]	/
Stripe Size?	[Not Striped]	
Serlialize I/O?	no	+
Mirror Pool for First Copy		+
Mirror Pool for Second Copy		+
Mirror Pool for Third Copy		+
User ID		+
Group ID		+
Permissions []		X
Enable LVM Encryption	no	+
Auth Method		+
Method Details	[]	
Auth Method Name	[]	+
Serlialize I/O?	no	

The new LV `jerryclv` is created, and information is propagated to the other cluster nodes. The output from this process is shown in Example 7-16.

Example 7-16 C-SPOC: Adding a new logical volume results

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

jordan: jerryclv

WARNING: Encryption for volume group "leeconvg" is enabled, but the logical volume "jerryclv" is not encrypted.
To enable the encryption for logical volume,
You can run "clmgr modify lv jerryclv [...]" or
"use Change a Logical Volume from smitty cl_lv menu".

Note: LVM Encryption is *not* enabled by default for the LV, unlike when creating the VG.

Creating a jfslog2 logical volume

To add a new jfs2log LV `jerrycloglv` into the `leevg` VG, we used the same procedure that is described in “Creating a logical volume” on page 287. In the SMIT panel, which is shown in Example 7-15 on page 287, we select `jfs2log` as the *type* of LV:

Logical volume TYPE [jfs2log]

Important: If an LV of type `jfs2log` is created, C-SPOC automatically runs the `logform` command so that the volume can be used. Also, though file systems are not allowed in “Online on All Available Nodes” RGs C-SPOC does allow its creation because it is only an LV.

Creating a file system

The following example shows how to create a Journaled File System 2 (JFS2) file system on a previously defined LV.

Important: File systems are not allowed on VGs that are a resource in an “Online on All Available Nodes” type RG.

Complete the following steps:

1. Run **smitty cspoc** and then select **Storage → File Systems → Add a File System**, as shown in Figure 7-21.

Add an Enhanced Journaled File System on a Previously Defined Logical Volume			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
[TOP]		[Entry Fields]	
Resource Group		bdbrg	
* Node Names		jordan,jessica	
Logical Volume name		jerryclv	
Volume Group		leevg	
* MOUNT POINT		[/jerrycfs] /	
PERMISSIONS		read/write +	
Mount OPTIONS		[] +	
Block Size (bytes)		4096 +	
Inline Log?		yes +	
Inline Log size (MBytes)		[] #	
Logical Volume for Log		+ +	
Extended Attribute Format		Version 1	
ENABLE Quota Management?		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 7-21 C-SPOC: Creating a JFS2 file system on an existing logical volume

Note: This JFS2 file system was created by using an inline log, which is a best practice.

2. Choose a VG from the list (leevg, in our case).
3. Choose the type of file system (Enhanced, Standard, Compressed, or Large File Enabled).
4. Select the previously created LV, jerryclv, from the list. Complete the necessary fields.

The /jerrycfs file system is now created. The contents of /etc/filesystems are updated on both nodes. If the RG and VG are online, the file system is mounted automatically after creation. It also automatically enables mountguard on the file system, as shown in Example 7-17.

Example 7-17 C-SPOC: Creating a JFS2 file system on an existing logical volume result

COMMAND STATUS		
Command: OK	stdout: yes	stderr: no

Before command completion, additional instructions may appear below.

```
jordan: File system created successfully.  
jordan: 28208 kilobytes total disk space.  
jordan: New File System size is 57344  
jordan: /jerrycfs is now guarded against concurrent mounts.
```

F1=Help	F2=Refresh	F3=Cancel	F6=Command
F8=Image	F9=Shell	F10=Exit	/=Find
n=Find Next			

Mirroring a logical volume

To mirror an LV, complete the following steps:

1. Run **smitty cspoc** and select **Storage → Logical Volumes → Set Characteristics of a Logical Volume → Add a Copy to a Logical Volume.**
2. Choose the VG.
3. Select the LV.
4. Select disks for the mirror copy.
5. Enter the number of copies and other usual LVM parameters, including the mirror pool names, as shown in Example 7-18.

Example 7-18 C-SPOC: Mirroring a logical volume

Add a Copy to a Logical Volume

Type or select values in the entry fields.

Press Enter AFTER making all wanted changes.

[Entry Fields]		
Volume Group Name	leevg	
Resource Group Name	xsitevmRG	
* LOGICAL VOLUME name	xsitelv1	
Reference node		
* NEW TOTAL number of logical partitions	2	+
copies		
PHYSICAL VOLUME names		
POSITION on physical volume	outer_middle	+
RANGE of physical volumes	minimum	+
MAXIMUM NUMBER of PHYSICAL VOLUMES	[]	#
to use for allocation		
Allocate each logical partition copy	yes	+
on a SEPARATE physical volume?		
SYNCHRONIZE the data in the new	no	+
logical partition copies?		
File containing ALLOCATION MAP	[]	/
Mirror Pool for First Copy	[]	+
Mirror Pool for Second Copy	[]	+
Mirror Pool for Third Copy	[]	+

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

Mirroring a volume group

To mirror a VG, complete the following steps:

1. Run **smitty cspoc** and select **Storage → Volume Groups → Mirror a Volume Group.**
2. Choose the VG.
3. Select the PVs. Usually, Auto-select is fine.

4. You can modify the usual **mirrorvg** parameters, such as the number of copies and the mirror pool settings (see Figure 7-22).

Mirror a Volume Group			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
[Entry Fields]			
* VOLUME GROUP name	leevg		
Resource Group Name	xsiteIvmRG		
Node List	jessica,jordan		
Reference node			
PHYSICAL VOLUME names			
Mirror sync mode	Foreground	+	
Number of COPIES of each logical partition	2	+	
Keep Quorum Checking On?	no	+	
Create Exact LV Mapping?	no	+	
Mirror Pool for First Copy	[]	+	
Mirror Pool for Second Copy	[]	+	
Mirror Pool for Third Copy	[]	+	
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell1	F10=Exit	Enter=Do	

Figure 7-22 C-SPOC: Mirror a Volume Group

Synchronizing the LVM mirror

To synchronize an LVM mirror, complete the following steps:

1. Run **smitty cspoc** and select **Storage** → **Volume Groups** → **Synchronize LVM Mirrors** → **Synchronize by Volume Group**.
2. Choose the VG.

3. You can change the following parameters, as shown in Figure 7-23:
- Number of Partitions to Sync in Parallel: Specify a number 1 - 32. Do not select a number higher than the number of disks in the mirror.
 - Synchronize All Partitions: Select Yes for mirroring all partitions regardless of their current synchronization status.
 - Delay Writes to VG from other cluster nodes during this Sync: If the VG belongs to a concurrent RG, you can defer the writes on the other nodes until the sync is finished.

Synchronize LVM Mirrors by Volume Group			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
[Entry Fields]			
VOLUME GROUP name	leevg		
Resource Group Name	xsite1vmRG		
* Node List	jessica,jordan		
Number of Partitions to Sync in Parallel	[1]	+#	
Synchronize All Partitions	no	+	
Delay Writes to VG from other cluster nodes during this Sync	no	+	
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

Figure 7-23 Synchronizing LVM mirrors

Unmirroring a logical volume

To unmirror an LV, complete the following steps:

1. Run **smitty cspoc** and select **Storage → Logical Volumes → Set Characteristics of a Logical Volume → Remove a Copy from a Logical Volume**.
2. Choose the VG.
3. Select the LV.
4. Select the disk that contains the mirror to remove (Example 7-19).
5. Enter the new number of LV copies.

Example 7-19 Remove a Copy from a Logical Volume

Remove a Copy from a Logical Volume			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
[Entry Fields]			
Volume Group Name	leevg		
Resource Group Name	xsite1vmRG		
* LOGICAL VOLUME name	xsite1lv1		
Reference node			
* NEW maximum number of logical partitions	1	+#	
copies			
PHYSICAL VOLUME names to remove copies from			
F1=Help	F2=Refresh	F3=Cancel	F4=List

F5=Reset
F9=Shell

F6=Command
F10=Exit

F7>Edit
Enter=Do

F8=Image

Unmirroring a volume group

To unmirror a VG, complete the following steps:

1. Start **smitty cspo** and select **Storage → Volume Groups → Unmirror a Volume Group** (see Example 7-20).
2. Choose the VG.
3. Select the disk that contains the mirror to remove.
4. Set Number of COPIES of each logical partition (LPAR) to the value that you want, which is usually 1.

Example 7-20 Unmirror a Volume Group

Unmirror a Volume Group

Type or select values in the entry fields.

Press Enter AFTER making all wanted changes.

[Entry Fields]

VOLUME GROUP name	leevg
Resource Group Name	xsite1vmRG
Node List	jessica,jordan
Reference node	jessica
PHYSICAL VOLUME names	hdisk3

Number of COPIES of each logical
partition

1

+

F1=Help
F5=Reset
F9=Shell

F2=Refresh
F6=Command
F10=Exit

F3=Cancel
F7>Edit
Enter=Do

F4=List
F8=Image

Increasing the size of a cross-site LVM mirrored file system

You can use C-SPOC to increase the size of a file system in a cross-site LVM mirrored configuration. The key is to *always* increase the size of the LV first to ensure proper mirroring, and then increase the size of the file system on the previously defined LV.

Important: Always add more space to a file system by adding more space to the LV first. Never add the extra space to the Journalized File System (JFS) first when using cross-site LVM mirroring because the mirroring *might not* be maintained properly.

Similar to creating an LV, be sure to allocate the extra space properly to maintain the mirrored copies at each site. To add more space, complete the following steps:

1. Run **smitty cl_lvsc**, select Increase the Size of a Shared Logical Volume, and press Enter.
2. Choose a VG and RG from the list (Figure 7-24).

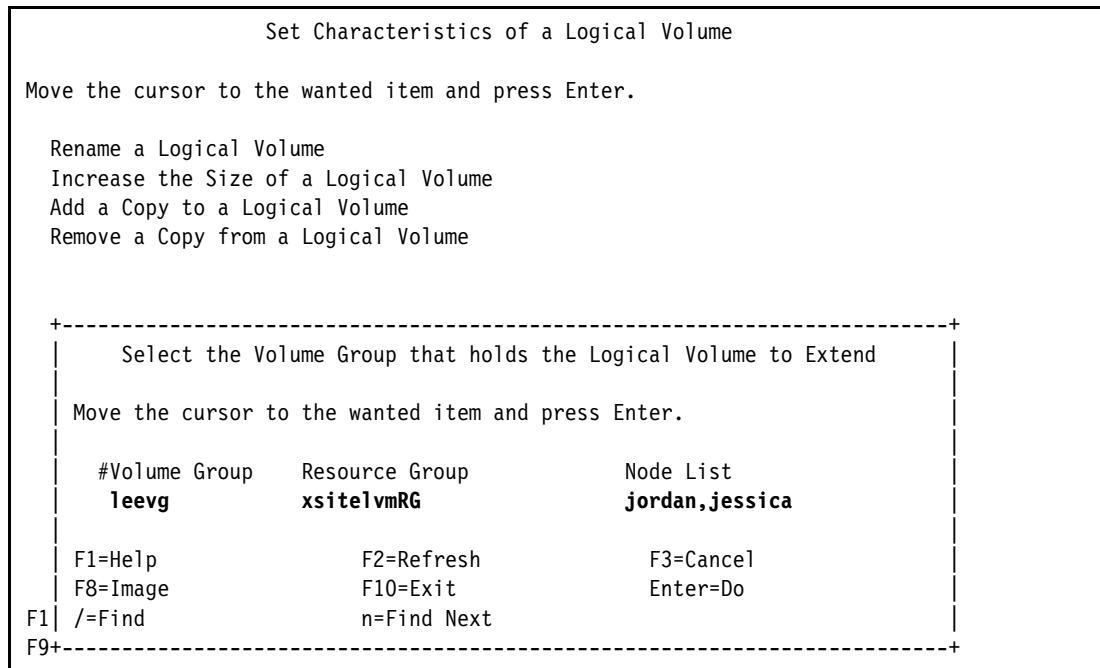
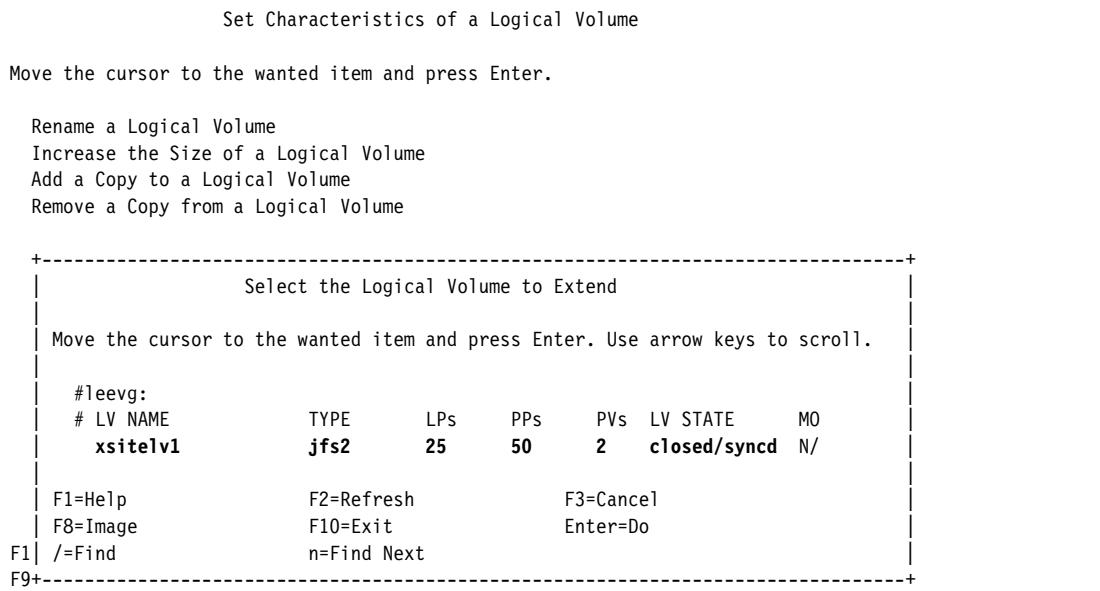


Figure 7-24 Shared volume group list

3. Choose the LV from the next list (Example 7-21). A list of disks is displayed that belong to the same VG as the LV that was previously chosen. The list is similar to the list that is displayed when you create an LV. Press F7, choose the disks, and press Enter.

Example 7-21 Logical volume list selection



Important: Do *not* use the Auto-select option, which is at the top of the list.

4. After selecting the target disks, the final menu opens (see Figure 7-25). Set the following options:
 - RANGE of physical volumes: minimum
 - Allocate each LPAR copy on a SEPARATE physical volume: superstrictThese settings are already set correctly if the LV was originally created correctly.

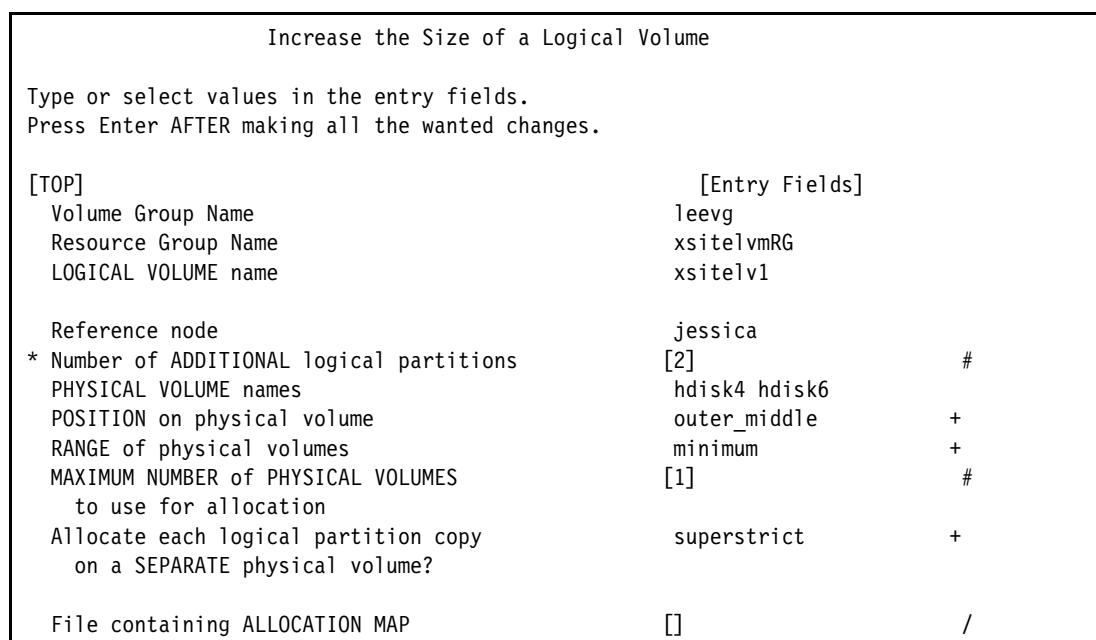


Figure 7-25 Increasing the size of a shared logical volume

5. After adding extra space, verify that the partition mapping is correct by running the `lslv -m lvname` command again, as shown Example 7-22. Highlighted in bold are the two new partitions that were added to the LV.

Example 7-22 Verifying the partition mapping

```
[jessica:root] / # lslv -m xsitev1
xsitev1:N/A
  LPAR   PP1  PV1          PP2  PV2          PP3  PV3
  0001  0193 hdisk4        0193 hdisk6
  0002  0194 hdisk4        0194 hdisk6
  0003  0195 hdisk4        0195 hdisk6
  0004  0196 hdisk4        0196 hdisk6
  0005  0197 hdisk4        0197 hdisk6
  0006  0198 hdisk4        0198 hdisk6
  0007  0199 hdisk4        0199 hdisk6
  0008  0200 hdisk4        0200 hdisk6
  0009  0201 hdisk4        0201 hdisk6
  0010  0202 hdisk4        0202 hdisk6
  0011  0203 hdisk4        0203 hdisk6
  0012  0204 hdisk4        0204 hdisk6
  0013  0205 hdisk4        0205 hdisk6
  0014  0206 hdisk4        0206 hdisk6
  0015  0207 hdisk4        0207 hdisk6
```

0016	0208	hdisk4	0208	hdisk6
0017	0209	hdisk4	0209	hdisk6
0018	0210	hdisk4	0210	hdisk6
0019	0211	hdisk4	0211	hdisk6
0020	0212	hdisk4	0212	hdisk6
0021	0213	hdisk4	0213	hdisk6
0022	0214	hdisk4	0214	hdisk6
0023	0215	hdisk4	0215	hdisk6
0024	0216	hdisk4	0216	hdisk6
0025	0217	hdisk4	0217	hdisk6
0026	0218	hdisk4	0218	hdisk6
0027	0219	hdisk4	0219	hdisk6
0028	0220	hdisk4	0220	hdisk6
0029	0221	hdisk4	0221	hdisk6
0030	0222	hdisk4	0222	hdisk6
0031	0223	hdisk4	0223	hdisk6
0032	0224	hdisk4	0224	hdisk6

6. Now, we add more space to the JFS2 file system. Run **smitty cl_fs** and select Change / Show Characteristics of a File System.
7. Choose the file system, VG, and RG.
8. Complete the remaining fields, as shown in Figure 7-26.

Change/Show Characteristics of an Enhanced Journaled File System

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

[TOP] Volume group name Resource Group Name * Node Names	[Entry Fields] leevg xsitevmRG jordan,jessica		
* File system name NEW mount point SIZE of file system Unit Size Number of Units	/xsitefs [/xsitefs] / Gigabytes + [1] # [] no + read/write +		
Mount GROUP Mount AUTOMATICALLY at system restart? PERMISSIONS Mount OPTIONS Start Disk Accounting? Block Size (bytes) Inline Log? Inline Log size (MBytes) Extended Attribute Format ENABLE Quota Management? Allow Small Inode Extents? Logical Volume for Log Encrypted File System	[] no + no + no + no + 4096 no [0] # [v1] no + yes + xsiteologlv + no		
Esc+1=Help Esc+5=Reset F9=Shell	Esc+2=Refresh F6=Command F10=Exit	Esc+3=Cancel F7>Edit Enter=Do	Esc+4>List F8=Image

Figure 7-26 Increasing the size of the shared Enhanced Journaled File System

9. Ensure that the size of the file system matches the size of the LV. If you are unsure, use the **lsfs -q mountpoint** command, as shown in Example 7-23.

Example 7-23 Verifying the file system and logical volume sizes

```
# lsfs -q /xsitefs
Name           Nodename   Mount Pt    VFS     Size    Options   Auto Accounting
/dev/xsitelv1  --         /xsitefs  jfs2   2097152 rw      no      no
(lv size: 2097152, fs size: 2097152, block size: 4096, sparse files: yes, inline
log: yes, inline log size: 8, EAformat: v1, Quota: no, DMAPI: no, VIX: yes, EFS:
no, ISNAPSHOT: no, MAXEXT: 0, MountGuard: yes, LFF:
no)
```

Removing a file system

To remove a shared file system with C-SPOC, complete the following steps:

1. Manually unmount this file system.
2. Run the **umount** command as follows:
umount /jerrycfs
3. Run **smitty cspoc** and select **Storage → File Systems → Remove a File System**.
4. Select the file system from the list. Confirm the options that are selected on the next SMIT panel, as shown in Figure 7-27, and then press Enter.

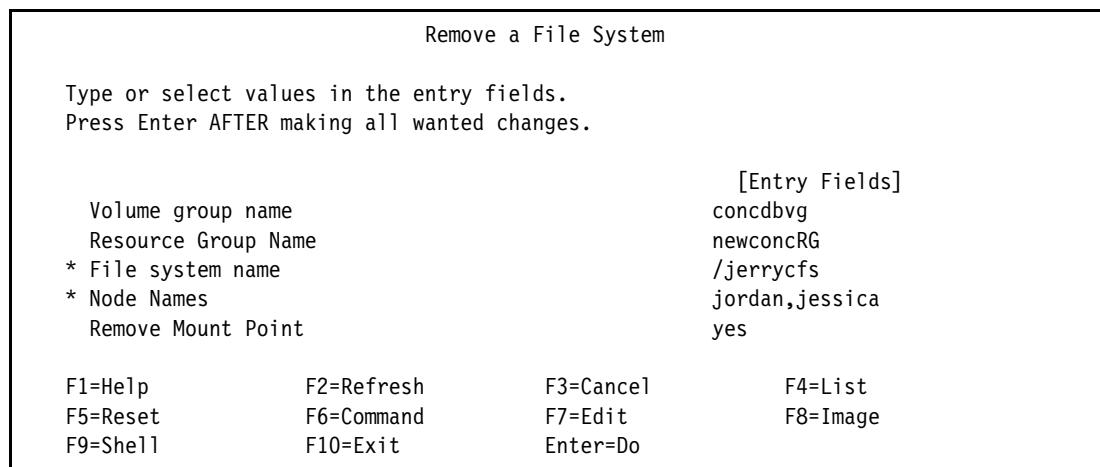


Figure 7-27 C-SPOC: Remove a File System

On completion, the file system is removed from all nodes in the cluster. Because the VG is a resource in the RG, and the VG still exists, there is no need to synchronize the cluster because technically the resources have not changed. However, if you remove the last one from the VG, then you might want to delete the VG, which is described in “Removing a volume group” on page 298.

Removing a logical volume

If the LV has a file system on it, then the previously documented procedure of removing a file system also removes its associated LV. This procedure should be used only for raw LVs and jfslog devices.

To remove a shared LV with C-SPOC, complete the following steps:

1. Run **smitty cspoc** and select **Storage → Logical Volumes → Remove a Logical Volume**.
2. Select the VG from the list.
3. Select the LV from the list.
4. Confirm the options that you selected on the next SMIT panel, as shown in Figure 7-28, and then press Enter.

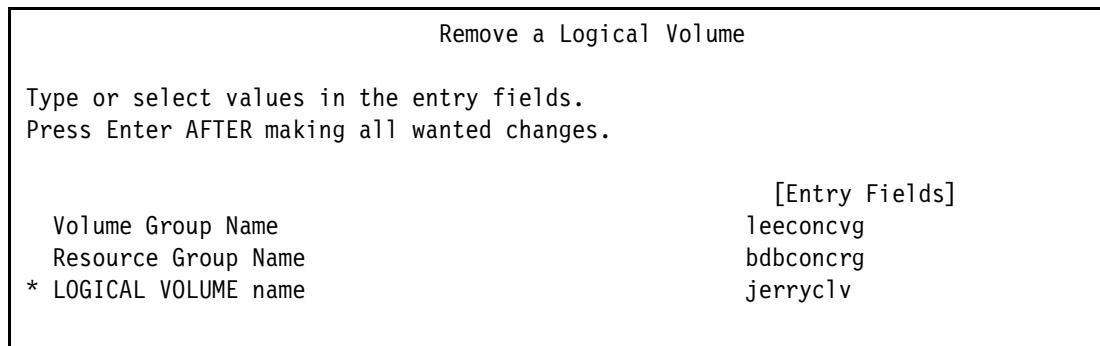


Figure 7-28 C-SPOC: Remove a Logical Volume

On completion, the LV is removed from all nodes in the cluster. Because the VG is a resource in the RG and the VG still exists, there is no need to synchronize the cluster because technically the resources have not changed. However, if you are removing the last RG from the VG, then you might want to delete the VG, which is described in “Removing a volume group” on page 298.

Removing a volume group

To remove a shared VG with C-SPOC, complete the following steps:

1. Ensure that all the file systems are unmounted by running **1svg -l vgnname** and looking for any file systems that show the LV state as open/syncd:

```
1svg -l leevg
leevg:
LV NAME      TYPE    LPs    PPs    PVs   LV STATE    MOUNT POINT
jerrylv       jfs2     10     10     1    open/syncd  /jerrycfs
```

2. If any file systems are mounted, run the **umount** command as follows:

```
umount /jerrycfs
```

3. Ensure that the VG is varied off on all nodes. If not, vary off manually on each node as follows:

```
varyoffvg leevg
```

Note: If the VG is a resource in an RG, you do *not* need to remove it from the RG first because this procedure does so automatically at the end.

4. Run **smitty cspoc** and select **Storage → Volume Groups → Remove a Volume Group**.
5. Choose the VG from list and press Enter.
6. Press Enter again at the ARE YOU SURE confirmation dialog panel. Another confirmation dialog panel, opens, as shown in Figure 7-29 on page 299. A cluster synchronization is required if the VG was removed from an RG.

```

COMMAND STATUS

Command: OK          stdout: yes          stderr: no

Before command completion, additional instructions may appear below.

cl_rmvvg: Volume group leevg has been removed on all nodes

cl_rmvvg: Discovering Volume Group Configuration...
cl_rmvvg: The PowerHA SystemMirror configuration has been changed - volume group
leevg has been removed from resource group bdbrg. The configuration must be
synchronized to make this change effective across the cluster

```

Figure 7-29 C-SPOC: Removing a volume group

7.4.6 C-SPOC command-line interface

C-SPOC has a fully supported CLI. The same tasks that can be run from the C-SPOC SMIT menu can also be run from the CLI.

The CLI is oriented for root users who must run certain tasks with shell scripts rather than through a SMIT menu. The C-SPOC CLI commands are in the `/usr/es/sbin/cluster/cspoc` directory, and they all have a name with the `cli_` prefix.

Similar to the C-SPOC SMIT menus, the CLI commands log their operations in the `cspoc.log` file on the node where the CLI command was run.

A list of the commands is shown in Example 7-24. Although the names are descriptive regarding what function each one offers, full descriptions of each command are available in [PowerHA SystemMirror Commands](#).

Example 7-24 C-SPOC: CLI command listing

```
[jessica:root] /cspoc # ls -al cli_*
-rwxr-xr-x 1 root system 4072 Sep 23 08:34 cli_assign_pvids
-rwxr-xr-x 1 root system 2454 Sep 23 08:34 cli_chfs
-rwxr-xr-x 1 root system 2278 Sep 23 08:34 cli_chgrpmem
-rwxr-xr-x 1 root system 2388 Sep 23 08:34 cli_chlv
-rwxr-xr-x 1 root system 2564 Sep 23 08:34 cli_chvg
-rwxr-xr-x 1 root system 2446 Sep 23 08:34 cli_crfs
-rwxr-xr-x 1 root system 2751 Sep 23 08:34 cli_crivfs
-rwxr-xr-x 1 root system 3329 Sep 23 08:34 cli_extendlv
-rwxr-xr-x 1 root system 6211 Sep 23 08:34 cli_extendvg
-rwxr-xr-x 1 root system 5606 Sep 23 08:34 cli_importvg
-rwxr-xr-x 1 root system 3313 Sep 23 08:34 cli_mirrorvg
-rwxr-xr-x 1 root system 3659 Sep 23 08:34 cli_mklv
-rwxr-xr-x 1 root system 3232 Sep 23 08:34 cli_mklvcopy
-rwxr-xr-x 1 root system 6555 Sep 23 08:34 cli_mkvg
-rwxr-xr-x 1 root system 2710 Sep 23 08:34 cli_on_cluster
-rwxr-xr-x 1 root system 3264 Sep 23 08:34 cli_on_node
-rwxr-xr-x 1 root system 3648 Sep 23 08:34 cli_reducevg
-rwxr-xr-x 1 root system 3879 Sep 23 08:34 cli_replacepv
-rwxr-xr-x 1 root system 2412 Sep 23 08:34 cli_rmfs
-rwxr-xr-x 1 root system 2883 Sep 23 08:34 cli_rmlv
-rwxr-xr-x 1 root system 3224 Sep 23 08:34 cli_rmlvcopy
-rwxr-xr-x 1 root system 2398 Sep 23 08:34 cli_syncvg
```

```
-rwxr-xr-x 1 root system 3336 Sep 23 08:34 cli_unmirrorvg  
-rwxr-xr-x 1 root system 2386 Sep 23 08:34 cli_updatevg
```

7.5 Time synchronization

PowerHA does not perform time synchronization for you, so be sure to implement time synchronization within your clusters. Some applications require time synchronization, but from an administration perspective, running **ntpd** eases administration within a clustered environment.

7.6 Cluster verification and synchronization

Verification and synchronization of the PowerHA cluster ensures that all resources under PowerHA control are configured and that all rules regarding resource ownership and other parameters are consistent across nodes in the cluster.

The PowerHA cluster stores the information about all cluster resources and cluster topology, and also several other parameters in PowerHA that are specific object classes in the ODM. PowerHA ODM files must be consistent across all cluster nodes so that cluster behavior works as designed. Cluster verification checks the consistency of PowerHA ODM files across all nodes and also verifies whether PowerHA ODM information is consistent with required AIX ODM information. If verification is successful, the cluster configuration can be synchronized across all the nodes. Synchronization is effective immediately in an active cluster. Cluster synchronization copies the PowerHA ODM from the local nodes to all remote nodes.

Note: If the cluster is not synchronized and failure of a cluster topology or resource component occurs, the cluster might be unable to fail over as designed. PowerHA provides the capability to run cluster verification on a regular or daily basis. For more information about this feature, see 7.6.6, “Running automatic corrective actions during verification” on page 310.

7.6.1 Cluster verification and synchronization by using SMIT

By using SMIT (run **smitty sysmirror**), at least four verification and synchronization paths are available:

- ▶ Cluster Nodes and Networks
- ▶ Cluster Applications and Resources
- ▶ Custom Cluster Configuration
- ▶ Problem Determination Tools (verification only path)

Cluster Nodes and Networks

To use this verify and synchronize path, run **smitty sysmirror** and select **Cluster Nodes and Networks** → **Verify and Synchronize Cluster Configuration**.

When you use the path Cluster Nodes and Networks, synchronization occurs automatically after successful verification of the cluster configuration. There are no additional options in this menu. This option does *not* use the feature that automatically corrects errors that are found during verification. For more information about automatically correcting errors that are found during verification, see 7.6.6, “Running automatic corrective actions during verification” on page 310.

Cluster Applications and Resources

The same type of verification and synchronization that is in Cluster Nodes and Networks is also available by running `smitty sysmirror` and selecting **Cluster Applications and Resources** → **Verify and Synchronize Cluster Configuration**.

Custom Cluster Configuration

To use the Custom Cluster Configuration path to verify your cluster, run `smitty sysmirror` and select **Custom Cluster Configuration** → **Verify and Synchronize Cluster Configuration (Advanced)**.

The Custom Cluster Configuration Verification and Synchronization path parameters depend on the cluster services state. If any node is active in the cluster, it is considered as an active cluster. Only if *all* nodes are inactive is it considered as an inactive cluster.

Figure 7-30 shows the SMIT panel that is displayed when cluster services *are* active. Performing a synchronization in an active cluster is also called DARE.

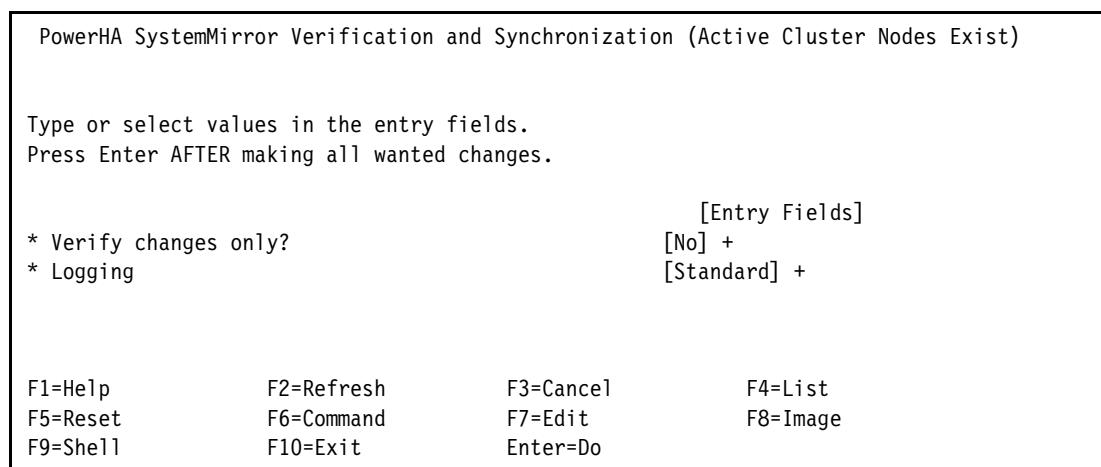


Figure 7-30 Verification and Synchronization panel: active cluster

In an active cluster, the SMIT panel parameters are as follows (also shown in Figure 7-30):

- ▶ Verify changes only:
 - Select No to run a full check of topology and resources.
 - Select Yes to verify only the changes that were made to the cluster configuration (PowerHA ODM) since the last verification.
- ▶ Logging: Select Verbose to send the full output to the console, which otherwise is directed to the `clverify.log` file.

Figure 7-31 shows the SMIT panel that is displayed when cluster services are *not* active. Change the field parameters (the default option is both verification and synchronization, as shown in the example) and press Enter.

PowerHA SystemMirror Verification and Synchronization			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
[Entry Fields]			
* Verify, Synchronize or Both	[Both] +		
* Include custom verification library checks	[Yes] +		
* Automatically correct errors found during verification?	[No] +		
* Force synchronization if verification fails?	[No] +		
* Verify changes only?	[No] +		
* Logging	[Standard] +		
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

Figure 7-31 Verification and Synchronization panel: inactive cluster

- ▶ Verify, Synchronize, or Both:
 - Select Verify to run verification only.
 - Select Synchronize to run synchronization only.
 - Select Both to run verification, and when it completes, synchronization is done (with the Both option, you can use the Force synchronization if verification fails option).
- ▶ Include custom verification library checks: This option can be set to either Yes or No.
- ▶ Automatically correct errors found during verification: This option can be set to No, Interactively, or Yes. For more information, see 7.6.6, “Running automatic corrective actions during verification” on page 310.
- ▶ Force synchronization if verification fails:
 - Select No to stop synchronization from commencing if the verification procedure returns errors.
 - Select Yes to force synchronization regardless of the result of verification. In general, do not force synchronization. In some specific situations, if synchronization must be forced, ensure that you fully understand the consequences of these cluster configuration changes.
- ▶ Verify changes only:
 - Select No to run a full check of topology and resources.
 - Select Yes to verify only the changes that occurred in the PowerHA ODM files since the time of the last verification operation.
- ▶ Logging: Select Verbose to send the full output to the console, which is otherwise directed to the `clverify.log` file.

Note: Synchronization can be initiated on either an active or inactive cluster. If some nodes in the cluster are inactive, synchronization can be initiated only from an active node by using DARE. For more information about DARE, see 7.6.3, “Dynamic cluster reconfiguration with DARE” on page 306.

Problem Determination Tools verification path

To use the Problem Determination Tools path to run verification, run `smitty sysmirror` and select **Problem Determination Tools → PowerHA SystemMirror Verification → Verify PowerHA SystemMirror Configuration**.

If you are using the Problem Determination Tools path, you have more options for verification, such as defining custom verification methods. However, synchronizing the cluster from here is not possible. The SMIT panel of the Problem Determination Tools verification path is shown in Figure 7-32.

Note: Verification by using the Problem Determination Tools path can be initiated either from active or inactive nodes.

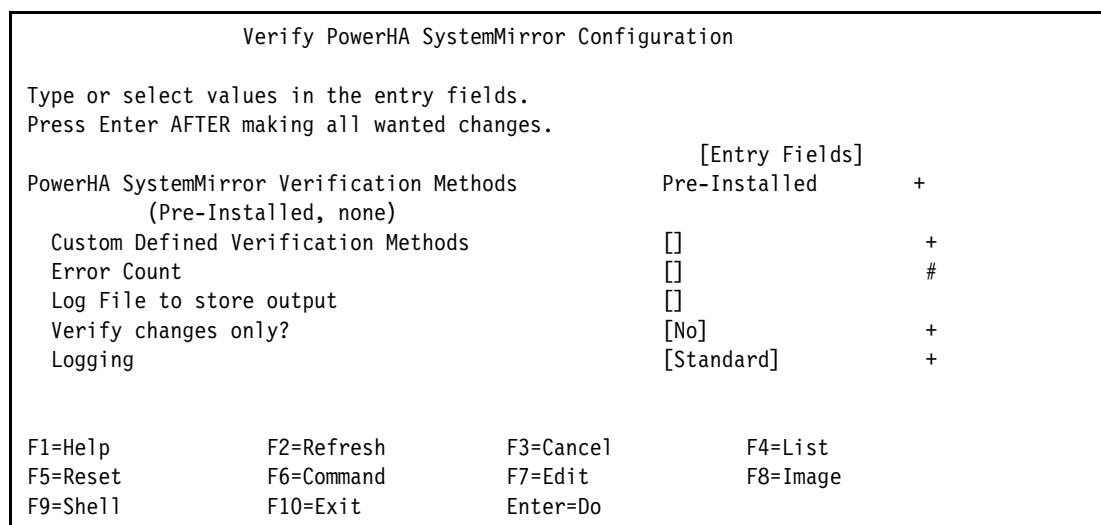


Figure 7-32 Verification panel by using the Problem Determination Tools path

If verification fails, correct the errors and repeat verification to ensure that the problems are resolved as soon as possible. The messages that are output from verification indicate where the error occurred (for example, on a node, a device, or a command). In 7.6.5, “Verification log files” on page 309, we describe the location and purpose of the verification logs.

7.6.2 Cluster verification and synchronization by using a CLI (clmgr)

Like most operations, verification and synchronization can be performed from the CLI by running `clmgr`. Here are examples and descriptions of each option. The following examples are not exhaustive of all the options that are available, but are representative of the most commonly used options.

Verifying a cluster

You can do a simple verification from any node in the cluster by running **clmgr verify cluster**. The output, even with an existing problem, is shown in Example 7-25.

Example 7-25 Clmgr cluster verification

```
# clmgr verify cluster

Verification to be performed on the following:
  Cluster Topology
  Cluster Resources

Retrieving data from available cluster nodes. This might take a few minutes.

  Start data collection on node jordan
  Start data collection on node jessica
  Collector on node jessica completed
  Collector on node jordan completed
  Data collection complete
WARNING: No backup repository disk is UP and not already part of a VG for nodes:
  - jordan
  - jessica

For nodes with a single Network Interface Card per logical
network is configured, it is recommended to include the file
'/usr/es/sbin/cluster/netmon.cf' with a "pingable"
IP address as described in the 'PowerHA SystemMirror Planning Guide'.
WARNING: File 'netmon.cf' is missing or empty on the following nodes:
jordan
jessica
  Completed 10 percent of the verification checks
  Completed 20 percent of the verification checks
  Completed 30 percent of the verification checks
This cluster uses Unicast heartbeat
ERROR: Logical volume testlv2 not found for VG vgQ4Nsap on node jessica.
  Completed 40 percent of the verification checks
  Completed 50 percent of the verification checks
s  Completed 60 percent of the verification checks
    Completed 70 percent of the verification checks

Verifying XD Solutions...

  Completed 80 percent of the verification checks
  Completed 90 percent of the verification checks
  Completed 100 percent of the verification checks
Python version is the same on all the nodes in a cluster.
WARNING: There is no split and merge policy set.
You may also configure a quarantine policy that uses a disk fencing or an active node halt
policy.
The quarantine policy can be used instead of a split and merge policy or both can be used
at the same time for added protection.

Verification exiting with error count: 1
```

Synchronizing a cluster

Anytime that a cluster change is made, it is often necessary to synchronize the cluster. A cluster verification reports whether there are any existing cluster problems. Also, a check can be performed to see whether any unsynced changes exist by running **clmgr -a UNSYNCED_CHANGES query cluster**, as shown in Example 7-26. The possible settings are true or false, which are self-explanatory. This command can be run on any node on the cluster and should yield the same results regardless on which node it runs.

Example 7-26 Checking for unsynced changes

```
# clmgr -a UNSYNCED_CHANGES query cluster  
UNSYNCED_CHANGES="true"
```

Performing a cluster verification often points to which node a problem exists on, which infers that syncing from the opposite node is needed. To perform a cluster synchronization from the CLI, run **clmgr sync cluster**, as shown in Example 7-27. Then, verify again that no unsynced changes exist.

Example 7-27 Clmgr cluster synchronization

```
# clmgr sync cluster  
  
Committing any changes, as required, to all available nodes...  
  
Adding any necessary PowerHA SystemMirror entries to /etc/inittab and  
/etc/rc.net for IPAT on node jordan.  
  
Checking for added nodes  
1 tunable updated on cluster redbook_cluster.  
Adding any necessary PowerHA SystemMirror entries to /etc/inittab and  
/etc/rc.net for IPAT on node jessica.  
  
Verification has completed normally.  
  
clsnapshot: Creating file /var/hacmp/clverify/pass/clver_pass_snapshot.odm...  
  
clsnapshot: Succeeded creating Cluster Snapshot: clver_pass_snapshot.  
  
Verification to be performed on the following:  
Cluster Topology  
Cluster Resources  
  
Retrieving data from available cluster nodes. This might take a few minutes.  
  
Start data collection on node jordan  
Start data collection on node jessica  
Collector on node jessica completed  
Collector on node jordan completed  
Data collection complete  
WARNING: No backup repository disk is UP and not already part of a VG for nodes:  
- jordan  
- jessica
```

For nodes with a single Network Interface Card per logical network configured, it is recommended to include the file '/usr/es/sbin/cluster/netmon.cf' with a "pingable" IP address as described in the 'PowerHA SystemMirror Planning Guide'.
WARNING: File 'netmon.cf' is missing or empty on the following nodes:
jordan
jessica

```
Completed 10 percent of the verification checks
Completed 20 percent of the verification checks
Completed 30 percent of the verification checks
This cluster uses Unicast heartbeat
Completed 40 percent of the verification checks
Completed 50 percent of the verification checks
Completed 60 percent of the verification checks
Completed 70 percent of the verification checks
```

Verifying XD Solutions...

```
Completed 80 percent of the verification checks
Completed 90 percent of the verification checks
Completed 100 percent of the verification checks
Python version is the same on all the nodes in a cluster.
WARNING: There is no split and merge policy set.
You may also configure a quarantine policy that uses a disk fencing or an active node halt
policy.
The quarantine policy can be used instead of a split and merge policy or both can be used
at the same time for added protection.
```

Verification has completed normally.

```
# clmgr -a UNSYNCED_CHANGES query cluster
UNSYNCED_CHANGES="false"
```

Cross-Cluster Verification utility

The Cross-Cluster Verification (CCV) utility, which is provided by the **c1ccv** command that was introduced in PowerHA 7.2.4, compares specific attributes of two different cluster configurations. The CCV utility compares data that is collected from different clusters, cluster snapshots, the active configuration directory, or the default configuration directory of a local cluster. This comparison can be helpful in determining configuration differences between clusters. This determination is especially helpful when there is a requirement for the cluster configurations to be consistent. However, there are specific requirements that are needed before you use the CCV tool. For more information, see [Verifying two different clusters by using the CCV utility](#).

7.6.3 Dynamic cluster reconfiguration with DARE

With PowerHA, you may make most changes to both the cluster topology and the cluster resources while the cluster is running. This feature is referred to as DARE. You can make several supported resource and topology changes in the cluster and then use the dynamic reconfiguration event to apply those changes to the active cluster without having to bring cluster nodes offline, which result in the whole operation being faster, especially for complex configuration changes.

Considerations:

- ▶ When the cluster synchronization (DARE) takes place, action is taken on any resource or topology component to be changed or removed immediately.
- ▶ Running a DARE operation on a cluster that has nodes running at different versions of the PowerHA code, for example, during a cluster migration, is *not* supported.
- ▶ You cannot perform a DARE operation while any node in the cluster is in the UNMANAGED state.

The following changes can be made to a topology in an active cluster by using DARE:

- ▶ Add or remove nodes.
- ▶ Add or remove network interfaces.
- ▶ Add or remove networks.
- ▶ Change the network type from public to private.
- ▶ Swap a network interface card (NIC).
- ▶ Change between unicast and multicast for heartbeating.

Restriction: Although dynamically changing a network type between public and private is supported, you cannot actively host a mix of boot and service IP addresses, as shown below:

ERROR: Network net_ether_01 has a mix of interface types.
Only networks with all boot or all service labels can be converted to private.

The following changes can be made to the resources in an active cluster that uses DARE:

- ▶ Add, remove, or change an application server.
- ▶ Add, remove, or change application monitoring.
- ▶ Add or remove the contents of one or more RGs.
- ▶ Add, remove, or change a tape resource.
- ▶ Add or remove RGs.
- ▶ Add, remove, or change the order of participating nodes in an RG.
- ▶ Change the node relationship of the RG.
- ▶ Change the RG processing order.
- ▶ Add, remove, or change the fallback timer policy that is associated with an RG. The new fallback timer does not have any effect until the RG is brought online on another node.
- ▶ Add, remove, or change the settling time for RGs.
- ▶ Add or remove the node distribution policy for RGs.
- ▶ Add, change, or remove parent-child or location dependencies for RGs (some limitations apply here).
- ▶ Add, change, or remove inter-site management policy for RGs.
- ▶ Add, remove, or change pre-events or post-events.

Dynamic reconfiguration can be initiated only from an active cluster node, which means from a node that has cluster services running. The change must be made from a node that is active so that the cluster can be synchronized.

Before changing a cluster definition, ensure that these items are true:

- ▶ The same version of PowerHA is installed on all nodes.
- ▶ Some nodes are running PowerHA and are able to communicate with each other. No node should be in an UNMANAGED state.
- ▶ The cluster is stable and the hacmp.out log file does not contain recent event errors or config_too_long events.

Depending on the cluster configuration and on the specific changes that you want to make in an active cluster environment, there are many options and limitations when performing a dynamic reconfiguration event. These items must all be understood, including the consequences of changing an active cluster configuration. Before you make dynamic changes in an active PowerHA environment, see [Administering PowerHA SystemMirror](#).

7.6.4 Changing between multicast and unicast

To change from multicast to unicast communication mode or unicast to multicast mode, complete the following steps.

Important: If you switch to multicast, the physical network must already have multicast enabled; otherwise, this change might result in unwanted results, including an outage.

1. Verify that the existing Cluster Aware AIX (CAA) communication mode is set to multicast, as shown in Example 7-28.

Example 7-28 Verifying the CAA communication mode

```
#[jessica:root] / # lscluster -c
Cluster Name: xsite_cluster
Cluster UUID: 97efba5c-f4fa-11e3-98bd-eeaf01717802
Number of nodes in cluster = 2
    Cluster ID for node jessica: 1
    Primary IP address for node jessica: 192.168.100.51
    Cluster ID for node jordan: 2
    Primary IP address for node jordan: 192.168.100.52
Number of disks in cluster = 1
    Disk = hdisk1 UUID = 06db51a6-a39a-f9e3-c916-b9b897f1f447 cluster_major
= 0 cluster_minor = 1
Multicast for site LOCAL: IPv4 228.168.100.51 IPv6 ff05::e4a8:6433
Communication Mode: multicast
Local node maximum capabilities: HNAME_CHG, UNICAST, IPV6, SITE
Effective cluster-wide capabilities: HNAME_CHG, UNICAST, IPV6, SITE
```

2. Change the heartbeat mechanism from multicast to unicast by running **smitty cm_define_repos_ip_addr**. The final SMIT menu is displayed, as shown in Example 7-29.

Example 7-29 Changing the heartbeat mechanism

```
Define Repository Disk and Cluster IP Address

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

* Cluster Name [Entry Fields]
  xsite_cluster
* Heartbeat Mechanism
  Unicast +
  Repository Disk
  00f6f5d015a4310b
  Cluster Multicast Address
  228.168.100.51
  (Used only for multicast heartbeat)
```

Note: Once a cluster has been defined to AIX, all that can be modified is the Heartbeat Mechanism

3. Verify and synchronize the cluster.
4. Check that the new CAA communication mode is now set to unicast, as shown in Example 7-30.

Example 7-30 Verifying the new CAA communication mode

```
[jessica:root] / # lscluster -c
Cluster Name: xsite_cluster
Cluster UUID: 97efba5c-f4fa-11e3-98bd-eeaf01717802
Number of nodes in cluster = 2
    Cluster ID for node jessica: 1
    Primary IP address for node jessica: 192.168.100.51
    Cluster ID for node jordan: 2
    Primary IP address for node jordan: 192.168.100.52
Number of disks in cluster = 1
    Disk = hdisk1 UUID = 06db51a6-a39a-f9e3-c916-b9b897f1f447 cluster_major =
0 cluster_minor = 1
Multicast for site LOCAL: IPv4 228.168.100.51 IPv6 ff05::e4a8:6433
Communication Mode: unicast
Local node maximum capabilities: HNAME_CHG, UNICAST, IPV6, SITE
Effective cluster-wide capabilities: HNAME_CHG, UNICAST, IPV6, SITE
```

7.6.5 Verification log files

During cluster verification, PowerHA collects configuration data from all cluster nodes while it runs through the list of checks. The verbose output is saved to the clverify.log file. The log file is rotated.

Example 7-31 shows the /var/hacmp/clverify directory contents with verification log files.

Example 7-31 Output with verification log files

root@jessica	/var/hacmp/clverify]	ls -al clverify.lo*		
-rw-----	1	root	system	112276 Oct 27 08:48 clverify.log
-rw-----	1	root	system	112090 Oct 27 08:43 clverify.log.1
-rw-----	1	root	system	112546 Oct 27 00:00 clverify.log.2
-rw-----	1	root	system	112379 Oct 26 15:04 clverify.log.3
-rw-----	1	root	system	112600 Oct 26 15:00 clverify.log.4
-rw-----	1	root	system	112490 Oct 26 14:58 clverify.log.5
-rw-----	1	root	system	112420 Oct 26 14:54 clverify.log.6
-rw-----	1	root	system	113637 Oct 26 14:53 clverify.log.7
-rw-----	1	root	system	114854 Oct 26 14:52 clverify.log.8
-rw-----	1	root	system	114854 Oct 26 14:52 clverify.log.9

On the local node, where you initiate the cluster verification command, detailed information is collected in the log files, which contain a record of all data that is collected, the tasks that are performed, and any errors. These log files are written to the following directories and are used by a service technician to determine the location of errors:

- ▶ If verification succeeds: /var/hacmp/clverify/pass/nodename/
- ▶ If verification fails: /var/hacmp/clverify/fail/nodename/

Notes:

- ▶ To run, verification requires 4 MB of available space per node in the /var file system. Typically, the /var/hacmp/clverify/clverify.log files require an extra 1 - 2 MB of disk space. At least 42 MB of available space is a best practice for a 4-node cluster.
- ▶ The default log file location for most PowerHA log files is now /var/hacmp, but there are some exceptions. For more information, see [PowerHA Administration Guide](#).

7.6.6 Running automatic corrective actions during verification

With PowerHA, some errors can be automatically corrected during cluster verification. The default action mainly depends on the state of the cluster, active or inactive, and if you are using the advanced option that is described in “Custom Cluster Configuration” on page 301.

The automatic corrective action feature can correct only some types of errors, which are detected during the cluster verification. The following errors can be addressed:

- ▶ PowerHA shared VG timestamps are outdated on a node.
- ▶ The /etc/hosts file on a node does not contain all PowerHA-managed IP addresses.
- ▶ A file system is not created on a node, although disks are available.
- ▶ A file systems auto-mount is enabled.
- ▶ Disks are available, but the VG is imported to a node.
- ▶ Shared VGs that are configured as part of a PowerHA RG have their automatic varyon attribute set to Yes.
- ▶ Required /etc/services entries are missing on a node.
- ▶ Required PowerHA snmpd entries are missing on a node.
- ▶ Required PowerHA network options settings.
- ▶ Corrective actions when using IP version 6 (IPv6).

With no prompting:

- ▶ Correct error conditions that appear in /etc/hosts.
- ▶ Correct error conditions that appear in /usr/es/sbin/cluster/etc/clhosts.client.
- ▶ Update /etc/services with missing entries.
- ▶ Update /etc/snmpd.peers and /etc/snmp.conf files with missing entries.

With prompting:

- ▶ Update auto-varyon on this VG.
- ▶ Update VG definitions for this VG.
- ▶ Keep PowerHA VG timestamps in sync with the VGDA.
- ▶ Auto-import VGs.
- ▶ Reimport VGs with missing file systems and mount points.
- ▶ The file system automount flag is set in /etc/filesystems.
- ▶ Set a network option.
- ▶ Set inoperative cluster nodes interfaces to the boot time interfaces.
- ▶ Bring active resources offline.

- ▶ Update automatic error notification stanzas.
- ▶ Perform a corrective action of starting **ntpd** daemons.
- ▶ Perform a corrective action of assigning link-local addresses to ND-capable network interfaces.

Enabling and disabling auto-corrective actions

You can use the auto-corrective actions through both SMIT and the CLI. For more information about the SMIT option, see “Custom Cluster Configuration” on page 301.

For the **clmgr** CLI interface, the option for auto-corrective actions is an attribute that is known as FIX. The options that are available for use with this attribute vary based on what exact action is being invoked as shown in Example 7-32.

Example 7-32 Clmgr auto-corrective options through the FIX attribute

```
clmgr verify cluster FIX={no|yes}
clmgr sync cluster FIX={no|yes}
clmgr online cluster FIX={no|yes|interactively}
clmgr online site FIX={no|yes|interactively}
clmgr online node FIX={no|yes|interactively}
```

7.6.7 Automatic cluster verification

PowerHA provides automatic verification in the following cases:

- ▶ Each time that you start cluster services on a node.
- ▶ Every 24 hours (automatic cluster configuration monitoring, which is enabled by default).

During automatic verification, PowerHA detects and, if you use the auto-correct option during startup, corrects several common configuration issues. This automatic behavior ensures that if you did not manually verify and synchronize a node in your cluster before starting cluster services, PowerHA does so.

By using the SMIT menus, you can set the parameters for the periodic automatic cluster verification checking utility by running **smitty sysmirror** and selecting **Problem Determination Tools** → **PowerHA SystemMirror Verification** → **Automatic Cluster Configuration Monitoring**.

The following fields are in the SMIT panel:

- ▶ Automatic cluster configuration verification: Select Disable or Enable.
- ▶ Node name: Select nodes where the utility will run. Selecting Default means that the first node, in alphabetical order, verifies the configuration.
- ▶ HOUR (00 - 23): Defines the time for when the utility will start. The default value is 00 (midnight), and it can be changed manually.
- ▶ Debug: Select Yes to enable or No to disable debug mode.

Figure 7-33 shows the SMIT panel for the Automatic Cluster Configuration Monitoring parameters setting for running `smitty clautover.dialog`.

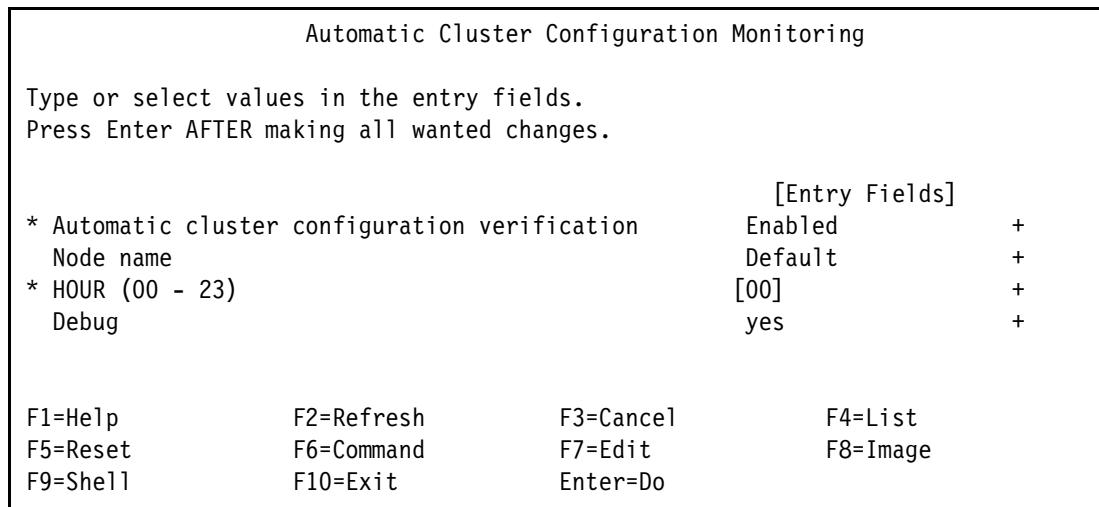


Figure 7-33 Automatic Cluster Configuration Monitoring

You can check the verification result of automatic cluster verification in the `autoverify.log` file in the `/var/hacmp/log` directory. For more information about general verification log files, see 7.6.5, “Verification log files” on page 309.

7.7 Monitoring PowerHA

PowerHA provides a highly available application environment by masking or eliminating failures that might occur either on hardware or software components of the environment. Masking the failure means that the active resources are moved from a failed component to the next available component of that type. So, all highly available applications continue to operate and clients can access and use them despite the failure.

As a result, it is possible that a component in the cluster failed and that you are unaware of the fact. The danger here is that while PowerHA can survive one or possibly several failures, each failure that escapes your notice threatens the cluster's ability to provide a highly available environment because the redundancy of the cluster components is diminished.

To avoid this situation, regularly check and monitor the cluster. PowerHA offers various utilities to help you with cluster monitoring and other items:

- ▶ Automatic cluster verification (see 7.6.7, “Automatic cluster verification” on page 311)
- ▶ Cluster status checking utilities
- ▶ RG information commands
- ▶ Topology information commands
- ▶ Log files
- ▶ Error notification methods
- ▶ Application monitoring
- ▶ Measuring application availability
- ▶ Monitoring clusters from the enterprise system administration and monitoring tools

You can use either ASCII SMIT, PowerHA SMUI, or the `c1mgr` CLI to configure and manage cluster environments.

7.7.1 Cluster status checking utilities

Several common status checking utilities are available.

SMUI

The PowerHA SMUI provides a single interface to monitor the status of multiple clusters in the enterprise. For a demonstration that includes installing and using the PowerHA SMUI, see this [YouTube video](#).

The **clstat** command

The **clstat** command in `/usr/es/sbin/cluster/clstat` is a helpful tool that you can use for cluster status monitoring. It uses the **clinfo** library routines to display information about the cluster, including the name and state of the nodes, networks, network interfaces, and RGs.

To use this utility, the **clinfoES** subsystem must be active on nodes where the **clstat** command is initiated.

The **clstat** command is supported in two modes: ASCII mode and X Window mode. ASCII mode can run on any physical or virtual ASCII terminal, including xterm or aixterm windows. If the cluster node runs graphical X Window mode, **clstat** displays the output in a graphical window. Before running the command, ensure that the DISPLAY variable is exported to the X server, and that X client access is allowed.

Figure 7-34 shows the syntax of the **clstat** command.

```
clstat [-c cluster ID | -n cluster_name] [-i] [-r seconds] [-a|-o] [-s]

-c cluster ID > run in automatic (non-interactive) mode for the
                  specified cluster.
-n name          > run in automatic (non-interactive) mode for the
                  specified cluster name.
-i              > run in interactive mode
-r seconds      > number of seconds between each check of the
                  cluster status
-a              > run ascii version.
-o              > run once and exit.
-s              > display both up and down service labels.
-h              > display help for clstat configuration issues.
```

Figure 7-34 The **clstat** command syntax

Consider the following information about the **clstat** command in the figure:

- ▶ **clstat -a** runs the program in ASCII mode.
- ▶ **clstat -o** runs the program once in ASCII mode and exits (useful for capturing output from a shell script or cron job).
- ▶ **clstat -s** displays service labels that are both up and down; otherwise, it displays only service labels, which are active.

Example 7-33 shows the `clstat -o` command output from our test cluster.

Example 7-33 The clstat -o command output

```
# clstat -o

          clstat - Cluster Status Monitor
          -----
Cluster: glvmttest      (1236788768)
Thu Mar 26 15:25:57 EDT 2009
          State: UP           Nodes: 2
          SubState: STABLE

          Node: glvm1          State: UP
          Interface: glvm1 (2)      Address: 9.12.7.4
          State: UP
          Interface: glvm1_ip1 (1)    Address: 10.10.30.4
          State: UP
          Interface: glvm1_data1 (0)   Address: 10.10.20.4
          State: UP
          Resource Group: liviu       State: Online
          Resource Group: rosie       State: Online

          Node: glvm2          State: UP
          Interface: glvm2 (2)      Address: 9.12.7.8
          State: UP
          Interface: glvm2_ip2 (1)    Address: 10.10.30.8
          State: UP
          Interface: glvm2_data2 (0)   Address: 10.10.20.8
          State: UP
          Resource Group: liviu       State: Online (Secondary)
          Resource Group: rosie       State: Online (Secondary)
```

The `cldump` command

The `cldump` command in `/usr/es/sbin/cluster/utilities/cldump` provides a snapshot of the key cluster status components: the cluster, the nodes in the cluster, the networks and network interfaces that are connected to the nodes, and the RG status on each node.

The `cldump` command does not have any arguments, so you run `cldump` from the CLI.

Troubleshooting common SNMP problems

Both `clstat` and `cldump` commands depend on SNMP. The default in newer levels of AIX is snmpv3, which often causes these utilities to not function properly after initial installation. The following information can help you modify the configuration to get these utilities to work. It can help you to resolve the two common SNMP problems. Usually, fixing these problems solves the issues and you might not need to go through the other sections.

Tip: Common issues with possible resolutions are available in the help output of both `clstat` and `cldump` through the `-h` flag of each command.

Access permission

Check for access permission to the PowerHA portion of the SNMP Management Information Base (MIB) in the SNMP configuration file by completing the following steps:

1. Find the defaultView entries in the /etc/snmpdv3.conf file, which is shown in Example 7-34.

Example 7-34 The defaultView entries in the snmpdv3.conf file

```
# grep defaultView /etc/snmpdv3.conf
#VACM_VIEW defaultView      internet          - included -
VACM_VIEW defaultView      1.3.6.1.4.1.2.2.1.1.1.0 - included -
VACM_VIEW defaultView      1.3.6.1.4.1.2.6.191.1.6 - included -
VACM_VIEW defaultView      snmpModules        - excluded -
VACM_VIEW defaultView      1.3.6.1.6.3.1.1.4 - included -
VACM_VIEW defaultView      1.3.6.1.6.3.1.1.5 - included -
VACM_VIEW defaultView      1.3.6.1.4.1.2.6.191 - excluded -
VACM_ACCESS group1 -- noAuthNoPriv SNMPv1 defaultView - defaultView -
VACM_ACCESS director_group -- noAuthNoPriv SNMPv2c defaultView - defaultView -
```

Beginning with AIX 7.1, as a security precaution, the snmpdv3.conf file is included with the Internet access commented out (#). The preceding example shows the unmodified configuration file; the Internet descriptor is commented out, which means that there is no access to most of the MIB, including the PowerHA information. Other included entries provide access to other limited parts of the MIB. By default in AIX 7.1 and later, the PowerHA SNMP-based status commands work only if you edit the snmpdv3.conf file. The two ways to provide access to the PowerHA MIB are by modifying the snmpdv3.conf file as follows:

- Uncomment (remove the #) from the following Internet line, which provides access to the entire MIB:
VACM_VIEW defaultView internet - included -
- If you do not want to provide access to the entire MIB, add the following line, which provides access to only the PowerHA MIB:
VACM_VIEW defaultView risc6000c1smuxpd - included -

2. After editing the SNMP configuration file, stop and restart **snmpd**, and then refresh the cluster manager by using the following commands:

```
stopsrc -s snmpd
startsrc -s snmpd
refresh -s clstrmgrES
```

3. Test the SNMP-based status commands again. If the commands work, you do not need to go through the rest of the section.

IPv6 entries

If you use PowerHA SystemMirror 7.1.2 or later, check for the correct IPv6 entries in the configuration files for **clinfoES** and **snmpd**. In PowerHA 7.1.2, an entry is added to the /usr/es/sbin/cluster/etc/c1hosts file to support IPv6. However, the required corresponding entry is not added to the /etc/snmpdv3.conf file, which causes intermittent problems with the **c1stat** command.

The following two ways address this problem:

- ▶ If you do not plan to use IPv6:
 - a. Comment the line in the /usr/es/sbin/cluster/etc/clhosts file and then restart clinfoES by using the following commands:

```
# ::1      # PowerHA SystemMirror
stopsrc -s clinfoES
startsrc -s clinfoES
```
 - b. Try the SNMP-based status commands again. If the commands work, you do not need to go through the remainder of this section.
- ▶ If you plan to use IPv6 in the future:
 - a. Add the following line to the /snmpdv3.conf file:

```
COMMUNITY public    public    noAuthNoPriv :: 0      -
```
 - b. If you are using a different community (other than public), substitute the name of that community for the word public.
 - c. After editing the SNMP configuration file, stop and restart snmpd, and then refresh the cluster manager by using the following commands:

```
stopsrc -s snmpd
startsrc -s snmpd
refresh -s clstrmgrES
```
 - d. Try the SNMP-based status commands again.

Tip: For more information about how to customize SNMP from the default public community, see [Configuring SNMP with a nonpublic community name for use with clstat in a PowerHA for AIX environment](#).

Checking the cluster subsystem status

You can check the PowerHA or Reliable Scalable Cluster Technology (RSCT) subsystem status by running the `lssrc` command with the `-s` or `-g` switches. The command displays the subsystem name, group, PID, and status (active or inoperative). Examples of using the command are as follows:

- ▶ Display subsystem information for a specific subsystem:
`lssrc -s subsystem_name`
- ▶ Display subsystem information for all subsystems in a specific group:
`lssrc -g subsystem_group_name`

Note: After PowerHA is installed and configured, the Cluster Manager daemon (clstrmgrES) starts automatically at boot time. The Cluster Manager must be running before any cluster services can start on a node. Because the clstrmgrES daemon is now a long-running process, you cannot use the `lssrc -s clstrmgrES` command to determine the state of the cluster. Use the following command to check the clstrmgr state:

```
lssrc -ls clstrmgrES
```

The clshowsrv command

You can display the status of PowerHA subsystems by using the `clshowsrv` command (`/usr/es/sbin/cluster/utilities/clshowsrv`). It displays the status of all subsystems that are used by PowerHA, or the status of a selected subsystem. The command output format is the same as the `lssrc -s` command output.

Figure 7-35 shows the syntax of the **clshowsrv** command.

```
clshowsrv { -a | -v | subsystem ...}
```

Figure 7-35 The *clshowsrv* command syntax

Here are some examples of the command:

- ▶ Display the status of PowerHA subsystems: `clstrmgrES`, `clinfoES`, and the CAA subsystem of cluster communications daemon (`clcomd`):

```
clshowsrv -a
```

- ▶ Display the status of all PowerHA, RSCT, and CAA subsystems:

```
clshowsrv -v
```

Example 7-35 shows the output of the **clshowsrv -v** command from our test cluster when cluster services are running.

Example 7-35 The *clshowsrv -v* command output

```
Local node: "jessica" ("jessica", "jessica")
    Cluster services status: "NORMAL" ("ST_STABLE")
    Remote communications: "UP"
    Cluster-Aware AIX status: "UP"

Remote node: "jordan" ("jordan", "jordan")
    Cluster services status: "NORMAL" ("ST_STABLE")
    Remote communications: "UP"
    Cluster-Aware AIX status: "UP"

Status of the RSCT subsystems used by PowerHA SystemMirror:
Subsystem      Group          PID      Status
  cthags        cthags        20840832  active
  ctrmc         rsct          16843220  active

Status of the PowerHA SystemMirror subsystems:
Subsystem      Group          PID      Status
  clstrmgrES   cluster       22872528  active
  clevmgrdES   cluster       25493856  active

Status of the CAA subsystems:
Subsystem      Group          PID      Status
  clconfd      caa           19661210  active
  clcomd       caa           19005808  active
```

You can also view the **clshowsrv -v** output through the SMIT menus by running **smitty sysmirror** and selecting **System Management (C-SPOC) → PowerHA SystemMirror Services → Show Cluster Services**.

7.7.2 Other cluster monitoring tools

Monitoring a PowerHA clustered environment often varies depending on what status information is reported and how it is reported (for example, CLI output, athena browser, or enterprise SNMP software) when obtaining the cluster status. The IBM **c1stat** facility is provided as a compiled binary file:

- ▶ It cannot be customized in any way.
- ▶ It can be run only from an AIX OS partition.
- ▶ It provides basic information regarding node, adapter, and RG status.

Enterprise monitoring solutions such as IBM Tivoli Monitoring are often complex, have cost implications, and might not provide the information that you require in a format that you require. An effective solution is to write your own custom monitoring scripts that are tailored for your environment.

The Query HA (**qha**) tool is publicly available, but it is not included with the PowerHA software.

Note: Custom examples of **qha** and other tools are in the *Guide to IBM PowerHA SystemMirror for AIX Version 7.1.3, SG24-8167*.

Query HA (**qha**)

The **qha** tool was created in 2001. It was updated and works on levels up to PowerHA 7.2.7 (at the time of writing). It primarily provides an in-cluster status view, which does not rely on the SNMP protocol and **clinfo** infrastructure. Query HA can also be easily customized.

Rather than reporting about whether the cluster is running or unstable, the tool's focus is on the internal status of the cluster manager. Although not officially documented, Chapter 6, "Cluster maintenance" on page 195 provides a list of the internal **clstrmgr** states. This status information helps you understand what is happening within the cluster, especially during event processing (with cluster changes such as start, stop, RGs moves, application failures, and more). When you compare this information next to other information, such as the running event, the RG status, online network interfaces, and the varied on VGs, **qha** provides an excellent overall status view of the cluster. It also helps with problem determination, and understanding PowerHA event flow during, for example, **node_up** or **fallover** events and when searching through cluster and **hacmp.out** files.

Note: The **qha** tool may be downloaded from [QHA - PowerHA Cluster Status Utility](#).

Example 7-36 shows a sample status output from the **qha -nevmc** command.

Example 7-36 A qha status output

```
Cluster: redbook_cluster (7270)
          21:26:37 270ct22

jordan    iState: ST_STABLE
bdbrg      ONLINE      ()
CAA Unicasting: (UP IPv4 10.2.30.83->10.2.30.84)
CAA SAN Comms: | DISK Comms: UP
en0 hasvc jordan
- leevg(4) leeconcvg(10) vgsaplocal(3) -

jessica   iState: ST_STABLE
CAA Unicasting: (UP IPv4 10.2.30.84->10.2.30.83)
CAA SAN Comms: | DISK Comms: UP
```

```
en0 jessica  
- vgsaplocal(9) -
```

7.7.3 Topology information commands

Use the following topology information commands to learn about the cluster topology and attributes that are associated with the CAA cluster configuration:

- ▶ **cltopinfo**
- ▶ **lscluster**

The **cltopinfo** command

The **cltopinfo** command (`/usr/es/sbin/cluster/utilities/cltopinfo`) lists the cluster topology information by using a format that is simple to read and understand.

Figure 7-36 shows the **cltopinfo** command syntax.

```
cltopinfo [-c] [-n] [-w] [-i]
```

Figure 7-36 The **cltopinfo** command syntax

The command options are as follows:

- | | |
|---------------------|--|
| cltopinfo -c | Shows the cluster name and the security mode (Standard or Enhanced). |
| cltopinfo -i | Shows all interfaces that are configured in the cluster. The information includes the interface label, the network to which it is attached (if appropriate), the IP address, netmask, prefix length, node name, and device name. |
| cltopinfo -n | Shows all nodes that are configured in the cluster: for each node, a list of all defined networks; for each network, all defined interfaces. |
| cltopinfo -w | Shows all networks that are configured in the cluster: for each network, lists all nodes that are attached to that network; for each node, lists the defined interfaces. |

You can also use SMIT menus to display various formats of the topology information:

- ▶ Display by cluster, as shown in Example 7-37:

To display the same output as shown by the default **cltopinfo** command, run **smitty sysmirror** and select **Cluster Nodes and Networks → Manage the Cluster → Display PowerHA SystemMirror Configuration**.

Example 7-37 Cltopinfo output through SMIT

```
Cluster Name: redbook_cluster
Cluster Type: Standard
Heartbeat Type: Unicast
Repository Disk: hdisk0 (00f92db1ba302638)
```

There are 2 nodes and 1 network defined

```
NODE jordan:
    Network net_ether_01
        hasvc   10.2.30.183
        jordan   10.2.30.83
```

```

NODE jessica:
    Network net_ether_01
        hasvc 10.2.30.183
        jessica 10.2.30.84

Resource Group bdbrg
    Startup Policy Online on Home Node Only
    Fallback Policy Fallback To Next Priority Node In The List
    Fallback Policy Never Fallback
    Participating Nodes jordan jessica
    Service IP Label hasvc

```

- ▶ Display by node, as shown in Example 7-38.

To display the same output as shown by the **cltopinfo -n** command, run **smitty sysmirror** and select **Cluster Nodes and Networks → Manage Nodes → Show Topology Information by Node → Show All Nodes**.

Example 7-38 Cltopinfo -n output through SMIT

```

NODE jordan:
    Network net_ether_01
        hasvc 10.2.30.183
        jordan 10.2.30.83

NODE jessica:
    Network net_ether_01
        hasvc 10.2.30.183
        jessica 10.2.30.84

```

- ▶ Display by network, as shown in Example 7-39.

To display the same output as shown by **cltopinfo -w**, run **smitty sysmirror** and select **Cluster Nodes and Networks → Manage Networks and Network Interfaces → Show Topology Information by Network → Show All Networks**.

Example 7-39 Cltopinfo -w output through SMIT

```

Network net_ether_01
    NODE jordan:
        hasvc 10.2.30.183
        jordan 10.2.30.83
    NODE jessica:
        hasvc 10.2.30.183
        jessica 10.2.30.84

```

- ▶ Display by network interface, as shown in Example 7-40.

To display the same output as shown by **cltopinfo -i**, run **smitty sysmirror** and select **Cluster Nodes and Networks → Manage Networks and Network Interfaces → Show Topology Information by Network Interface → Show All Network Interfaces**.

Example 7-40 Cltopinfo -i output through SMIT

IP Label	Network	Type	Node	Address	If	Netmask	Prefix Length
hasvc	net_ether_01	ether	jordan	10.2.30.183	en0	255.255.255.0	24
jordan	net_ether_01	ether	jordan	10.2.30.83	en0	255.255.255.0	24
hasvc	net_ether_01	ether	jessica	10.2.30.183	en0	255.255.255.0	24
jessica	net_ether_01	ether	jessica	10.2.30.84	en0	255.255.255.0	24

The lscluster CAA command

This **lscluster** command lists the attributes that are associated with the CAA cluster configuration. Figure 7-37 shows the syntax.

```
lscluster { -i | -d | -c [ -n clustername ] } | { -m [ nodename ] | -s | -i  
interfacename | -d diskname }
```

Figure 7-37 The *lscluster* command syntax

The command options are as follows:

- | | |
|---------------------|---|
| lscluster -c | Lists the cluster configuration. |
| lscluster -d | Lists the cluster storage interfaces. |
| lscluster -g | Lists the currently active network gateway interfaces as reported by each node. |
| lscluster -i | Lists the network device driver (NDD) and pseudo-NDD interfaces that are configured on each of the CAA nodes. CAA might not use all the interfaces to exchange heartbeat packets. The storage framework communication (sfwcom) interface is displayed as UP only if this interface is configured and available. Otherwise, it is not displayed. |
| lscluster -k | Lists the assigned network gateway interfaces for each node. The interfaces can be down or not even configured. You can specify this flag only with the -g flag. |
| lscluster -m | Lists the cluster node configuration information. This information includes a list of points of contact. Points of contact are cluster configuration interfaces that are used by the cluster to exchange heartbeat packets. If a point of contact has no CAA traffic for an extended period, it is removed from the list of points of contact. |
| lscluster -n | Allows the cluster names to be queried for all interfaces, storage, or cluster configurations (applicable only with -i , -d , or -c flags). |
| lscluster -s | Lists the cluster network statistics on the local node. |

Example 7-41 shows a sample output from the **lscluster** command.

Example 7-41 The *lscluster* command output

```
[jessica:root] / # lscluster -m
Calling node query for all nodes...
Node query number of nodes examined: 2

        Node name: jordan
        Cluster shorthand id for node: 1
        UUID for node: fe8cd108-5286-11ed-8013-96d7548b5a02
        State of node: UP NODE_LOCAL
                    Reason: NONE
        Smoothed rtt to node: 0
        Mean Deviation in network rtt to node: 0
        The number of clusters node is a member in: 1
        CLUSTER NAME      SHID      UUID
        redbook_cluster   0         fea44b80-5286-11ed-8013-96d7548b5a02
        SITE NAME         SHID      UUID
        LOCAL             1         51735173-5173-5173-5173-517351735173

        Points of contact for node: 0
```

```

-----  

Node name: jessica  

Cluster shorthand id for node: 2  

UUID for node: fe8cd14e-5286-11ed-8013-96d7548b5a02  

State of node: UP  

    Reason: NONE  

Smoothed rtt to node: 7  

Mean Deviation in network rtt to node: 3  

The number of clusters node is a member in: 1  

CLUSTER NAME      SHID      UUID  

redbook_cluster    0         fea44b80-5286-11ed-8013-96d7548b5a02  

SITE NAME          SHID      UUID  

LOCAL              1         51735173-5173-5173-5173-517351735173  

Points of contact for node: 1  

-----  

Interface   State  Protocol  Status  SRC_IP->DST_IP  

-----  

tcpsock->02   UP     IPv4       none    10.2.30.83->10.2.30.84
-----
```

7.7.4 Resource group information commands

Use the following commands to get information about RGs:

- ▶ **cIRGinfo**
- ▶ **cfindres**

The **cIRGinfo** command

You can get attribute information about RGs that are in the cluster by using the **cIRGinfo** command (`/usr/es/sbin/cluster/utilities/cIRGinfo`). The output shows the location and state of one or more specified RGs. The output also displays the global state of the RG and the special state of the RG on a local node.

Figure 7-38 shows the **cIRGinfo** command syntax.

```
cIRGinfo [-h] [-v] [-s|-c] [-t] [-p] [-a] [-m] [i] [resgroup1] [resgroup2]
```

Figure 7-38 The cIRGinfo command syntax

The command options are as follows:

- | | |
|--------------------|---|
| cIRGinfo -h | Displays the usage message. |
| cIRGinfo -v | Produces verbose output. |
| cIRGinfo -a | Displays the location of an RG and its destination after a cluster event. Use this flag in pre-event and post-event scripts, especially in PowerHA SystemMirror clusters that have dependent RGs. When PowerHA SystemMirror processes dependent RGs, multiple RGs can be moved at once with the <code>rg_move</code> event. |
| cIRGinfo -t | Displays the delayed timer information, all delayed fallback timers, and settling timers that are active on the local node. This flag can be used only if the cluster manager is active on the local node. |
| cIRGinfo -s | Produces colon-separated output. |

c1RGinfo -c	Same as the -s option, that is, it produces colon-separated output.
c1RGinfo -p	Displays the node that temporarily has the highest priority for this instance.
c1RGinfo -m	Displays the status of application monitors in the cluster.
c1RGinfo -i	Displays any administrator directed online or offline operations.

The RG can be in the following status conditions:

Online	The RG is operating properly on one or more nodes in the cluster.
Offline	The RG is not operating in the cluster and is not in an error condition. The Offline state means that either the user requested this state or dependencies were not met.
Unmanaged	Cluster services were stopped with the Unmanage option.
Acquiring	An RG is coming up on one of the nodes in the cluster. In normal conditions, the status changes to Online.
Releasing	The RG is in the process of being released from ownership by one node. Under normal conditions, after being successfully released from a node, the RG's status changes to Offline.
Error	The RG reported an error condition. User interaction is required.
Temp Error	A recoverable error occurred.
Unknown	The RG's status cannot be obtained, possibly because of loss of communication; all nodes in the cluster might not be running; or an RG dependency is not met (another RG that depends on this RG failed to be acquired first).

If the cluster services are not running on the local node, the command determines a node where the cluster services are active and obtains the RG information from the active cluster manager.

Example 7-42 shows output of the **c1RGinfo** command.

Example 7-42 c1RGinfo command output

Group Name	Group State	Node
rosie	ONLINE	g1vm1@USA
	ONLINE SECONDARY	g1vm2@UK
liviu	ONLINE	g1vm1@USA
	ONLINE SECONDARY	g1vm2@UK

The **clfindres** command

An alternative to using the **c1RGinfo** command, you can use the **clfindres** command that is in /usr/es/sbin/cluster/utilities. The **clfindres** command is a link to the **c1RGinfo** command, so all the options and output are identical.

7.7.5 Log files

PowerHA writes the messages that it generates to the system console and to several log files. Because each log file contains a different subset of the types of messages that are generated by PowerHA, you can see different views of the cluster status by viewing separate log files.

The default locations of log files are used in this section. If you redirected any logs, check the appropriate location.

Viewing log files

By running `smitty cspoc` and selecting PowerHA SystemMirror Log, C-SPOC offers the following utilities to view log files:

- ▶ **View/Save/Remove PowerHA SystemMirror Event Summaries**
Display the contents, save the cluster event summary to a file, or remove the summary history.
- ▶ **View Detailed PowerHA SystemMirror Log Files**
Display or live view watch the PowerHA (detailed) scripts log (`/var/hacmp/log/hacmp.out`), the PowerHA event summary log (`/var/hacmp/adm/cluster.log`), or the C-SPOC system log file (`/var/hacmp/log/cspoc.log`).
- ▶ **Change>Show PowerHA SystemMirror Log File Parameters**
Set the formatting option (default, standard, `html-low`, or `html-high`) for the selected node.
- ▶ **Change>Show Cluster Manager Log File Parameters**
Set the `c1strmgrES` debug level (standard or high).
- ▶ **Change>Show a Cluster Log Directory**
Change the default directory for a log file and select a new directory.
- ▶ **Change>Show All Cluster Logs Directory**
Change the default directory for all log files and select a new directory.
- ▶ **Collect Cluster log files for Problem Reporting**
Collect the cluster log file snap data by using the `c1snap` command, which is necessary for additional problem determination and analysis. You can select the debug option here, include RSCT log files, and select nodes that are included in this data collection. If not specified, the default location for the snap collection is in the `/tmp/ibmsupt/hacmp` directory for `c1snap` and `/tmp/phoenix.snap0ut` for phoenix `snap`.
- ▶ **Change>Show Group Services Log File Size**
Change or show the maximum log file size for the group services daemon. The default is `-1`, which is unlimited.

List of log files

The PowerHA log files are as follows:

- ▶ `/usr/es/sbin/cluster/ui/agent/logs/smui-agent.log`
The `smui-agent.log` file contains information about the local agent that is installed on each PowerHA SystemMirror node.
- ▶ `/usr/es/sbin/cluster/ui/agent/logs/notify-event.log`
The `notify-event.log` file contains information about all PowerHA SystemMirror events that are sent from the SMUI agent to the SMUI server.

- ▶ `/usr/es/sbin/cluster/ui/agent/logs/agent_deploy.log`
 The `agent_deploy.log` file contains information about the deployment configuration of the SMUI agent on the local node.
- ▶ `/usr/es/sbin/cluster/ui/agent/logs/uiagent.log`
 The `uiagent.log` file contains information about the startup log of the agent on that node.
- ▶ `/usr/es/sbin/cluster/ui/server/logs/smui-server.log`
 The `smui-server.log` file contains information about the PowerHA SMUI server.
- ▶ `/usr/es/sbin/cluster/ui/server/logs/uiserver.log`
 The `uiserver.log` file contains information about the startup log of the SMUI server on that node.
- ▶ `/var/hacmp/adm/cluster.log`
 The `cluster.log` file is the main PowerHA log file. PowerHA error messages and messages about events that are related to PowerHA are appended to this log with the time and date at which they occurred.
- ▶ `/var/hacmp/adm/history/cluster.mmddyyyy`
 The `cluster.mmddyyyy` file contains time-stamped, formatted messages that are generated by PowerHA scripts. The system creates a cluster history file whenever cluster events occur, identifying each file by the file name extension `mmddyyyy` (where `mm` indicates the month, `dd` indicates the day, and `yyyy` indicates the year).
- ▶ `/var/log/clcomd/clcomd.log clcomd.log.n` (`n` indicates a number 1 - 6.)
 The `clcomd.log` file contains timestamped, formatted messages that are generated by the CAA communication daemon. This log file contains an entry for every connect request that is made to another node and the return status of the request.
- ▶ `/var/log/clcomd/clcomddiag.log` (`n` indicates a number 1 - 6.)
 The `clcomddiag.log` file contains timestamped, formatted messages that are generated by the CAA communication daemon when tracing is enabled. This log file is typically used by IBM Support personnel for troubleshooting.
- ▶ `/var/hacmp/clverify/clverify.log clverify.log.n` (`n` indicates a number 1 - 5.)
 The `clverify.log` file contains verbose messages that are output during verification. Cluster verification consists of a series of checks that are performed against various PowerHA configurations. Each check attempts to detect either a cluster consistency issue or an error. The verification messages follow a common, standardized format, where feasible, indicating such information as the nodes, devices, and command in which the error occurred.
- ▶ `/var/hacmp/clverify/ver_collect_d1par.log ver_collect_d1par.log.n` (`n` indicates a number 1 - 10.)
 The `ver_collect_d1par.log` file contains information that is gathered for and from Resource Optimized High Availability (ROHA) and dynamic logical partition (DLPAR) operations, and is generated from the `/usr/es/sbin/cluster/diag/ver_collect_d1par` script.
- ▶ `/var/hacmp/clverify/ver_odmclean_d1par.log ver_odmclean_d1par.log.n` (`n` indicates a number 1 - 10.)
 The `ver_odmclean_d1par.log` file contains output from a data collection cleaning request from running `/usr/es/sbin/cluster/diag/ver_collect_d1par -c`.

- ▶ `/var/hacmp/availability/clavailability.log`
 The `clavailability.log` contains detailed information about the statistics that are used by availability metrics tool.
- ▶ `/var/hacmp/log/autoverify.log`
 The `autoverify.log` file contains logging for auto-verify and auto-synchronize.
- ▶ `/var/hacmp/log/async_release.log`
 The `async_release.log` is created from the asynchronous process of a DLPAR operation during an acquire or release RG event.
- ▶ `/var/hacmp/log/clavan.log`
 The `clavan.log` file tracks when each application that is managed by PowerHA is started or stopped and when the node stops on which an application is running. By collecting the records in the `clavan.log` file from every node in the cluster, a utility program can determine how long each application has been up, and also compute other statistics that describe application availability time.
- ▶ `/var/hacmp/log/c1_event_summaries.txt`
 The `c1_event_summaries.txt` file contains event summaries that are pulled from the `hacmp.out` log file when the logs are cycled through the `c1cycle` cronjob. However, the file is not automatically truncated and can grow to be large over time. The `c1_event_summaries.txt` file can be cleared by running `smitty cm_dsp_evs` and selecting Delete Event Summary History. You might want to save a copy of the file first, which can also be done on the same initial menu.
- ▶ `/var/hacmp/log/c1configassist.log`
 The `c1configassist.log` file contains detailed information that is generated by the Two-Node Cluster Configuration Assistant.
- ▶ `/var/hacmp/log/c12siteconfig_assist.log`
 The `c12siteconfig_assist.log` file contains detailed information that is generated by the Two-Site Cluster Configuration Assistant.
- ▶ `/var/hacmp/log/clinfo.log clinfo.log.n` (n indicates a number 1 - 7.)
 The `clinfo.log` file is typically installed on both client and server systems. Client systems do not have the infrastructure to support log file cycling or redirection. The `clinfo.log` file records the activity of the `clinfo` daemon.
- ▶ `/var/hacmp/log/c1_testtool.log`
 The `testtool.log` file stores output from the test when you run the Cluster Test Tool from SMIT, which also displays the status messages.
- ▶ `/var/hacmp/log/cloudroha.log`
 This log contains any cloud-related authentication issues. It also has data on cloud operations such as query, acquire, or release that are performed on LPARs of IBM Power Virtual Servers.
- ▶ `/var/hacmp/log/c1passwd.log`
 This log contains debug information from the `/usr/es/sbin/cluster/utilities/c1_chpasswdutil` utility.
- ▶ `/var/hacmp/log/clstrmgr.debug clstrmgr.debug.n` (n indicates a number 1 - 7.)
 The `clstrmgr.debug` log file contains timestamped, formatted messages that are generated by Cluster Manager activity. This file is typically used only by IBM Support personnel.

- ▶ /var/hacmp/log/clstrmgr.debug.long clstrmgr.debug.long.n (n indicates a number 1 - 7.)

The `clstrmgr.debug.long` file contains high-level logging of cluster manager activity, in particular its interaction with other components of PowerHA and with RSCT, which event is running, and information about RGs (for example, state and actions to be performed, such as acquiring or releasing them during an event).

- ▶ /var/hacmp/log/clutils.log

The `clutils.log` file contains the results of the automatic verification that runs on one user-selectable PowerHA cluster node once every 24 hours.

When cluster verification completes on the selected cluster node, this node notifies the other cluster nodes with the following information:

- The name of the node where verification was run.
- The date and time of the last verification.
- Results of the verification.

The `clutils.log` file also contains messages about any errors that are found and actions that are taken by PowerHA for the following utilities:

- The PowerHA File Collections utility.
- The Two-Node Cluster Configuration Assistant.
- The Cluster Test Tool.
- The OLPW conversion tool.

- ▶ /var/hacmp/log/cspoc.log

The `cspoc.log` file contains the logs of C-SPOC commands that run on the local node.

- ▶ /var/hacmp/log/cspoc.log.long

The `cspoc.log` file contains the logs of C-SPOC commands with verbose logging enabled.

- ▶ /var/hacmp/log/cspoc.log.remote

The `cspoc.log.remote` file contains the logs of C-SPOC commands on remote nodes with the `ksh` option `xtrace` enabled (set `-x`). To enable this logging, you must set the following environment variable on the local node where the C-SPOC operation runs:

`VERBOSE_LOGGING_REMOTE=high`

This environment variable creates a log file on the remote node that is named `cspoc.log.remote`, and it contains a set of `-x` output from the operations run there. This file is useful in debugging failed LVM operations on the remote node.

- ▶ /var/hacmp/log/hacmp.out hacmp.out.n (n indicates a number 1 - 7.)

The `hacmp.out` file records the output that is generated by the event scripts as they run. This information supplements and expands on the information in the `/var/hacmp/adm/cluster.log` file. To receive verbose output, set the debug level runtime parameter to `high` (the default).

- ▶ /var/hacmp/log/loganalyzer/loganalyzer.log

This log is the output log file from the `clanalyze` tool.

- ▶ /var/hacmp/log/migration.log

The `migration.log` file contains high-level logs of cluster activity while the cluster manager on the local node operates in a migration state.

- ▶ /var/hacmp/log/clevents.log

The `clevents.log` file contains logs of the IBM Systems Director interface.

- ▶ `/var/hacmp/log/clver_collect_gmvg_data.log`
This log is used only with Geographic Logical Volume Manager (GLVM) configurations to gather disk and gmvg data.
- ▶ `/var/hacmp/log/dnssa.log`
The dnssa.log file contains logs of Smart Assist for DNS.
- ▶ `/var/hacmp/log/dhcpsa.log`
The dhcpsa.log file contains logs of Smart Assist for DHCP.
- ▶ `/var/hacmp/log/domino_server.log`
The domino_server.log file contains logs of the Smart Assist for Domino Server.
- ▶ `/var/hacmp/log/filenetsa.log.log`
The domino_server.log file contains logs of Smart Assist for IBM FileNet® P8.
- ▶ `/var/hacmp/log/memory_statistics.log`
This log contains output from the `/usr/es/sbin/cluster/utilities/cl_memory_statistics` utility to collect the memory statistics. It is invoked during cluster verification and synchronization.
- ▶ `/var/hacmp/log/oraclesa.log`
The oraclesa.log file contains logs of the Smart Assist for Oracle database facility.
- ▶ `/var/hacmp/log/oraappsa.log`
The oraappsa.log file contains logs of the Smart Assist for Oracle application facility.
- ▶ `/var/hacmp/log/printServersa.log`
The oraappsa.log file contains logs of the Smart Assist for Print Subsystem facility.
- ▶ `/var/hacmp/log/sa.log`
The sa.log file contains logs that are generated by Application Discovery of Smart Assist.
- ▶ `/var/hacmp/log/sapsa.log`
The sa.log file contains logs that are generated by Smart Assist for SAP Netweaver.
- ▶ `/var/hacmp/log/ihssa.log`
The ihssa.log file contains logs of Smart Assist for IBM HTTP Server.
- ▶ `/var/hacmp/log/sax.log`
The sax.log file contains logs of the IBM Systems Director Smart Assist facility.
- ▶ `/var/hacmp/log/tsm_admin.log`
The tsm_admin.log file contains logs of the Smart Assist for Tivoli Storage Manager admin center.
- ▶ `/var/hacmp/log/tsm_client.log`
The tsm_client.log file contains logs of the Smart Assist for Tivoli Storage Manager client.
- ▶ `/var/hacmp/log/tsm_server.log`
The tsm_server.log file contains logs of the Smart Assist for Tivoli Storage Manager server.
- ▶ `/var/hacmp/log/emuhaclmp.log`
This log is a legacy log for event emulation. Although the log is still created during installation, the event emulation capability is deprecated.

- ▶ `/var/hacmp/log/maxdbsa.log`
The `maxdbsa.log` file contains logs of Smart Assist for MaxDB.
- ▶ `/var/hacmp/log/hswizard.log`
The `hswizard.log` file contains logs of Smart Assist for SAP LiveCache Hot Standby.
- ▶ `/var/hacmp/log/wmqsa.log`
The `wmqsa.log` file contains logs of Smart Assist for IBM MQ.
- ▶ `/tmp/clconvert.log`
This file contains a record of the conversion progress when upgrading PowerHA. The file is created by the `cl_convert` utility.

Log space requirements

Cluster administrators should ensure that enough space is available for all log files in the file systems. The minimum amount of space that is required in `/var` depends on the number of nodes in the cluster. You can use the following estimations to help calculate the value for each cluster node:

- ▶ 2 MB should be available for writing the `clverify.log[0-9]` files.
- ▶ 4 MB per node is needed for writing the verification data from the nodes.
- ▶ 20 MB is needed for writing the `clcomd` log information.
- ▶ 1 MB per node is needed for writing the ODM cache data.

For example, for a 4-node cluster, you need the following amount of space in the `/var` file system:

$$2 + (4 \times 4) + 20 + (4 \times 1) = 42 \text{ MB}$$

Some additional log files that gather debug data might require more space in the `/var` file system. This requirement depends on other factors, such as the number of shared VGs and file system. Cluster verification issues a warning if not enough space is allocated to the `/var` file system. At the time of writing with PowerHA 7.2.7, the log verification check is what is shown in Example 7-43.

Example 7-43 Log space verification check for `/var`

WARNING: There is 564 MB of available space remaining on the file system: `/var` on node: jordan

This is less than the recommended available space of 1704 MB for the log files on that file system:

Viewing log files

By running `smitty cspoc` and selecting PowerHA SystemMirror Log, C-SPOC offers the following utilities to view log files:

- ▶ View/Save/Remove PowerHA SystemMirror Event Summaries
Display the contents, save the cluster event summary to a file, or remove the summary history.
- ▶ View Detailed PowerHA SystemMirror Log Files
Display or live view watch the PowerHA (detailed) scripts log (`/var/hacmp/log/hacmp.out`), PowerHA event summary log (`/var/hacmp/adm/cluster.log`), or C-SPOC system log file (`/var/hacmp/log/cspoc.log`).

- ▶ Change>Show PowerHA SystemMirror Log File Parameters
Set the formatting option (default, standard, html-low, or html-high) for the selected node.
- ▶ Change>Show Cluster Manager Log File Parameters
Set the `c1strmgrES` debug level (standard or high).
- ▶ Change>Show a Cluster Log Directory
Change the default directory for a log file and select a new directory.
- ▶ Change>Show All Cluster Logs Directory
Change the default directory for all log files and select a new directory.
- ▶ Collect Cluster log files for Problem Reporting
Collect cluster log file snap data (`c1snap`), which is necessary for additional problem determination and analysis. You can select the debug option here, include RSCT log files, and select nodes that are included in this data collection. If not specified, the default location for the snap collection is in the `/tmp/ibmsupt/hacmp/` directory for `c1snap` and `/tmp/phoenix.snapOut` for phoenix `snap`.
- ▶ Change>Show Group Services Log File Size
Change or show the maximum log file size for the group services daemon. The default is -1, which is unlimited.

Changing the log directories

To change the default directory of a specific log file in the SMIT menu, run `smitty cspoc`, select **PowerHA SystemMirror Log → Change>Show a Cluster Log Directory**, and then select the log file to change.

The SMIT fast path is `smitty clusterlog_redir.select`. The default log directory is changed for all nodes in the cluster. The cluster should be synchronized after changing the log parameters.

Note: As a best practice, use only local file systems if you change the default log locations, rather than shared or NFS file systems. Having logs on shared or NFS file systems can cause problems if the file system must unmount during a failover event. Redirecting logs to shared or NFS file systems can also prevent cluster services from starting during node reintegration.

7.7.6 Using the `clanalyze` log analysis tool

The `clanalyze` tool (`/usr/es/sbin/cluster/clanalyze`) was introduced in PowerHA 7.2.2. It provides a simple way to search logs across all nodes in a cluster. It does require the logs to be in their default directory to find them. It provides the capability to do the following tasks:

- ▶ Analyzes the log files and provides an error report that is based on error strings or timestamps.
- ▶ Analyzes the core dump file from the AIX error log.
- ▶ Analyzes the log files that are collected through the `snap` and `c1snap` utilities.
- ▶ Analyzes a user-specified snap file based on error strings that are provided and generates a report.

The following analysis options are provided:

- ▶ Time window search: For example, an analysis of specific time slots.
- ▶ Error patterns that are supported for analysis (not case-sensitive in PowerHA 7.2.3 and later):
 - diskfailure
 - applicationfailure
 - interfacefailure
 - networkfailure
 - globalnetworkfailure
 - nodefailure
 - sitefailure
- ▶ Last error.
- ▶ All errors.

There is a progress indicator for a long-running analysis:

- ▶ The analysis can take some time if there is much data.
- ▶ The analysis process writes progress information to a file, and the progress indicator process reads and displays it.
- ▶ Granularity is limited but achieves the goal of demonstrating that the process is not hung. The progress indicator message looks like the following one:
49% analysis is completed. 150sec elapsed.

There is a sorted report for timeline comparison.

The tool also provides recommendations wherever possible. There are options to analyze both a live cluster or stored log files. For more information, see the [clanalyze command](#).

Examples

Example 7-44 shows the output from `clanalyze -a -p "Nodefailure"`. It also shows an example of when the tool cannot provide recommendations.

Example 7-44 Clanalyze node failure analysis output

```
# clanalyze -a -p "Nodefailure"
EVENT: Node Failure
-----
Time at which Node failure occurred: 2018-08-14T00:20:54
Node name: r1m2p31
Type of node failure: Information lost. Logs got
recycled
Description for node failure: SYSTEM SHUTDOWN BY USER
Probable cause for node failure: SYSTEM SHUTDOWN
Reason for node failure(0=SOFT IPL 1=HALT 2=TIME REBOOT): 0
Time at which Node failure occurred: 2018-08-07T06:10:46
Node name: r1m2p31
Type of node failure: Information lost. Logs got
recycled
Description for node failure: SYSTEM SHUTDOWN BY USER
Probable cause for node failure: SYSTEM SHUTDOWN
Reason for node failure(0=SOFT IPL 1=HALT 2=TIME REBOOT): 0
Note: Any field left blank indicates that the element does not exist in log files
Analysis report is available at
/var/hacmp/log/loganalyzer/analysis/report/2018-08-14/report.11534708
Analysis completed successfully
```

Example 7-45 shows a search for a disk failure.

Example 7-45 Clanalyze disk failure analysis output

```
# clanalyze -a -p "Diskfailure"
EVENT: Disk Failure
-----
Time at which Disk failure occurred: 2018-07-12T10:04:37
Node in which failure is observed: r1m2p31
RG affected due to failure: RG2
VG associated with affected RG: VG2
Disks responsible for failure: hdisk4
Details on CAA Repository disk failures
Date/Time: Wed Jul 12 00:56:02 2018
Node ID: r1m2p31
Resource Name: hdisk2
Description: Local node cannot access the cluster repository disk.
Probable Causes: Cluster repository disk is down or not reachable.
Failure Causes: A hardware problem prevents local node from accessing the cluster
repository disk.
Recommended Actions:
The local node was halted to prevent data corruption.
Correct hardware problem that caused loss of access to cluster
repository disk.
ERRPT DETAILS on node:r1m2p31
-----
Details on LDMP_COMPLETE
Date/Time: Mon Jul 17 23:37:25 2018
Node ID: r1m2p31
Volume group: VG1
```

7.7.7 SMUI log file viewing

The PowerHA SMUI provides the capability to view and compare logs side by side, which can be beneficial, especially for troubleshooting.

The following logs can be viewed from within the SMUI.

- ▶ hacmp.out
- ▶ errpt
- ▶ clutils.log
- ▶ clverify.log
- ▶ autoverify.log
- ▶ clstrmgr.debug
- ▶ cluster.log

Each log file is color-coded for identification and comparison. Also, for log files that have multiple iterations (hacmp.out, hacmp.out.1, hacmp.out.2, and so on), they are consolidated into one large file.

To view the logs, choose a cluster, click the Logs header, choose a specific log file, and then choose a node. Repeat this process as needed and use the search function. The SMUI allows multiple log windows to be viewed simultaneously, as shown in Figure 7-39 on page 333.

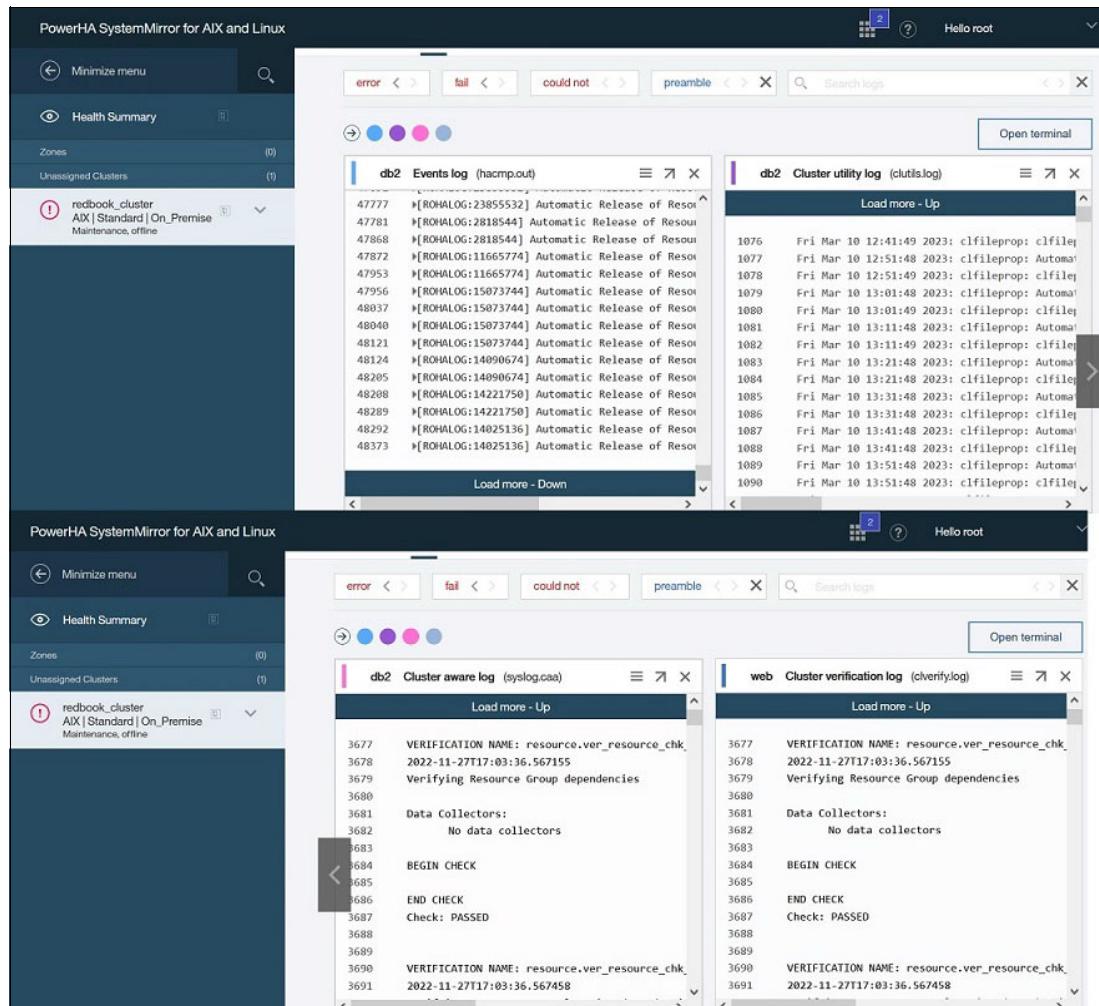


Figure 7-39 SMUI log viewing

7.7.8 Error notification

You can use the AIX Error Notification facility to add another layer of HA to a PowerHA environment. You can add notification for failures of resources for which PowerHA does not provide recovery by default.

For more information about automatic error notification, with examples of using and configuring it, see 11.6, “Automatic error notification” on page 488.

7.7.9 Application monitoring

PowerHA SystemMirror checks for running applications by using the configured application monitor.

Why we implement application monitors

By default, PowerHA with RSCT monitors the network infrastructure. Application health, beyond the availability of the network that is used by clients to get to the application, is not monitored. For this reason, configuring application monitors in PowerHA is an important consideration.

In addition, the introduction of the Unmanaged Resource Groups option, stopping cluster services (which leaves the applications running without cluster services) makes application monitors a crucial factor in maintaining application availability.

When cluster services are restarted to manage the RGs again, the process of acquiring resources checks each resource to determine whether it is online. If it is running, acquiring that resource is skipped.

For example, for the application running the server start script, this check is done by using an application monitor. The application monitor's returned status determines whether the application server start script runs.

What if no application monitor is defined? If that case, the cluster manager runs the application server start script, which might cause problems for applications that cannot deal with another instance being started, which might happen if the start script is run again when the application is already running.

Configuring application monitors

Two types of application monitors can be configured with PowerHA:

- ▶ *Process monitors* detect the termination of one or more processes of an application by using RSCT Resource Monitoring and Control (RMC). Any process that appears in the output of `ps -ef` can be monitored by using a PowerHA process monitor.
- ▶ *Custom monitors* check the health of an application with a user-written custom monitor method at user-specified polling intervals. With this monitor, the administrator may check for anything that can be defined as a determining factor in an application's health. A check can be a check for the ability to log in to an application, to open a database, to write a dummy record, or to query an application's internal state. A return code (RC) from the user-written monitor of 0 indicates that the application is healthy, and no further action is taken. A nonzero RC indicates that the application is not healthy and recovery actions will take place.

For each PowerHA application server that is configured in the cluster, you can configure up to 128 application monitors, but the total number of application monitors in a cluster cannot exceed 128.

Application monitors can be configured to run in various modes:

- ▶ Long-running mode
- ▶ Startup mode
- ▶ Both modes

In long-running mode, the monitor periodically checks that the application is running successfully. The checking frequency is set through the Monitor Interval. The checking begins after the Stabilization Interval expires, the RG that owns the application server is marked online, and the cluster is stabilized.

In startup mode, PowerHA checks the process (or calls the custom monitor) at an interval that is equal to one-twentieth of the Stabilization Interval of the startup monitor. The monitoring continues until the following events occur:

- ▶ The process is active.
- ▶ The custom monitor returns a 0.
- ▶ The Stabilization Interval expires.

If successful, the RG is put into the online state; otherwise, the cleanup method is invoked. In both modes, the monitor checks for the successful startup of the application server and periodically checks that the application is running successfully.

To configure an application monitor by using SMIT, complete the following steps:

1. Run **smitty sysmirror** and select **Cluster Applications and Resources → Resources → Configure User Applications (Scripts and Monitors) → Application Monitors**.
2. Select from either the **Configure Process Application Monitors** menu or the **Configure Custom Application Monitors** menu.

Tip: The SMIT fast path for the application monitor configuration is **smitty cm_appmon**.

Process application monitoring

The process application monitoring facility uses RMC, so the facility does not require any custom scripts. It detects only the application process termination and does not detect any other malfunction of the application.

When PowerHA finds that the monitored application process (or processes) is terminated, it tries to restart the application on the current node until a specified retry count is exhausted.

To add a new process application monitor by using SMIT, use one of the following approaches:

- ▶ Run **smitty sysmirror** and select **Cluster Applications and Resources → Resources → Configure User Applications (Scripts and Monitors) → Application Monitors → Configure Process Application Monitors → Add a Process Application Monitor**.
- ▶ Use the **smitty cm_appmon** fast path.

Figure 7-40 shows the SMIT panel with field entries for configuring an example process application monitor.

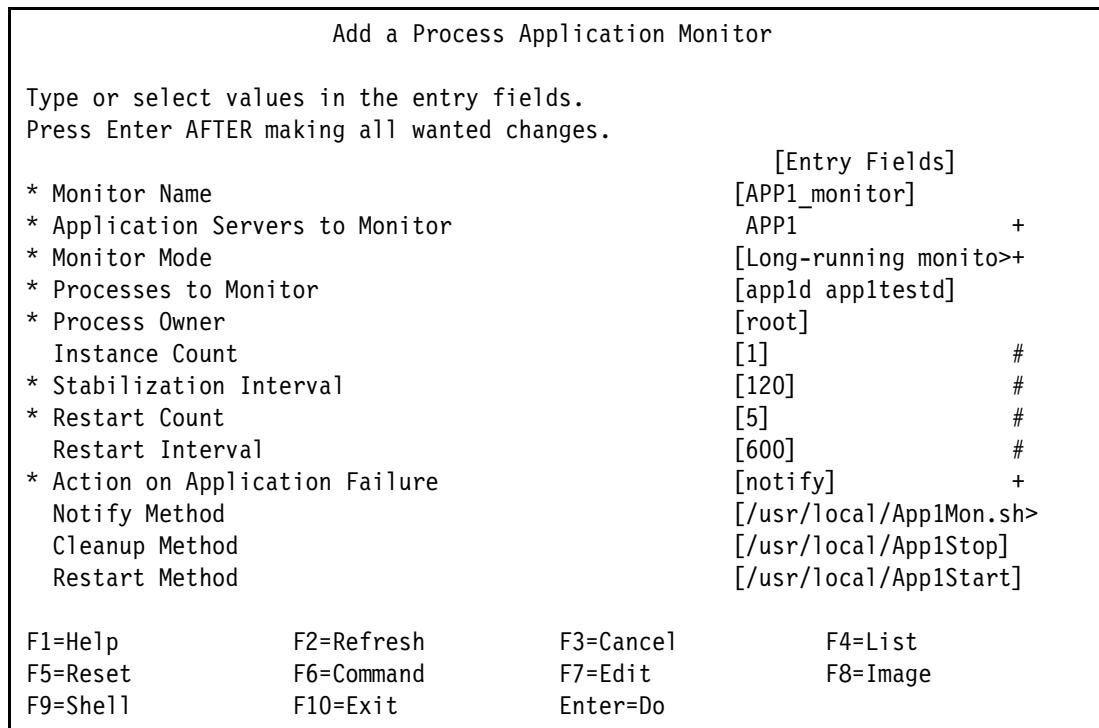


Figure 7-40 Adding process application monitor SMIT panel

In our example, the application monitor is called APP1_monitor and is configured to monitor the APP1 application server. The default monitor mode, Long-running monitoring, was selected. A Stabilization Interval of 120 seconds was selected.

Note: The Stabilization Interval is one of the most critical values in the monitor configuration. It must be set to a value that is long enough that if it expires, the application definitely failed to start. If the application is in the process of a successful start and the Stabilization Interval expires, cleanup is attempted and the RG is placed into the ERROR state. The consequences of the cleanup process vary by application, and the method might provide unwanted results.

The application processes that are monitored are app1d and app1testd. These processes must be in the output of a `ps -ef` command when the application is running so that they can be monitored through a process monitor. They are owned by root, and only one instance of each is expected to be running, as determined by the Process Owner and Instance Count values.

If the application fails, Restart Method is run to recover the application. If the application fails to recover to a running state after the number of restart attempts exceed Retry Count, Action on Application Failure runs. The action can be notify or failover. If notify is selected, no further action is taken after running the Notify Method. If failover is selected, the RG containing the monitored application moves to the next available node in the RG.

Cleanup Method and Restart Method define the scripts for stopping and restarting the application after a failure is detected. The default values are the start and stop scripts that are defined in the application server configuration.

Custom application monitoring

A custom application monitor offers another possibility to monitor the application availability by using custom scripts, which can simulate client access to the services. The scripts are provided by the application. Based on the exit code of this script, the monitor establishes whether the application is available or not. If the script exits with RC 0, then the application is available. Any other RC means that the application is not available.

To add a new custom application monitor by using SMIT, use one of the following approaches:

- ▶ Run `smitty sysmirror` and select **Cluster Applications and Resources** → **Resources** → **Configure User Applications (Scripts and Monitors)** → **Application Monitors** → **Configure Custom Application Monitors** → **Add a Custom Application Monitor**.
- ▶ Run the `smitty cm_cfg_custom_appmon` fast path.

The SMIT panel and its entries for adding this method into the cluster configuration are similar to the process application monitor add SMIT panel that is shown in Figure 7-40 on page 336.

The only different fields in the Configuring Custom Application Monitors SMIT menu are as follows:

Monitor Method	Defines the full path name for the script that provides a method to check the application status. If the application is a database, this script can connect to the database and run an SQL select sentence for a specific table in the database. If the result of the SQL select sentence is correct, it means that the database works normally.
Monitor Interval	Defines the time (in seconds) between each occurrence of Monitor Method being run.
Hung Monitor Signal	Defines the signal that is sent to stop the Monitor Method if it does not return within <i>Monitor Interval</i> seconds. The default action is SIGKILL(9).

How application monitors work

There are two aspects of application monitors to consider:

- ▶ Application Monitor Initial Status processing: How the status is determined, and in what order application monitors are considered for initial application status checking.
- ▶ Application Monitor General processing: The relationship and processing of long-running versus startup monitors.

Application Monitor Initial Status processing

As a protection mechanism, before the cluster manager invokes the application server start script, it uses an application monitor to determine the status of the application. This processing is done for each application server as follows:

- ▶ If no application monitor is defined for the application server or if the application monitor returns a failure status (RC!=0 for custom monitors, processes not running via RMC for process monitors), the application server start script is invoked.
- ▶ If the application monitor returns a success status (RC=0 for custom monitors, processes running via RMC for process monitors), the application server start script is not run.

If only one application monitor is defined for an application server, the process is as simple as stated previously.

If more than one application monitor is defined, the selection priority is based on the Monitor Type (custom or process) and the Invocation (both, long-running, or startup). The ranking of the combinations of these two monitor characteristics is as follows:

- ▶ Both, Process
- ▶ Long-running, Process
- ▶ Both, Custom
- ▶ Long-running, Custom
- ▶ Startup, Process
- ▶ Startup, Custom

The highest priority application monitor that is found is used to test the state of the application. When creating multiple application monitors for an application, be sure that your highest ranking monitor according to the foregoing list returns a status that can be used by the cluster manager to decide whether to invoke the application server start script.

When more than one application monitor meets the criteria as the highest ranking, the sort order is unpredictable (because qsort is used). However, it consistently produces the same result.

Fortunately, there is a way to test which monitor will be used. The routine that is used by the cluster manager to determine the highest ranking monitor is as follows:

```
/usr/es/sbin/cluster/utilities/cl_app_startup_monitor
```

An example of using this utility for the application server called testmonApp, which has three monitors that are configured, is as follows:

```
/usr/es/sbin/cluster/utilities/cl_app_startup_monitor -s testmonApp -a
```

The output for this command, which is shown in Example 7-46, shows three monitors:

- ▶ Mon: Custom, Long-running
- ▶ bothuser_testmon: Both, Custom
- ▶ longproctestmon: Process, Long-running

Example 7-46 Application monitor startup monitor

```
application = [testmonApp]
    monitor_name = [Mon]
    resourceGroup = [NULL]
    MONITOR_TYPE = [user]
    PROCESSES = [NULL]
    PROCESS_OWNER = [NULL]
    MONITOR_METHOD = [/tmp/longR]
    INSTANCE_COUNT = [NULL]
    MONITOR_INTERVAL = [60]
    HUNG_MONITOR_SIGNAL = [9]
    STABILIZATION_INTERVAL = [60]
    INVOCATION = [longrunning]

    application = [testmonApp]
    monitor_name = [bothuser_testmon]
    resourceGroup = [NULL]
    MONITOR_TYPE = [user]
    PROCESSES = [NULL]
    PROCESS_OWNER = [NULL]
    MONITOR_METHOD = [/tmp/Bothtest]
    INSTANCE_COUNT = [NULL]
```

```

MONITOR_INTERVAL = [10]
HUNG_MONITOR_SIGNAL = [9]
STABILIZATION_INTERVAL = [20]
INVOCATION = [both]

application = [testmonApp]
monitor_name = [longproctestmon]
resourceGroup = [NULL]
MONITOR_TYPE = [process]
PROCESSES = [httpd]
PROCESS_OWNER = [root]
MONITOR_METHOD = [NULL]
INSTANCE_COUNT = [4]
MONITOR_INTERVAL = [NULL]
HUNG_MONITOR_SIGNAL = [9]
STABILIZATION_INTERVAL = [60]
INVOCATION = [longrunning]

```

Monitor [longproctestmon] is detecting whether the application is running
queryProcessState - Called.

Arguments:

```

selectString=[ProgramName == "httpd" && Filter == "ruser == \"root\""]
expression=[Processes.CurPidCount < 4]

```

Application monitor [longproctestmon] exited with code (17)

Application monitor[longproctestmon] exited with code (17) - returning success

In the example, three monitors can be used for initial status checking. The highest ranking is the long-running process monitor, longproctestmon. Recall that the Monitor Type for custom monitors is user.

Note: A startup monitor is used for initial application status checking only if no long-running (or both) monitor is found.

Application monitor general processing

An initial application status check is performed by using one of the monitors. For more information, see “Application Monitor Initial Status processing” on page 337.

If necessary, the application server start script is invoked. Simultaneously, all startup monitors are invoked. Only when all the startup monitors indicate that the application has started by returning successful status is the application considered online (and can lead to the RG going to the ONLINE state).

The Stabilization Interval is the *timeout* period for the startup monitor. If the startup monitor fails to return a successful status, the application's RG goes to the ERROR state.

After the startup monitor returns a successful status, there is a short period during which the RG state moves to ONLINE, usually from ACQUIRING.

For each long-running monitor, the Stabilization Interval is allowed to elapse, and then the long-running monitor is invoked. The long-running monitor continues to run until a problem is encountered with the application.

If the long-running monitor returns a failure status, the retry count is examined. If it is nonzero, it is decremented, the Cleanup Method is invoked, and then the Restart Method is invoked. If the retry count is zero, the cluster manager processes either a failover event or a notify event, as determined by the Action on Application Failure setting for the monitor.

After the Restart Interval expires, the retry count is reset to the configured value.

Application monitor example

Here are the initial settings for the application monitors that are used in this example:

```
App mon name: Mon
Invocation: longrunning
Monitor method: /tmp/longR
Monitor interval: 60
Stab Interval: 60
Restart count: 3
Restart method: /tmp/Start (the application server start script)
Restart Interval: 396 (default)
```

```
App mon name: startup
Invocation: startup
Monitor method: /tmp/start-up
Monitor interval: 20
Stab Interval: 20
Restart count: 3
Restart method: /tmp/Start (the application server start script)
Restart Interval: 132 (default)
```

Application server start, stop, and monitor script functions

The restart method /tmp/Start sleeps for 10 seconds, then starts the “real” application (/tmp/App simulates a real application and is called in the background). The /tmp/App method writes the status of the monitored resources group to the log file every second for 20 seconds, and then sleeps for the balance of 10 minutes.

Both the /tmp/longR and /tmp/start-up methods check for /tmp/App in the process table. If /tmp/App is found in the process table, the RC is 0; if not found, the RC is 1.

The /tmp/Stop method finds and kills the /tmp/App process in the process table to cause a failure.

Scenario 1: Normal (no error) application start

Logging was sent to a common log that is shown in Example 7-47.

Example 7-47 Logging sent to a common log

```
/tmp/longR 12:24:47 App RC is 1
/tmp/Start 12:24:48 App start script invoked
/tmp/start-up 12:24:48 App RC is 1
/tmp/start-up 12:24:49 App RC is 1
/tmp/start-up 12:24:50 App RC is 1
/tmp/start-up 12:24:51 App RC is 1
/tmp/start-up 12:24:52 App RC is 1
/tmp/start-up 12:24:53 App RC is 1
/tmp/start-up 12:24:54 App RC is 1
/tmp/start-up 12:24:55 App RC is 1
/tmp/start-up 12:24:56 App RC is 1
```

```
/tmp/start-up 12:24:57 App RC is 1
/tmp/App 12:24:58 testmon ACQUIRING
/tmp/start-up 12:24:58 App RC is 0
/tmp/App 12:24:59 testmon ACQUIRING
/tmp/App 12:25:00 testmon ONLINE
/tmp/longR 12:26:00 App RC is 0
/tmp/longR 12:27:00 App RC is 0
```

What happened in Scenario 1

The following results were part of Scenario 1:

- ▶ The long-running monitor (/tmp/longR) was invoked, returning RC=1.
- ▶ The application server start script (/tmp/Start) and startup monitor (/tmp/start-up) were invoked 1 second later.
- ▶ The startup monitor returns RC=1, iterating every second (1/20 of the 20-second Stabilization Interval).
- ▶ After the programmed 10-second sleep, the start script launches the “real” application /tmp/App.
- ▶ The startup monitor finds /tmp/App running and returns 0.
- ▶ Five seconds later, PowerHA marks the RG online (other tests showed less than 5 seconds, but not consistently). At 60-second intervals after the RG is marked as ONLINE, the longR monitor is invoked, returning RC=0.

Conclusions from Scenario 1

We drew the following major conclusions from Scenario 1:

- ▶ A long-running monitor is used to perform an initial application status check over the startup monitor.
- ▶ A startup monitor is invoked simultaneously with the start script and monitors at 1/20 of the Stabilization Interval of the startup monitor (not the long-running monitor).
- ▶ A delay of 2 - 5 seconds exists between the time the startup monitor returns RC=0 and the RG is marked as ONLINE.
- ▶ The long-running monitor is invoked after a delay of exactly the Stabilization Interval.

Scenario 2: Application fails to launch

In this scenario, the application fails to launch within the Stabilization Interval (for example, the start-up monitor condition was never met).

Common logging is shown in Example 7-48.

Example 7-48 Common logging

```
/tmp/longR 11:32:52 App RC is 1
/tmp/Start 11:32:53 App start script invoked
/tmp/start-up 11:32:53 App RC is 1
/tmp/start-up 11:32:54 App RC is 1
/tmp/start-up 11:32:55 App RC is 1
/tmp/start-up 11:32:56 App RC is 1
/tmp/start-up 11:32:57 App RC is 1
/tmp/start-up 11:32:58 App RC is 1
/tmp/start-up 11:32:59 App RC is 1
/tmp/start-up 11:33:00 App RC is 1
/tmp/start-up 11:33:01 App RC is 1
```

```
/tmp/start-up 11:33:02 App RC is 1
/tmp/start-up 11:33:03 App RC is 1
/tmp/start-up 11:33:04 App RC is 1
/tmp/start-up 11:33:05 App RC is 1
/tmp/start-up 11:33:06 App RC is 1
/tmp/start-up 11:33:07 App RC is 1
/tmp/start-up 11:33:08 App RC is 1
/tmp/start-up 11:33:09 App RC is 1
/tmp/start-up 11:33:10 App RC is 1
/tmp/start-up 11:33:11 App RC is 1
/tmp/start-up 11:33:12 App RC is 1
/tmp/Stop 11:33:16 App stopped
```

What happened in Scenario 2

The following events took place in Scenario 2:

- ▶ The long-running monitor (/tmp/longR) was invoked and returned RC=1.
- ▶ The application server start script (/tmp/Start) and the startup monitor (/tmp/start-up) were invoked 1 second later.
- ▶ The startup monitor returned RC=1, iterating every second.
- ▶ After the 20-second startup monitor Stabilization Interval, the Cleanup Method (/tmp/Stop, which is also the application server stop script) was invoked.
- ▶ The RG went into an ERROR state.

To see what happened in more detail, the failure as logged in /var/hacmp/log/hacmp.out (on the final RC=1 from the start-up monitor) is shown Example 7-49.

Example 7-49 The failure as logged in hacmp.out

```
+testmon:start_server[start_and_monitor_server+102] RETURN_STATUS=1
+testmon:start_server[start_and_monitor_server+103] : exit status of
cl_app_startup_monitor is: 1
+testmon:start_server[start_and_monitor_server+103] [[ 1 != 0 ]]
+testmon:start_server[start_and_monitor_server+103] [[ false = true ]]
+testmon:start_server[start_and_monitor_server+109] cl_RMupdate resource_error
testmonApp start_server
2009-03-11T11:33:13.358195
2009-03-11T11:33:13.410297
Reference string: Wed.Mar.11.11:33:13.EDT.2009.start_server.testmonApp.testmon.ref
+testmon:start_server[start_and_monitor_server+110] echo ERROR: Application
Startup did not succeed.
ERROR: Application Startup did not succeed.
+testmon:start_server[start_and_monitor_server+114] echo testmonApp 1
+testmon:start_server[start_and_monitor_server+114] 1>>
/var/hacmp/log/.start_server.700610
+testmon:start_server[start_and_monitor_server+116] return 1
+testmon:start_server[+258] awk {
    if ($2 == 0) {
        exit 1
    }
}
+testmon:start_server[+258] cat /var/hacmp/log/.start_server.700610
+testmon:start_server[+264] SUCCESS=0
+testmon:start_server[+266] [[ REAL = EMUL ]]
```

```

+testmon:start_server[+266] [[ 0 = 1 ]]
+testmon:start_server[+284] awk {
    if ($2 == 1) {
        exit 1
    }
    if ($2 == 11) {
        exit 11
    }
}

+testmon:start_server[+284] cat /var/hacmp/log/.start_server.700610
+testmon:start_server[+293] SUCCESS=1
+testmon:start_server[+295] [[ 1 = 0 ]]
+testmon:start_server[+299] exit 1
Mar 11 11:33:13 EVENT FAILED: 1: start_server testmonApp 1

+testmon:node_up_local_complete[+148] RC=1
+testmon:node_up_local_complete[+149] : exit status of start_server testmonApp is:
1

```

Conclusions from Scenario 2

We drew the following major conclusions from this scenario:

- ▶ The startup monitor failing to find the application active within the Stabilization Interval results in two important situations:
 - The RG goes into an ERROR state (requiring manual intervention).
 - The Cleanup Method is run to stop any processes that might have started.
- ▶ The Cleanup Method should be coded so that it verifies that the cleanup is successful; otherwise, remnants of the failed start exist, possibly hindering a restart.
- ▶ If the Stabilization Interval is too short, the application start process is cut short. This stopped-while-starting situation might be confusing to the application start and stop scripts and to you during any debugging effort.

Suspending and resuming application monitoring

After configuring the application monitor, it is started automatically as part of the acquisition of the application server. If needed, it can be suspended while the application server is still online in PowerHA.

You can suspend application monitoring through the SMIT C-SPOC menus. Run **smitty cspoc**, select **Resource Group and Applications** → **Suspend/Resume Application Monitoring** → **Suspend Application Monitoring**, and then selecting the application server that is associated with the monitor that you want to suspend.

Use the same SMIT path to resume the application monitor. The output of resuming the application monitor that is associated with the application server APP1 is shown in Example 7-50.

Example 7-50 Output from resuming an application monitor

```

Dec 1 2022 16:24:48 cl_RMupdate: Completed request to resume monitors for application APP1.
Dec 1 2022 16:24:48 cl_RMupdate: The following monitors are in use for application APP1:test

```

7.7.10 Measuring application availability

You can use the Application Availability Analysis tool for measuring the amount of time that your highly available applications are available. The PowerHA software collects and logs the following information in a timestamped format:

- ▶ An application starts, stops, or fails.
- ▶ A node fails, shuts down, or comes online, and whether cluster services are started or shut down.
- ▶ An RG is taken offline or moved.
- ▶ Application monitoring is suspended or resumed.

According to the information that is collected by the application availability analysis tool, you can select a time for the measurement period, and the tool displays uptime and downtime statistics for a specific application during that period. Using SMIT, you can display this information:

- ▶ Percentage of uptime
- ▶ Amount of uptime
- ▶ Longest period of uptime
- ▶ Percentage of downtime
- ▶ Amount of downtime
- ▶ Longest period of downtime
- ▶ Percentage of time application monitoring was suspended

The Application Availability Analysis tool reports application availability from the PowerHA cluster perspective. It can analyze only those applications that were correctly configured in the cluster configuration.

This tool shows only the statistics that reflect the availability of the PowerHA application server, RG, and the application monitor (if configured). It cannot measure any internal failure in the application that can be detected by the user if it is not detected by the application monitor.

Using the Application Availability Analysis tool

You can use the Application Availability Analysis tool immediately after you define the application servers. The tool does not need any extra customization, and it automatically collects statistics for all application servers that are configured to PowerHA.

You can display the specific application statistics that are generated from the Application Availability Analysis tool by using SMIT menus. Run `smitty sysmirror` and select **System Management (C-SPOC) → Resource Group and Applications → Application Availability Analysis**.

Figure 7-41 on page 345 shows the SMIT panel that opens for the Application Availability Analysis tool in our test cluster. You can use the `smitty c1_app_AAA.dialog` fast path to get to the SMIT panel.

Application Availability Analysis	
Type or select values in the entry fields.	
Press Enter AFTER making all wanted changes.	
Fields	[Entry]
* Select an Application	[App1] +
* Begin analysis on YEAR (1970-2038)	[2012] #
* MONTH (01-12)	[03] #
* DAY (1-31)	[24] #
* Begin analysis at HOUR (00-23)	[16] #
* MINUTES (00-59)	[20] #
* SECONDS (00-59)	[00] #
* End analysis on YEAR (1970-2038)	[2012] #
* MONTH (01-12)	[03] #
* DAY (1-31)	[24] #
* End analysis at HOUR (00-23)	[17] #
* MINUTES (00-59)	[42] #
* SECONDS (00-59)	[00] #

Figure 7-41 Adding Application Availability Analysis SMIT panel

In the SMIT menu of the Application Availability Analysis tool, enter the selected application server, enter start and stop time for statistics, and run the tool. Example 7-51 shows the Application Availability Analysis tool output from our test cluster.

Example 7-51 Application Availability Analysis tool output

Analysis begins:	Saturday, 24-October-2022, 18:10
Analysis ends:	Saturday, 24-October-2022, 19:32
Application analyzed:	APP1
Total time:	0 days, 1 hour, 22 minutes, 0 seconds
Uptime:	
Amount:	0 days, 1 hour, 16 minutes, 51 seconds
Percentage:	93.72%
Longest period:	0 days, 1 hour, 10 minutes, 35 seconds
Downtime:	
Amount:	0 days, 0 hours, 5 minutes, 9 seconds
Percentage:	6.28%
Longest period:	0 days, 0 hours, 5 minutes, 9 seconds

Log records terminated before the specified ending time was reached.

Application monitoring was suspended for 75.87% of the time period analyzed.

Application monitoring state was manually changed during the time period analyzed.

Cluster services were manually restarted during the time period analyzed.



Cluster security

In this chapter, we describe the PowerHA security features and show how cluster security can be enhanced.

This chapter contains the following topics:

- ▶ Cluster security
- ▶ Using encrypted internode communication from CAA
- ▶ Secure remote command execution
- ▶ PowerHA and firewalls
- ▶ Federated security for cluster-wide security management

8.1 Cluster security

PowerHA SystemMirror 7 redesigned core clustering components. Core cluster functions, such as health monitoring and cluster communication, are done by Cluster Aware AIX (CAA), which is part of the AIX operating system.

Typically, data center clusters are deployed in trusted environments, so they might not need any security to protect cluster packets (which are custom to begin with and have no user-related data).

In Version 7, the repository disk provides CAA with inherent security. The repository disk is a shared disk across all the nodes of the CAA cluster, and it is used extensively and continuously by the CAA for health monitoring and configuration purposes. The expectation is that individual nodes have connectivity to the repository disk through the storage area network (SAN) fabric and pass all security controls of the SAN fabric regarding host access to the disk. Hosts can join the CAA cluster and become a member only if they have access to the shared repository disk. As a result, any other node trying to spoof and join the cluster cannot succeed unless it has an enabled physical connection to the repository disk.

The repository disk does not host any file system. This disk is accessed by clustering components in a raw format to maintain their internal data structures. These structures are internal to clustering software and not published anywhere.

Because of these reasons, most customers might choose to deploy clusters without enabling any encryption and decryption for the cluster. However, an administrator can choose to deploy CAA security; the various configuration modes that are supported are described in later sections.

CAA administration is root-based regarding security management.

8.1.1 The /etc/cluster/rhosts file

The /etc/cluster/rhosts file should contain a list, one entry per line, of either hostnames or the IP addresses of the hostnames of each node in the cluster.

Important: Be sure that the /etc/cluster/rhosts file has the following permissions:

- ▶ Owner: root
- ▶ Group: system
- ▶ Permissions: 0600

Initial cluster setup

During initial installation and configuration of the cluster, the /etc/cluster/rhosts file is empty. It must be populated. The node connection information is used by **c1cmd** and to create the CAA cluster. During the first synchronization of the PowerHA cluster, the CAA cluster is automatically created. After the CAA cluster is created, the entries in /etc/cluster/rhosts are no longer needed. However, do *not* remove the file. If you ever delete and re-create the cluster, or restore cluster configuration from a snapshot, you must populate /etc/cluster/rhosts again.

8.1.2 Additional cluster security features

The PowerHA Object Data Manager (ODM) files are stored in the /etc/es/objrepos directory. To improve security, their owner is root, and their group ID is hacmp. Most of their file permissions are 0640, with the following exceptions:

- ▶ The ODM file with 0600 is HACMPdisksubsys.
- ▶ The ODM files with 0664 are as follows:
 - HACMPadapter
 - HACMPcluster
 - HACMPnetwork
 - HACMPnode
 - HACMPsap_connector

All cluster utilities that are intended for public use have hacmp setgid turned on so that they can read the PowerHA ODM files. The hacmp group is created during PowerHA installation, if it is not already there.

8.1.3 Cluster communication over a virtual private network

You can set up PowerHA to use virtual private network (VPN) connections for internode communication. VPN support is provided by IP security features that are available at the AIX operating system level. VPN tunnels must be used to configure persistent IP addresses on each cluster node.

8.2 Using encrypted internode communication from CAA

PowerHA SystemMirror 7 Capacity on Demand (CoD) is implemented by CAA. The security setup can be done by using PowerHA `c1mgr` or CAA `c1ctrl` commands.

Message authentication and encryption rely on Cluster Security (CtSec) Services in AIX, and use the encryption keys that are available from Cluster Security Services. PowerHA SystemMirror message authentication uses Message Digest 5 (MD5) to create the digital signatures for the message digest. CAA encrypts the packets that are exchanged between the nodes by using a symmetric key (SymKey). This SymKey can be one of the types that are listed in Table 8-1.

Table 8-1 Symmetric key types

Symmetric key type	Key size
Advanced Encryption Standard (AES): MD5 with Advanced Encryption Standard	256 bits
DES: MD5 with Data Encryption Standard	64 bits
3DES: MD5 with Triple DES	192 bits

CAA exchanges the SymKey for certain configuration methods with host-specific certificate and private key pairs by using asymmetric encryption and decryption.

The PowerHA SystemMirror product does not include encryption libraries. Before you can use message authentication and encryption, the following AIX file sets must be installed on each cluster node. These file sets can be found on the AIX Expansion Pack.

- ▶ For data encryption with DES message authentication: rsct.crypt.des
- ▶ For data encryption standard Triple DES message authentication: rsct.crypt.3des
- ▶ For data encryption with AES message authentication: rsct.crypt.aes256

If you install the AIX encryption file sets after you have PowerHA SystemMirror running, restart the Cluster Communications daemon to enable PowerHA SystemMirror to use these file sets. To restart the Cluster Communications daemon, run the commands that are shown in Example 8-1.

Example 8-1 Restarting clcomd

```
# stopsrc -s clcomd
0513-044 The clcomd Subsystem was requested to stop.
# startsrc -s clcomd
0513-059 The clcomd Subsystem has been started. Subsystem PID is 37290470.
```

CAA provides the following methods of security setup regarding asymmetric or SymKeys:

- ▶ Self-signed certificate-private key pair

The administrator can choose this option for a simple setup. When the administrator uses this option, CAA generates a certificate and private key pair. The asymmetric key pair that is generated is type RSA (1024 bits). In this case, the administrator also provides a SymKey algorithm that is used (the key size is determined by the symmetric algorithm that is selected, as shown in Table 8-1 on page 349).

- ▶ User-provided certificate private key pair

With this option, administrators provide their own certificate and private key pair for each host. The administrator must store the pair in the same directory on each host in the cluster, and then invoke the security configuration interface. The certificate and private key pair must be type RSA and 1024 bits. The key pair must be in the Distinguished Encoding Rules (DER) format. The user also provides the SymKey algorithm that is used (the key size is determined by the symmetric algorithm that is selected, as shown in Table 8-1 on page 349).

- ▶ Fixed symmetric

For this option, administrators can choose not to set up a certificate and private key pair per node and instead provide a fixed SymKey of their own that is used for security. In this case, the administrator creates the key and stores the information in a directory (/etc/security/cluster/SymKey), provides that directory as input to the **clctr1** CAA command, and then chooses the symmetric algorithm to use.

CAA security also supports the following levels of security. At the time of writing, these levels are not differentiated at a fine granular level.

Medium or High All cluster packets are encrypted and decrypted.

Low or Disable CAA security is disabled.

Various CAA security keys are stored in the /etc/security/cluster/ directory. The default files are as follows (the location and file names are internal to CAA but should not be assumed):

- ▶ Certificate: /etc/security/cluster/cacert.der
- ▶ Private key: /etc/security/cluster/cakey.der
- ▶ Symmetric key: /etc/security/cluster/SymKey

8.2.1 Self-signed certificate configuration

As with most options, this configuration can be set up through the CLI by using the **clmgr** command or through System Management Interface Tool (SMIT). In this section, we demonstrate both approaches.

Enabling security

Example 8-2 shows how to enable security by using **clmgr**.

Example 8-2 Enabling a self-signed certificate by using clmgr

```
[jessica:root] / # clmgr manage cluster security LEVEL=High ALGORITHM=AES  
MECHANISM="SelfSigned"  
savesecconf: Security enabled successfully.  
savesecconf: Security enabled successfully.
```

```
Testing cluster communication by using the new security configuration...  
Cluster communication using the new security configuration appears to be  
functioning properly.
```

To enable security by using SMIT, complete the following steps:

1. Run **smitty cspoc** and select **Security and Users** → **PowerHA SystemMirror Cluster Security** → **Cluster Security Level**.

2. Press F4, and choose your preferred security level. Press Enter. In our example, we select High, as shown in Figure 8-1.

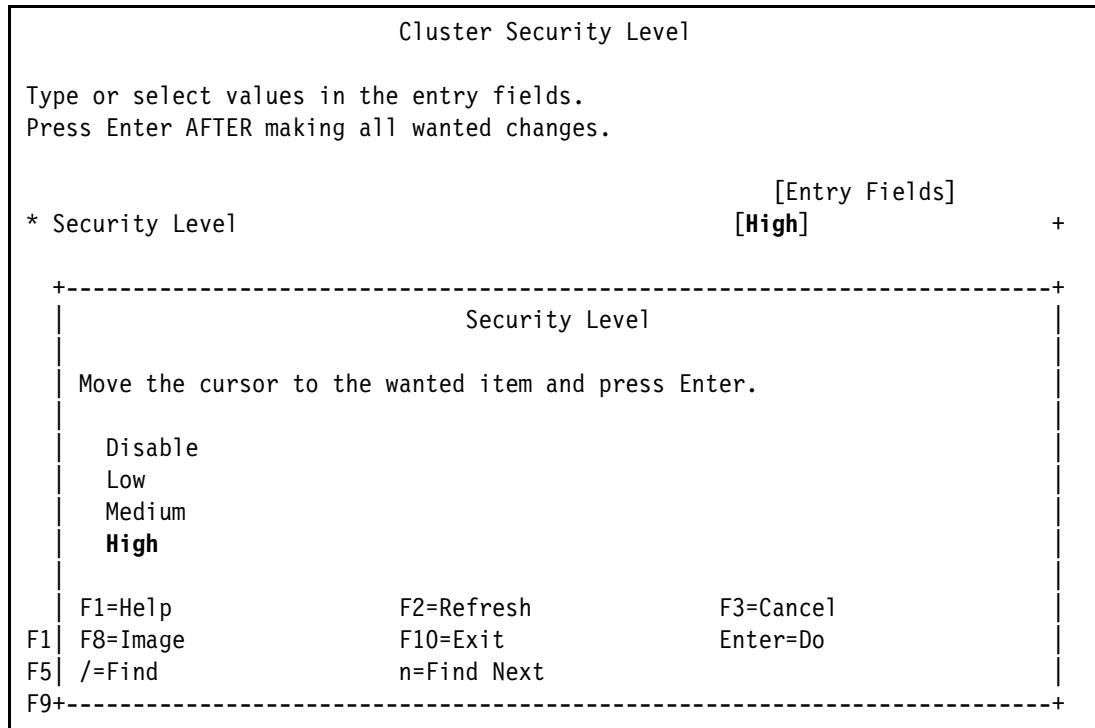


Figure 8-1 Setting the security level to High by using SMIT

3. Either back up one panel or use the fast path of `smitty clustsec`, select **Advanced Cluster Security Configuration** → **Setup Node Configuration Setup**, and then select an algorithm from the F4 list. We selected the options that are shown in Figure 8-2 on page 353.

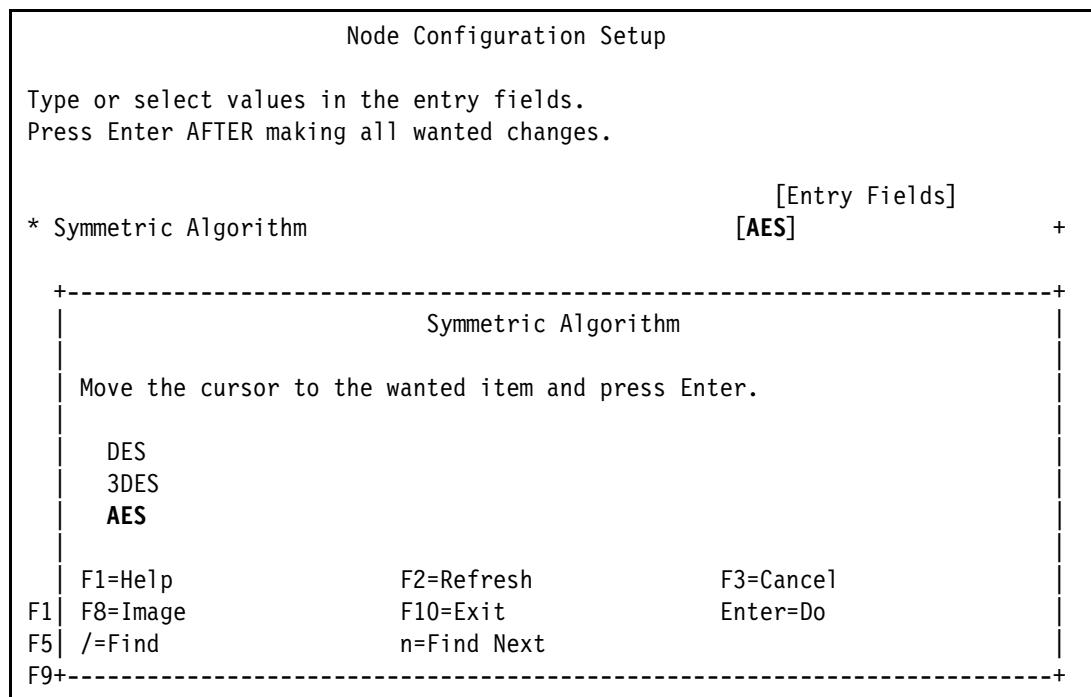


Figure 8-2 Setting the symmetric algorithm through SMIT

4. On successful completion, verify that the settings exist on the other node (jordan), as shown in Example 8-3.

Example 8-3 Verifying the security settings

```
[jordan:root] / # clmgr -a LEVEL query cluster
LEVEL="HIGH"
[jordan:root] / # clmgr -a MECHANISM query cluster
MECHANISM="self-sign"
[jordan:root] / # clmgr -a ALGORITHM query cluster
ALGORITHM="AES"
```

The mechanism automatically defaults to self-sign. You can change this setting in SMIT by running **smitty clustersec** and selecting **Advanced Cluster Security Configuration → Choose Security Mechanism**, as shown in Figure 8-3.

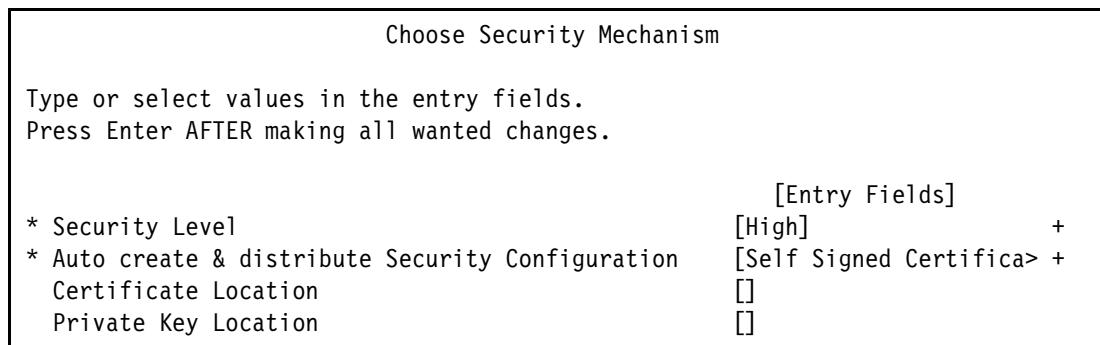


Figure 8-3 Setting the security mechanism through SMIT

Important: The settings are effective immediately and dynamically. Synchronizing the cluster is *not* required.

Disabling security

Disabling security can also be done by using the **clmgr** command or with SMIT. Example 8-4 shows disabling security by using the **clmgr** command.

Example 8-4 Disabling security by using clmgr

```
[jessica:root] / # clmgr manage cluster security LEVEL=Disable
Warning: disabling security for all cluster communication...
5... 4... 3... 2... 1...
```

Testing cluster communication by using the new security configuration...
Cluster communication using the new security configuration appears to be
functioning properly.

To disable security through SMIT, run **smitty clustersec**, select Cluster Security Level, and select Disable from the F4 list, as shown in Figure 8-4.

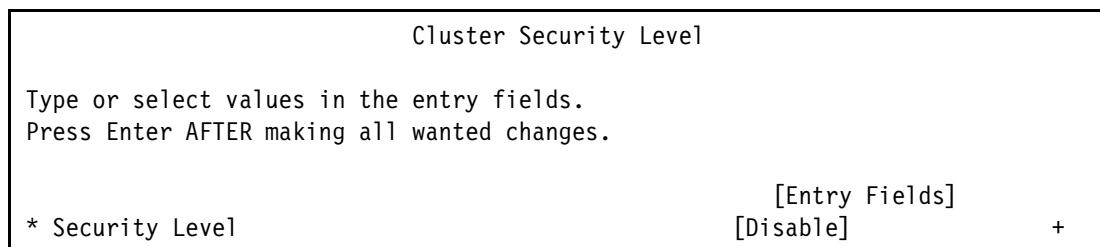


Figure 8-4 Disabling security by using SMIT

8.2.2 Custom certificate configuration

Administrators can provide their own certificate and key pair in DER format for CAA to enable security. In this section, we review the usage of OpenSSL tools to generate a custom certificate and private key pair that are provided as input to CAA.

Generating a certificate and private key pair by using OpenSSL

To generate the certificate and private key files, complete the following steps:

1. Create intermediate files and directories by running the following commands:

```
export CATOP=./demoCA
mkdir ${CATOP}
mkdir ${CATOP}/newcerts
echo "00" > ${CATOP}/serial
touch ${CATOP}/index.txt
```
2. Generate the key and certificate files in PEM format by running the following commands:

```
openssl genrsa -out cakey.pem -3 1024
openssl req -new -nodes -subj
'/countryName=US/stateOrProvinceName=DFW/organizationName=IBM/CN=jessica' -key
cakey.pem -out careq.pem
openssl ca -out cacert.pem -days 1095 -batch -keyfile cakey.pem -selfsign
-infiles careq.pem
```
3. Convert the certificate and private key files to DER format by running the following commands:

```
openssl x509 -in cacert.pem -inform PEM -out cacert.der -outform DER
openssl rsa -in cakey.pem -inform PEM -out cakey.der -outform DER
```

Enabling CAA for a custom certificate pair

Copy the generated files on all the nodes at the same location or generate the certificate and private key files on all the nodes at the same location, and then enable security with the certificate type as OpenSSL. You can enable CCA by using SMIT or the **clctrl** command.

To enable CAA by using SMIT, complete the following steps:

1. Run **smitty cspoc** and select **Security and Users → Advanced Cluster Security Configuration → Setup Node Configuration Setup**. Choose an algorithm, as shown in Figure 8-2 on page 353, and press Enter.
2. Either back up one menu, or use the fast path of **smitty clustsec_adv** and select **Choose Security Mechanism**. The menu that is shown in Figure 8-3 on page 354 opens. Complete the fields, making sure to choose OpenSSL Certificates. Press Enter.

To use the **clctrl** command, apply the following syntax:

```
clctrl -sec -s <SYM_KEY_ALG> -t <certificate_type> -c <certificate file path> -f
<private key file path>
```

The **-c** and **-f** flags are optional. Example 8-5 shows two samples of the command: The first one with the bare minimum requirements, and the other one with the exact file paths.

Example 8-5 Clctrl syntax example

```
[jessica:root] / # clctrl -sec -s 3DES -t OpenSSL
savesecconf: Security enabled successfully.

[jessica:root]clctrl -sec -s 3DES -t OpenSSL -c /etc/security/cluster/cacert.der
-f /etc/security/cluster/cakey.der
savesecconf: Security enabled successfully.
```

In either case, the changes are effective immediately and automatically updated across the cluster. There is no need for cluster synchronization.

8.2.3 Symmetric fixed key only configuration

Customers can choose to set up CAA security by using their own SymKey that is fixed for the entire cluster. Administrators generate the SymKey with a size for the algorithm that they plan to use. These keys are shown and can be determined by referring to Table 8-1 on page 349. Administrators must copy the key to all nodes by using the same file name and location, and then configure CAA security by using the **c1ctrl** CAA command.

If administrators want to replace the existing SymKey with a new one, they can update CAA to the new key and algorithm. When security is already enabled on the cluster, the user requests to enable the security with a different SymKey algorithm. This approach also applies the same security algorithm. The CAA security mechanism first disables the existing security and then enables the security with the requested SymKey algorithm or key.

Note: A simple method to generate a SymKey is to collect a random set of bytes and store it in a file. The example shows a key being generated for an AES 256 algorithm.

To generate a 256-bit key from the random device to a SymKey file, complete the following steps:

1. Run the **dd** command:

```
[jordan:root] / # dd if=/dev/random of=/tmp/SymKey bs=8 count=4  
4+0 records in.  
4+0 records out.
```

2. Copy the SymKey to the /etc/cluster/security directory to each node in the cluster. Then, enable security with the SymKey, as shown in Example 8-6.

Example 8-6 Enabling the security symmetric key

```
[jordan:root]c1ctrl -sec -x /etc/security/cluster/SymKey -s AES  
savesecconf: Security enabled successfully.
```

8.2.4 Symmetric key distribution by using an asymmetric key pair

CAA cluster components exchange the SymKey by using the asymmetric key pair as follows:

1. The administrator enables security after cluster creation by using one of these methods:
 - The **c1ctrl -sec** command
 - The **savesecconf** command
 - Through SMIT
2. The initiator node validates the security information that is provided by the user and then writes that security information to the repository disk. The following security policy information is written to the repository:
 - Security parameters of the SymKey algorithm
 - Security level
 - Certificate type
 - Certificate and private key files path
3. The initiator node generates the SymKey based on the specified algorithm. Then, it sends the public key request to all the member nodes in the cluster. Cluster node membership is defined by the information in the repository disk.
4. All target nodes receive the public key request and each node sends its public key (certificate data) to the initiator node.

5. When the initiator node receives certificate data from the nodes, it reads the public key from the certificate data, encrypts the SymKey with the individual node's public key, and sends the encrypted SymKey to the target nodes.
6. All target nodes receive the encrypted SymKey. Then, they decrypt it by using their own private key.
7. When the node receives the SymKey, it starts encrypting the packet data.
8. When the initiator node receives the public key from all the nodes, the initiator node starts encrypting the packet.

8.3 Secure remote command execution

Application start and stop scripts, customized cluster events, and other scripts might require the capability of running commands on remote nodes. Secure Shell (SSH) is a common method for securing remote command execution in today's networking environment.

As a best practice, use SSH. Dynamic logical partition (DLPAR) operations require SSH too. SSH and Secure Sockets Layer (SSL) together provide authentication, confidentiality, and data integrity. The SSH authentication scheme is based on public and private key infrastructure (SSL encrypts network traffic).

The following utilities can be used:

<code>ssh</code>	Secure remote shell, similar to <code>rsh</code> or <code>rlogin</code>
<code>scp</code>	Secure remote copy, similar to <code>rcp</code>
<code>sftp</code>	Encrypted file transfer utility, similar to <code>ftp</code>

8.4 PowerHA and firewalls

Consider the following factors when you place a PowerHA cluster behind a firewall:

- ▶ PowerHA does not require any open port on the firewall; no outgoing traffic originates from `c1cmd`, Reliable Scalable Cluster Technology (RSCT), or the Cluster Manager. You open only those ports that are required by your application, system management (for example, SSH), or both.
- ▶ Ensure that all service IP addresses can communicate with the outside network regardless of the interface to which they are bound. During a network failure, a service interface moves from one adapter to another one. If you are moving a resource group (RG), the service address moves to another node.
- ▶ Do not place a firewall between the nodes. In a PowerHA/XD cluster, your nodes might connect through a public network. In this case, use VPNs or another solution that is transparent to the cluster communications daemon.
- ▶ Be sure that the IP addresses that are listed in the `netmon.cf` file can be reached (`ping`) through your firewall.
- ▶ If you have `c1info` clients coming through a firewall, open the `c1info_client` port: 6174/tcp.
- ▶ Be sure that your firewall solution is redundant; otherwise, the firewall is a single point of failure (SPOF).

8.5 Federated security for cluster-wide security management

The AIX operating system provides a rich set of security capabilities. The goal of federated security is to enable the security administration of AIX security features across the cluster.

Federated security addresses Lightweight Directory Access Protocol (LDAP), role-based access control (RBAC), and Encrypted File System (EFS) integration into cluster management.

Through the federated security cluster, administrators can manage roles and the encryption of data across the cluster.

8.5.1 Federated security components

Federated security integrates components and features such as LDAP and RBAC into cluster management. The functional value of each component in the cluster management is shown in Figure 8-5.

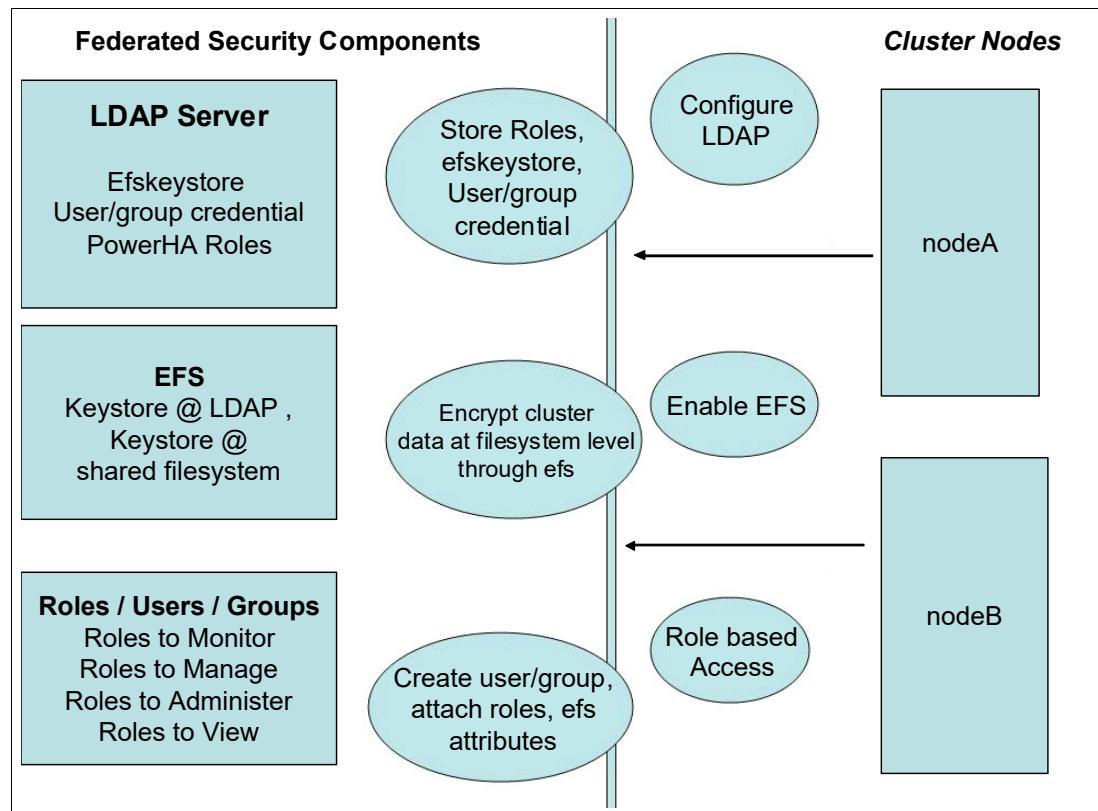


Figure 8-5 Federated security components

LDAP

The LDAP method is used by cluster nodes to allow centralized security authentication and access to user and group information.

The following information is stored by federated security at LDAP:

- ▶ PowerHA roles: Part of LDAP client configuration through PowerHA (for more information about PowerHA roles and LDAP client configuration, see “LDAP configuration” on page 362).
- ▶ EFS keystore: Exports EFS keystore data to LDAP. Stores the local user and group keystores to LDAP when EFS is enabled through PowerHA. For more information, see “Encrypted File System” on page 360.
- ▶ User and group account information for authorization.

The following supported LDAP servers can be configured for federated security:

- ▶ IBM Tivoli Director server
- ▶ Windows Active Directory server

All cluster nodes must be configured with the LDAP server and the client file sets. PowerHA provides options to configure the LDAP server and client across all cluster nodes.

SSL: An SSL connection is mandatory for binding LDAP clients to servers. Configure SSL in the cluster nodes.

The LDAP server and client configuration is provided through the PowerHA **smitty** option and the System Director PowerHA plug-in.

For the LDAP server and client setup, SSL must be configured. The SSL connection is mandatory for binding LDAP clients to servers.

LDAP server: The LDAP server must be configured on all cluster nodes. If an LDAP server exists, it can be incorporated into PowerHA for federated security usage.

For more information about the LDAP server and client configuration, see “Configuring LDAP” on page 360.

RBAC

Cluster administration is an important aspect of high availability (HA) operations, and security in the cluster is an inherent part of most system administration functions. Federated security integrates the AIX RBAC features to enhance the operational security.

During LDAP client configuration, four PowerHA defined roles are created in LDAP. These roles can be assigned to the user to provide restricted access to the cluster functions, based on the role.

ha_admin	Provides <i>administrator</i> authorization for the relevant cluster function. For example, taking a cluster snapshot is under administrator authorization.
ha_op	Provides <i>operator</i> authorization for the relevant cluster function. For example, “move cluster RG” is under operator authorization.

ha_mon	Provides <i>monitor</i> authorization for the relevant cluster function. For example, the command <code>c1RGinfo</code> is under monitor authorization.
ha_view	Provides <i>viewer</i> authorization. It has all read permissions for the cluster function.

Role creation: PowerHA roles are created when you configure the LDAP client in the cluster nodes.

Encrypted File System

The ECVG enables users on the system to encrypt their data in the Journaled File System 2 (JFS2) through their individual keystores. The keys are stored in a cryptographically protected keystore. On a successful login, the user keys are loaded in to the kernel and associated with the process credentials.

From the federated security perspective, the EFS keystores are stored in LDAP. There is an option to store the keystores through a shared file system in the cluster environment if LDAP is not configured in the cluster.

Tip: Store the EFS keystore in LDAP. As an option, if the LDAP environment is not configured, the keystore can be stored in a Network File System (NFS) mounted file system.

8.5.2 Federated security configuration requirement

The following prerequisites are necessary for a complete federated security environment:

- ▶ LDAP configuration:
 - Db2 9.7
 - GSKit file sets, preferably Version 8
 - LDAP Server file sets (Tivoli Director Server 6.3)
 - LDAP Client file sets (Tivoli Director Server 6.3)
- ▶ RBAC configuration
- ▶ EFS environment

The file sets for RBAC and EFS are available by default in AIX 6.1 and later, and no specific prerequisites are required. The challenge is to configure LDAP.

More information: For complete Db2 and LDAP configuration details, see [Configuring transparent LDAP for authentication and group lookup \(AIX\)](#).

Configuring LDAP

To install the LDAP configuration, complete the following steps:

1. Install and configure Db2.
2. Install the GSKit file sets.
3. Install the Tivoli Director Server (LDAP server and client) file sets.

Installing Db2

The Db2 installation steps are shown in Example 8-7 on page 361.

Example 8-7 Db2 installation steps

```
# ./db2_install

Default directory for installation of products - /opt/IBM/db2/V9.7
Do you want to choose a different directory to install [yes/no] ?
no
Specify one of the following keywords to install Db2 products.
ESE <<<< Select ESE >>>>
CLIENT
RTCL
Enter "help" to redisplay product names.
Enter "quit" to exit.
*****
ESE <<< selected option >>>>
Db2 installation is being initialized.
Total number of tasks to be performed: 46
Total estimated time for all tasks to be performed: 2369
Task #1 start
Description: Checking license agreement acceptance
Estimated time 1 second
Task #1 end
Task #47 end
The execution completed successfully.
```

GSKit file sets

Ensure that the GSKit file sets are installed in both server and client nodes, that is, in all cluster nodes, as shown in Example 8-8.

Example 8-8 GSKit file set installation

```
Installing GSKit (64-bit)
installpp -acgXd . GSKit8.gskcrypt64.ppc.rte
installpp -acgXd . GSKit8.gskssl64.ppc.rte
Installing GSKit (32-bit)
installpp -acgXd . GSKit8.gskcrypt32.ppc.rte
installpp -acgXd . GSKit8.gskssl32.ppc.rte
Install AIX Certificate and SSL base
installpp -acgXd . gksa.rte
installpp -acgXd . gskta.rte
```

Ensure that the SSL file sets are configured as shown in Example 8-9.

Example 8-9 SSL file sets

```
# lslpp -l | grep ssl
GSKit8.gskssl32.ppc.rte 8.0.14.7 COMMITTED IBM GSKit SSL Runtime With
GSKit8.gskssl64.ppc.rte 8.0.14.7 COMMITTED IBM GSKit SSL Runtime With
openssl.base 0.9.8.1100 COMMITTED Open Secure Socket Layer
openssl.license 0.9.8.801 COMMITTED Open Secure Socket License
openssl.man.en_US 0.9.8.1100 COMMITTED Open Secure Socket Layer
openssl.base 0.9.8.1100 COMMITTED Open Secure Socket Layer
```

Tivoli Directory Server (LDAP) file sets

The Tivoli Directory Server (LDAP) file sets are shown in Example 8-10.

Example 8-10 LDAP client and server file sets

```
# lslpp -l | grep idsldap
idsldap.clt32bit63.rte 6.3.0.3 COMMITTED Directory Server – 32 bit
idsldap.clt64bit63.rte 6.3.0.3 COMMITTED Directory Server – 64 bit
idsldap.clt_max_crypto32bit63.rte
idsldap.clt_max_crypto64bit63.rte
idsldap.cltbbase63.adt 6.3.0.3 COMMITTED Directory Server – Base Client
idsldap.cltbbase63.rte 6.3.0.3 COMMITTED Directory Server – Base Client
```

8.5.3 Federated security configuration details

After the required file sets are installed, federated security can be configured by using the following options:

- ▶ PowerHA smitty panel
- ▶ PowerHA SystemMirror PowerHA plug-in for Systems Director

LDAP configuration

To configure LDAP by using the SMIT panel, select **System Management (C-SPOC)** → **LDAP**, as shown in Figure 8-6.

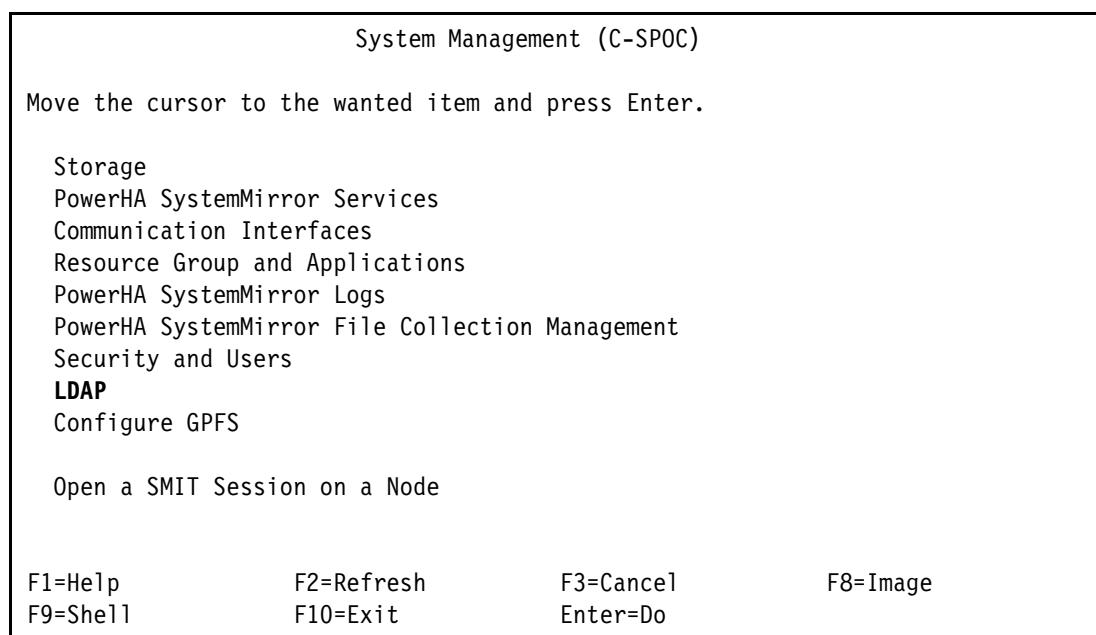


Figure 8-6 Configuring LDAP by using C-SPOC

Under the LDAP server configuration, two options are provided (Figure 8-7 on page 363):

- ▶ Configure a new LDAP server.
- ▶ Add an existing LDAP server.

If an LDAP server is already configured, the cluster nodes can use the existing LDAP server or configure a new LDAP server.

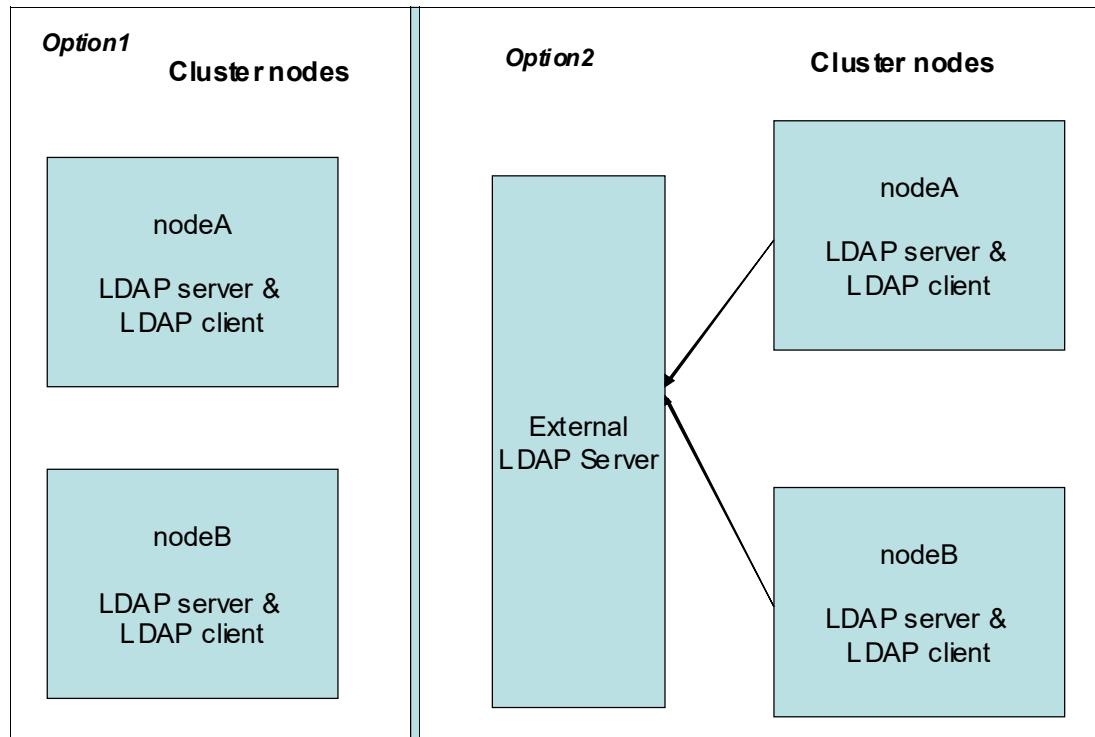


Figure 8-7 LDAP server configuration options

Configuring a new LDAP server

To configure a new LDAP server (Figure 8-8), enter `smitty sysmirror` and select **System Management (C-SPOC) → LDAP → LDAP Server → Configure a new LDAP server**.

Configure a new LDAP server																										
<p>Type or select values in the entry fields. Press Enter AFTER making all wanted changes.</p> <table border="0" style="width: 100%;"> <tr> <td style="vertical-align: top;"> <ul style="list-style-type: none"> * Hostnames * LDAP Administrator DN * LDAP Administrator password Schema type * Suffix / Base DN * Server port number * SSL Key path * SSL Key password * Version * Db2 instance password * Encryption seed for Key stash files </td> <td style="vertical-align: top; width: 10px;"></td> <td style="vertical-align: top;"> <p>[Entry Fields]</p> <table border="0" style="width: 100%;"> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[cn=admin]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">+</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">rfc2307aix</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[cn=aixdata,o=ibm]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[636]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> </table> </td> </tr> </table>				<ul style="list-style-type: none"> * Hostnames * LDAP Administrator DN * LDAP Administrator password Schema type * Suffix / Base DN * Server port number * SSL Key path * SSL Key password * Version * Db2 instance password * Encryption seed for Key stash files 		<p>[Entry Fields]</p> <table border="0" style="width: 100%;"> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[cn=admin]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">+</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">rfc2307aix</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[cn=aixdata,o=ibm]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[636]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> </table>	[cn=admin]	+	[]	#	rfc2307aix	/	[cn=aixdata,o=ibm]	#	[636]	/	[]	/	[]	#	[]	/	[]	#	[]	/
<ul style="list-style-type: none"> * Hostnames * LDAP Administrator DN * LDAP Administrator password Schema type * Suffix / Base DN * Server port number * SSL Key path * SSL Key password * Version * Db2 instance password * Encryption seed for Key stash files 		<p>[Entry Fields]</p> <table border="0" style="width: 100%;"> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[cn=admin]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">+</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">rfc2307aix</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[cn=aixdata,o=ibm]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[636]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">#</td> </tr> <tr> <td style="width: 30px; vertical-align: top; padding-right: 10px;">[]</td> <td style="width: 10px; text-align: right; vertical-align: bottom;">/</td> </tr> </table>	[cn=admin]	+	[]	#	rfc2307aix	/	[cn=aixdata,o=ibm]	#	[636]	/	[]	/	[]	#	[]	/	[]	#	[]	/				
[cn=admin]	+																									
[]	#																									
rfc2307aix	/																									
[cn=aixdata,o=ibm]	#																									
[636]	/																									
[]	/																									
[]	#																									
[]	/																									
[]	#																									
[]	/																									
F1=Help	F2=Refresh	F3=Cancel	F4=List																							
F5=Reset	F6=Command	F7>Edit	F8=Image																							
F9=Shell	F10=Exit	Enter=Do																								

Figure 8-8 New LDAP server configuration

Consider these key points:

- ▶ The existence of any LDAP instance is verified. The configuration continues only if the instance name is *not* ldapdb2. Have only one instance for federated security purposes.
- ▶ Internally, the configuration creates a peer-to-peer configuration to avoid LDAP instance failure. Therefore, a minimum of two nodes are expected as input.
- ▶ The configuration loads the local user and group information into LDAP.
- ▶ The configuration loads the RBAC AIX tables into LDAP.
- ▶ The configuration loads the EFS keystore that is defined for users and groups into LDAP.
- ▶ Various data trees are created in LDAP and are in the following file:
`/etc/security/ldap/sectoldif.cfg`

Encryption: The encryption seed must be a minimum of 12 characters.

The success of the LDAP configuration can be verified by using the ODM command that is shown in Example 8-11.

Example 8-11 ODM command to verify the LDAP configuration for federated security

```
# odmget -q "group=LDAPServer and name=ServerList" HACMPLDAP

HACMPLDAP:
    group = "LDAPServer"
    type = "IBMEexisting"
    name = "ServerList"
    value = "selma06,selma07"
```

Adding an existing LDAP server configuration

To add an existing LDAP server (Figure 8-9), enter **smitty sysmirror** and select **System Management (C-SPOC) → LDAP → LDAP Server → Add an existing LDAP server**.

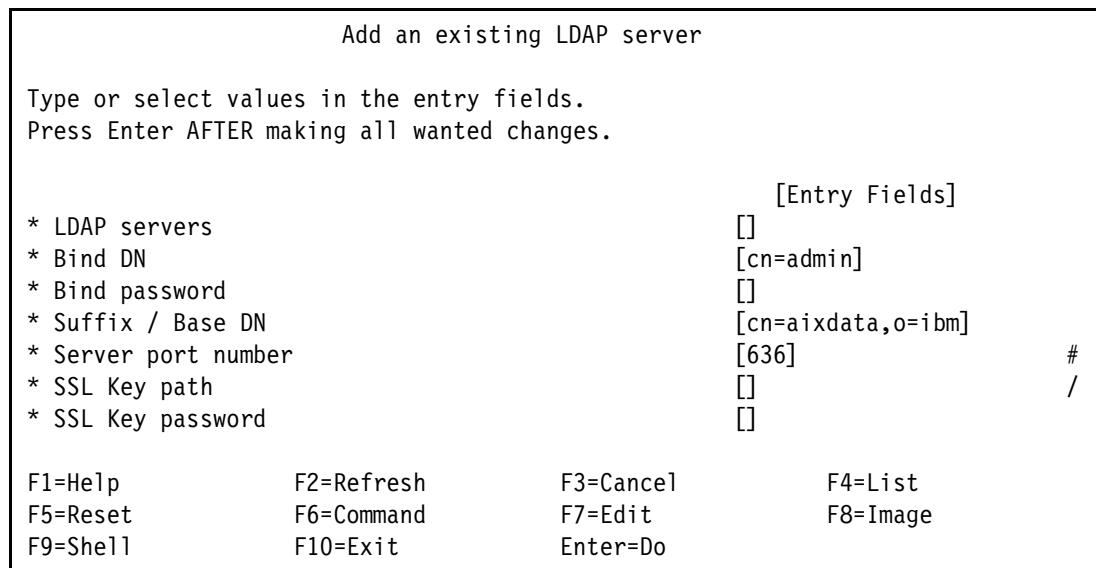


Figure 8-9 Adding an existing LDAP server

Consider these key points:

- ▶ Temporarily, the LDAP client is configured to verify the LDAP server input parameters. The compatible LDAP client file set must be at least at LDAP 6.2.
- ▶ User and group RBAC tables and EFS keystore are added to the existing LDAP server.

The success of adding an existing LDAP server is verified with the ODM command that is shown in Example 8-12.

Example 8-12 ODM command to verify the existing LDAP configuration for federated security

```
# odmget -q "group=LDAPServer and name=ServerList" HACMPLDAP

HACMPLDAP:
    group = "LDAPServer"
    type = "IBMEexisting"
    name = "ServerList"
    value = "selma06,selma07"
```

LDAP client configuration

To configure the LDAP client, enter **smitty sysmirror** and select **System Management (C-SPOC) → LDAP → LDAP Client → Configure LDAP client**, as shown in Figure 8-10.

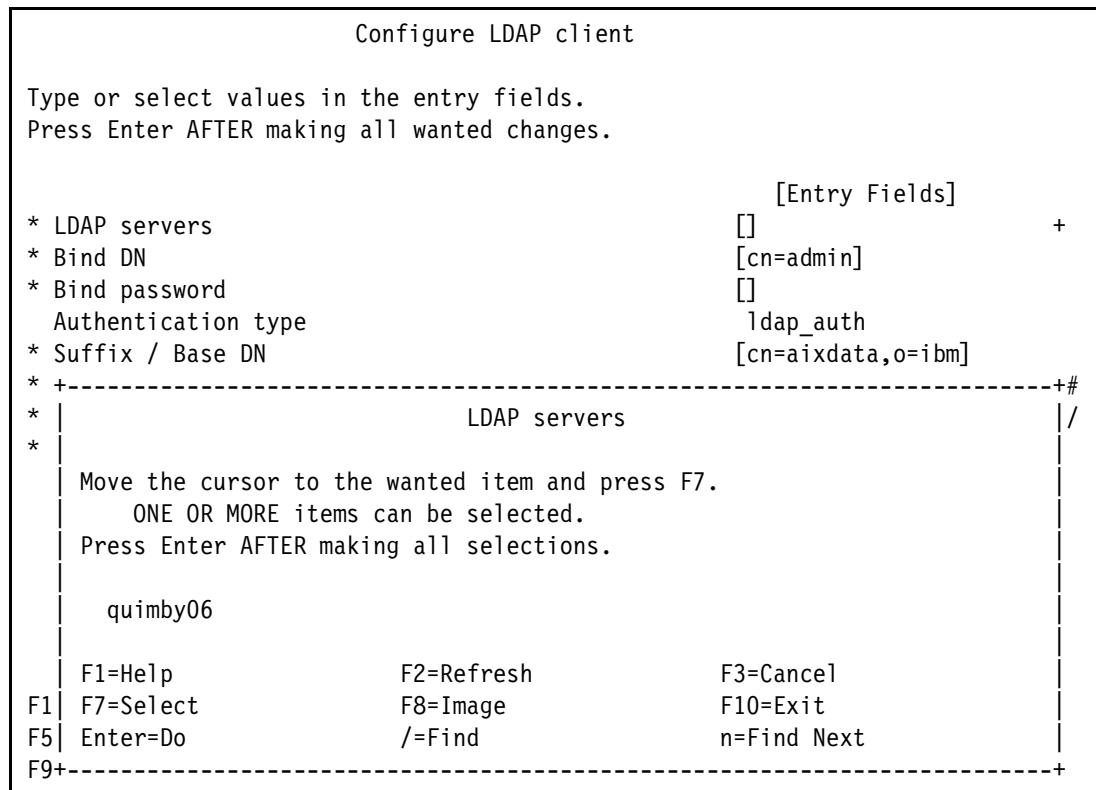


Figure 8-10 LDAP client configuration parameters

Consider these key points:

- ▶ Ensure that the LDAP client file sets and GSKit file sets are installed as described in 8.5.2, “Federated security configuration requirement” on page 360. The minimum compatible LDAP client file sets must be Version 6.2.
- ▶ The RBAC is configured during client configuration. The PowerHA defined roles are created in the LDAP server.
- ▶ LDAP client configuration generates SSL keys, extracts the server certificate, and binds with SSL.
- ▶ The home directory automatically at user login option, which is required by LDAP users, is enabled.

Verify the client configuration by using the ODM command that is shown in Example 8-13.

Example 8-13 ODM command to verify the LDAP client configuration

```
# odmget -q "group=LDAPClient and name=ServerList" HACMPLDAP

HACMPLDAP:
    group = "LDAPClient"
    type = "ITDSClient"
    name = "ServerList"
    value = "selma06,selma07"
```

You can also verify the client configuration by checking the LDAP client daemon status by using the command that is shown in Example 8-14.

Example 8-14 Verifying the client daemon status after the LDAP client configuration

```
# ps -eaf | grep secdapclntd
root 4194478      1  2 04:30:09      - 0:10 /usr/sbin/secdapclntd
```

RBAC configuration

During the LDAP client configuration, the PowerHA defined roles are created in the LDAP server.

Verify the configuration of the RBAC roles in the LDAP server by using the ODM command that is shown in Example 8-15.

Example 8-15 ODM command to verify the RBAC configuration into the LDAP server

```
# odmget -q "group=LDAPClient and name=RBACConfig" HACMPLDAP

HACMPLDAP:
    group = "LDAPClient"
    type = "RBAC"
    name = "RBACConfig"
    value = "YES"
```

Verify the four PowerHA defined roles that are created in LDAP, as shown in Example 8-16.

Example 8-16 Roles that are defined by PowerHA

```
# lsrole -a ALL | grep ha*
ha_admin
ha_op
ha_mon
ha_view
```

Example 8-16 shows that the RBAC is configured and can be used by the cluster users and groups. The usage scenario of roles by cluster users and groups are defined in “EFS configuration” on page 367.

EFS configuration

The EFS management scenario is shown in Figure 8-11.

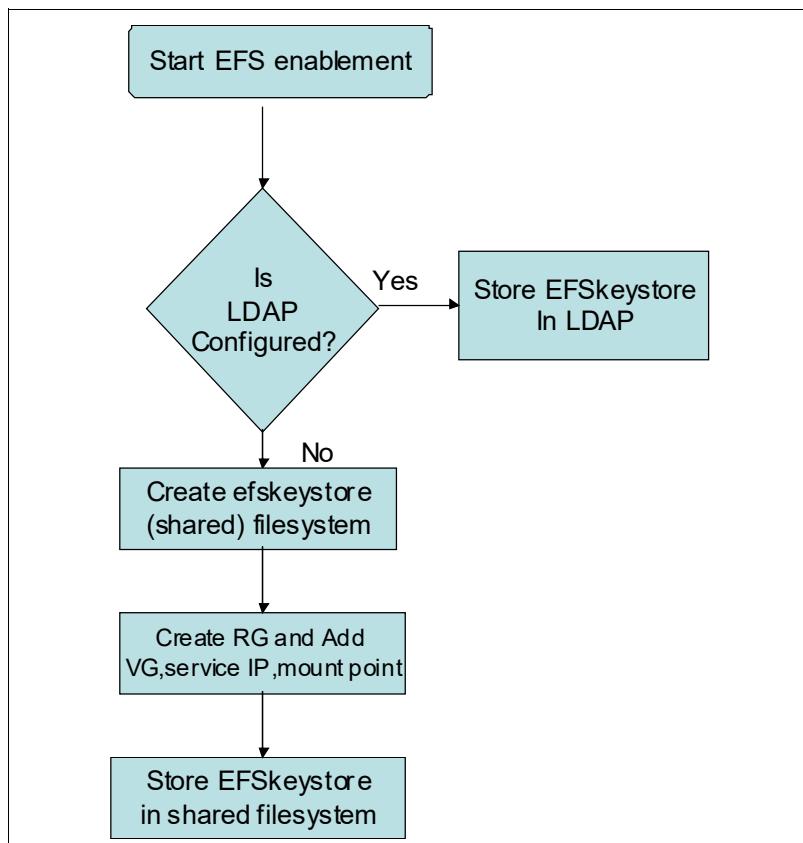


Figure 8-11 EFS management workflow

To configure the EFS management configuration (Figure 8-12), run `smitty sysmirror` and select **System Management (C-SPOC) → Security and Users → EFS Management**.

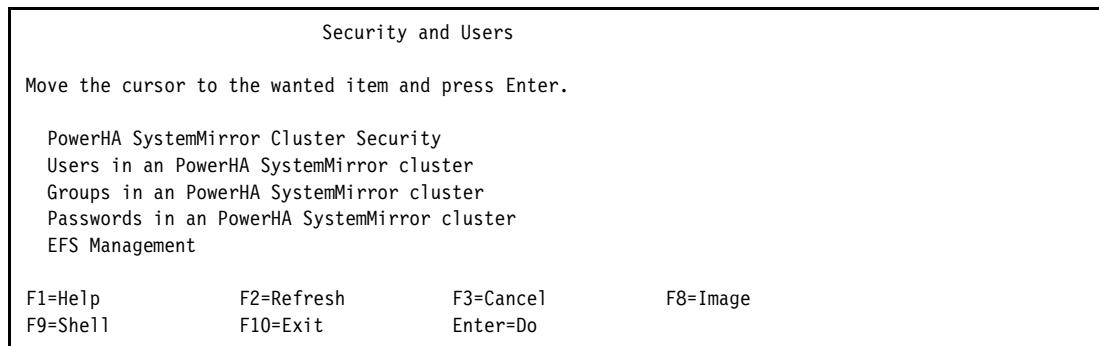


Figure 8-12 EFS management

Under EFS management, the options are provided to enable EFS and to store keystores either in LDAP or a shared file system.

Important: Federated security mandates that the LDAP configuration creates roles and stores EFS keystores. You can store EFS keystores under the shared file system only if LDAP is not configured.

EFS keystore in LDAP

If LDAP is configured, only the LDAP option is available to store the EFS keystore, as shown in Figure 8-13.

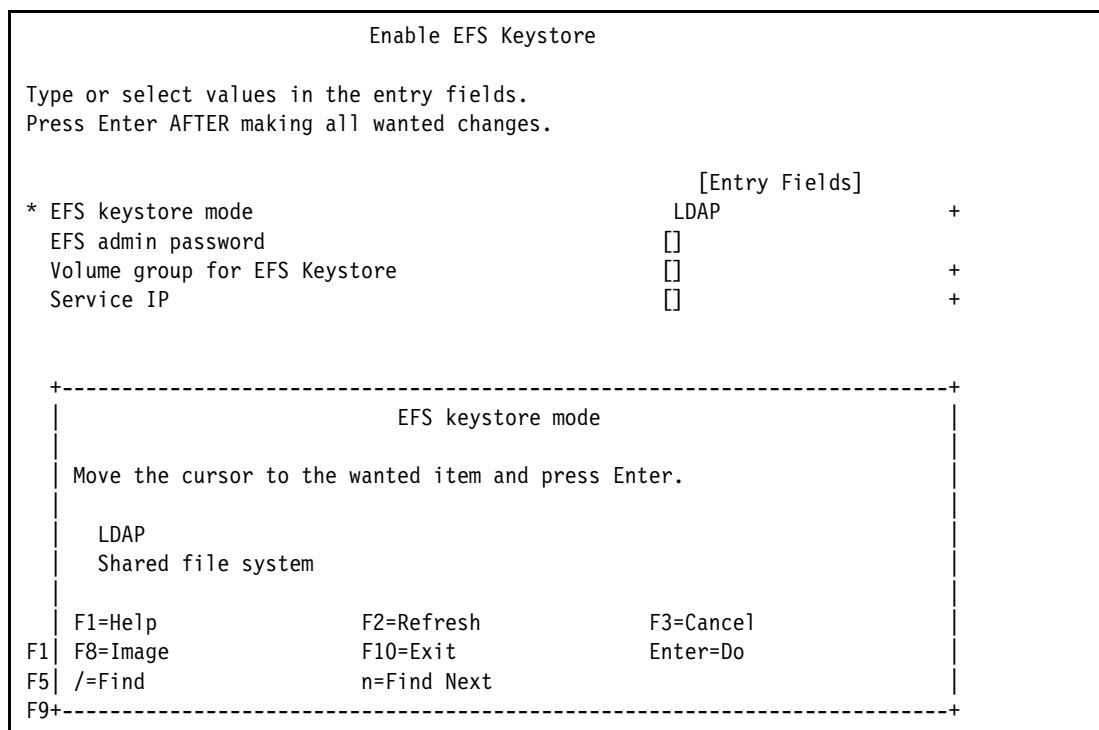


Figure 8-13 EFS keystore mode

Important: The volume group (VG) and service IP are invalid and ignored in LDAP mode.

Be aware of these key points:

- ▶ EFS is enabled by using the `/usr/sbin/efsenable -a -d cn=aixdata,o=ibm` command.
- ▶ You are prompted to enter the password to protect the initial keystore.

Verify the EFS enablement as understood by the cluster by using the command that is shown in Example 8-17.

Example 8-17 ODM command to verify the EFS enablement status

```
# odmget -q "group=EFSKeyStore AND name=mode" HACMPLDAP

HACMPLDAP:
    group = "EFSKeyStore"
    type = "EFS"
    name = "mode"
    value = "1"
```

EFS keystore in a shared file system

If LDAP is not configured but you want to use EFS to encrypt the cluster data, federated security provides an option to store the EFS keystore in a shared file system.

As shown in Figure 8-13 on page 368, to enable EFS and to store the EFS keystore in the shared file system, provide the VG and service IP details:

- ▶ The VG to store the EFS keystore in a file system.
- ▶ The service IP to mount the file system where the keystore is stored so that it is highly available to cluster nodes.

The configuration process includes these key steps:

1. Creates the EFS keystore file system in the specified VG.
2. Creates the EFS mount point on all cluster nodes.
3. Creates the RG to include the NFS exports with fallback as an NFS option.
4. Adds a specified VG in the RG.
5. Adds a service IP and a mount point in the RG.

Important: The file system creation, mount point, and NFS export are performed internally under the EFS keystore in a shared file system option.

Verify the configuration by using the ODM command that is shown in Example 8-18.

Example 8-18 ODM command to verify the EFS configuration in shared file system mode

```
# odmget -q "group=EFSKeyStore AND name=mode" HACMPLDAP

HACMPLDAP:
    group = "EFSKeyStore"
    type = "EFS"
    name = "mode"
    value = "2"
```

Advanced topics with examples

This part contains the following chapters:

- ▶ Chapter 9, “IBM PowerHA and IBM PowerVM” on page 373
- ▶ Chapter 10, “Extending resource group capabilities” on page 441
- ▶ Chapter 11, “Customizing resources and events” on page 479
- ▶ Chapter 12, “Network considerations” on page 497
- ▶ Chapter 13, “Cross-Site Logical Volume Manager stretched campus cluster” on page 525
- ▶ Chapter 14, “IBM PowerHA and IBM Power Virtual Server” on page 563
- ▶ Chapter 15, “Geographic Logical Volume Manager wizard” on page 575



IBM PowerHA and IBM PowerVM

In this chapter, we introduce the various virtualization features of PowerVM and the options that are available to a PowerHA cluster administrator. We describe the benefits of these features and how to configure them for use with PowerHA.

This chapter contains the following topics:

- ▶ Virtualization
- ▶ Virtual I/O Server
- ▶ Resource Optimized High Availability
- ▶ Live Partition Mobility

9.1 Virtualization

Virtualization is now common in the configuration of an IBM Power environment. Any environment, virtual or not, requires detailed planning and documentation so that administrators can effectively maintain and manage these environments.

When planning a virtual environment in which to run a PowerHA cluster, we must focus on improving hardware concurrency at the Virtual I/O Server (VIOS) level and also in the PowerHA cluster nodes. Typically, the VIOS hosts the physical hardware that is presented to the cluster nodes, so a critical question to address is, “What would happen to your cluster if any of those devices fail?”

The VIOS is considered a single point of failure (SPOF), so you should consider presenting shared disk and virtual Ethernet to your cluster nodes from extra VIOS partitions.

Figure 9-1 shows an example of considerations for PowerHA clusters in a virtualized environment.

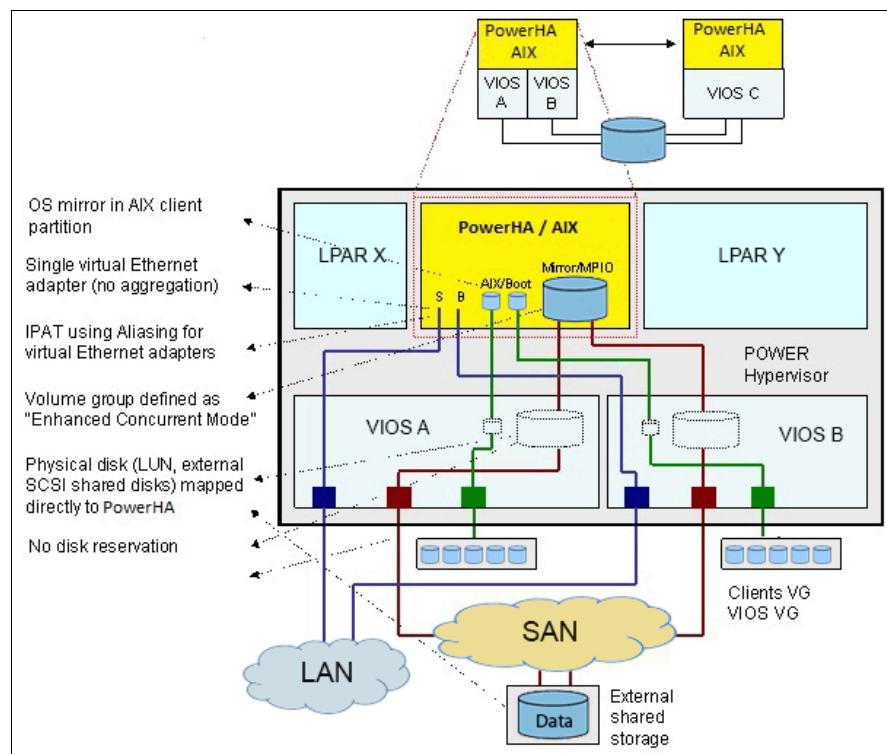


Figure 9-1 Example considerations for PowerHA with VIOS

For more information about configuring VIOSs, see *IBM PowerVM Virtualization Introduction and Configuration*, SG24-7940.

9.2 Virtual I/O Server

PowerHA supports using VIOS client partitions, and we can use virtual components such as virtual Ethernet adapters, virtual SCSI disks, and N_Port ID Virtualization (NPIV) with several important considerations, as in these examples:

- ▶ Management of active cluster shared storage is done at the cluster node level. The VIOS presents this storage only to the cluster nodes.
- ▶ Be sure that the *reservation policy* of all shared disks that are presented through the VIOS is set to no_reserve.
- ▶ All volume groups (VGs) that are created on VIOS clients and used for PowerHA clusters must be enhanced concurrent-capable, whether they are to be used in concurrent mode.
- ▶ Using Hardware Management Console (HMC) is required only if you want to use dynamic logical partition (DLPAR) with PowerHA.
- ▶ Integrated Virtualization Manager (IVM) contains a restriction on the number of virtual local area networks (VLANS) that a VIOS can have. The maximum number is four VLANs.

Several ways are available to configure AIX client partitions and resources for extra high availability (HA) with PowerHA. As a best practice, use at least two VIOSs for maintenance tasks at that level. An example of a PowerHA configuration that is based on VIOS clients is shown in Figure 9-2.

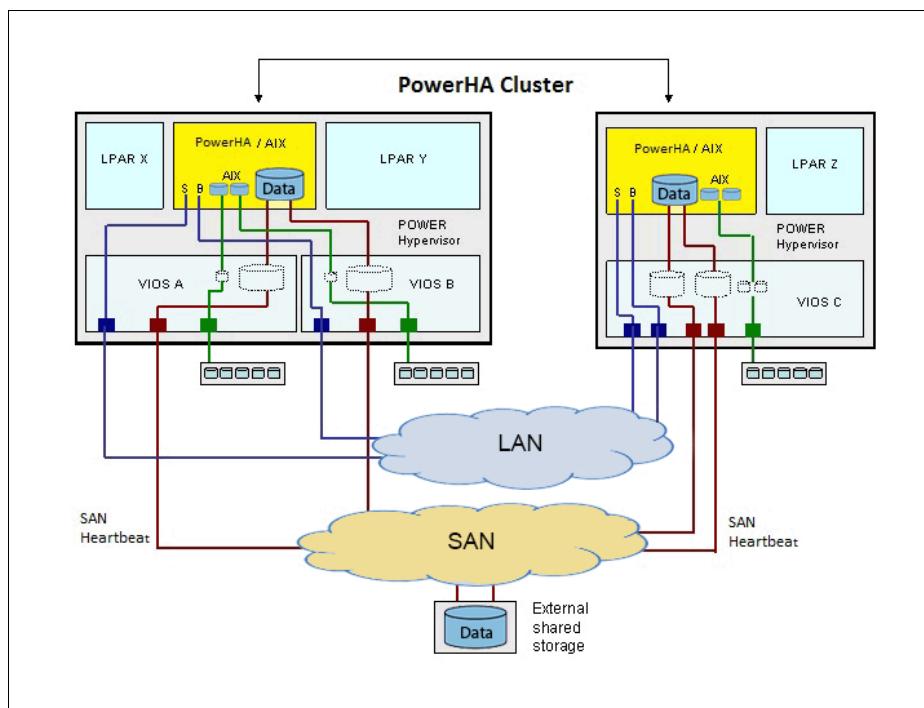


Figure 9-2 Example PowerHA configuration with VIOS

9.2.1 PowerHA and virtual storage

PowerHA requires enhanced concurrent volume groups (ECVGs) when you use virtual storage. This requirement applies to both vSCSI and NPIV. No hardware reserves are placed on the disks, and fast disk takeover is used if a VG must be taken over by another node.

If file systems are used on the standby nodes, they are not mounted until the point of failover because the VGs are in full active read/write mode only on the home node; the standby nodes have the VGs in passive mode, which does not allow access to the logical volumes (LVs) or file systems. If shared volumes (raw LVs) are accessed directly in enhanced concurrent mode (ECM), these volumes are accessible from multiple nodes, so access and locking must be controlled at a higher layer, such as databases.

All VG creation and maintenance is done by using the Cluster Single Point of Control (C-SPOC) function of PowerHA. The `bos.c1vm.enh` file set must be installed.

Example configuration

The following steps describe an example of how to set up concurrent disk access for a storage area network (SAN) disk that is assigned to two client partitions. Each client partition sees the disk through two VIOSs. On the disk, an ECVG is created. This configuration can be used to build a 2-node PowerHA test cluster on a single IBM Power machine.

1. Create the disk on the storage device.
2. Assign the disk to the VIOSs.
3. On the first VIOS, do the following tasks:

- a. Scan for the newly assigned disk:

```
$ cfgdev
```

- b. Change the SCSI reservation of that disk to `no_reserve` so that the SCSI reservation bit on that disk is not set if the disk is accessed:

```
$ chdev -dev hdiskN -attr reserve_policy=no_reserve
```

N is the number of the disk. Reservation commands are specific to the multipathing disk driver that is in use.

- c. Assign the disk to the first partition:

```
$ mkvdev -vdev hdiskN -vadapter vhostN [ -dev Name ]
```

N is the number of the disk; the vhost and the device name can be selected to what you want, but they can also be left out entirely. The system then creates a name automatically.

- d. Assign the disk to the second partition:

```
$ mkvdev -f -vdev hdiskN -vadapter vhostN [ -dev Name ]
```

4. On the second VIOS, do the following tasks:

- a. Scan for the disk:

```
$ cfgdev
```

- b. Change the SCSI reservation of that disk:

```
$ chdev -dev hdiskN -attr reserve_policy=no_reserve
```

- c. Assign the disk to the first cluster node:

```
$ mkvdev -vdev hdiskN -vadapter vhostN [ -dev Name ]
```

- d. Assign the disk to the second cluster node:


```
$ mkvdev -f -vdev hdiskN -vadapter vhostN [ -dev Name ]
```
- 5. On the first cluster node, do the following tasks:
 - a. Scan for the disk:


```
# cfgmgr
```
 - b. Create an enhanced concurrent-capable VG and a file system by using C-SPOC.

You now see the VGs and file systems on the second cluster node.

9.2.2 PowerHA and virtual Ethernet

Treat virtual Ethernet interfaces that are defined to PowerHA as single-adapter networks. Historically, this approach has meant that configuring the `netmon.cf` file to include a list of clients to `ping` is imperative. The file must be used to monitor and detect failures of the network interfaces. Because of the nature of the virtual Ethernet, other mechanisms to detect the failure of network interfaces are not effective. For more information about `netmon.cf`, see 12.5.1, “The `netmon.cf` format for virtual Ethernet environments” on page 517.

However, When you use virtual Ethernet with a Shared Ethernet Adapter (SEA) backed configuration, another option that is called `poll_uplink` is available on the virtual interface. For more information, see 12.6, “Using `poll_uplink`” on page 520.

For cluster nodes that use virtual Ethernet adapters, multiple configurations are possible for maintaining HA at the network layer. Consider the following factors:

- ▶ Configure dual VIOS to ensure HA of virtualized network paths.
- ▶ Use the servers that are already configured with virtual Ethernet settings because no special modification is required. For a VLAN-tagged network, the preferred solution is to use SEA failover; otherwise, consider using the network interface backup (NIB).
- ▶ One client-side virtual Ethernet interface simplifies the configuration; however, PowerHA might miss network events. For a more comprehensive cluster configuration, configure two virtual Ethernet interfaces on the cluster logical partition (LPAR) to enable PowerHA. Two network interfaces are required by PowerHA to track network changes, similar to physical network cards. Be sure to have two client-side virtual Ethernet adapters that use different SEAs to ensure that any changes in the physical network environment can be sent to the PowerHA cluster by using virtual Ethernet adapters, such as in a cluster with physical network adapters.

Explore the following configurations:

- ▶ Two Ethernet adapters in the PowerHA network with no SEA failover or NIB

In this configuration, each VIOS provides a virtual network adapter to the client on a separate VLAN. Without SEA failover or NIB, the redundancy is provided by PowerHA, such as in clusters with physical network adapters.
- ▶ NIB and a single Ethernet adapter in the PowerHA network

This configuration is similar to the previous configuration, but with NIB at the client side. This scenario cannot be used when VLAN tagging is required. PowerHA can miss network events in this setting because of one adapter on the cluster node.

- ▶ NIB and two Ethernet adapters per PowerHA network

This configuration is an improvement over the previous configuration. It can provide redundancy and load balancing across VIOSs. Also, PowerHA can track network events in this scenario.

- ▶ SEA failover and one virtual Ethernet adapter on the client side

A PowerHA configuration with SEA failover is helpful when VLAN tagging is used. Only one Ethernet adapter exists on the client side, and redundancy is provided by SEA failover. PowerHA cannot detect network events because of the single Ethernet adapter on each cluster node.

- ▶ SEA failover with two virtual Ethernet adapters in the cluster LPAR

A comprehensive setup that supports VLAN tagging and load sharing between VLANs. Two networks are defined and two virtual Ethernet adapters are configured for each network. Dual redundancy is provided with SEA failover and PowerHA. PowerHA can also track network events.

9.2.3 PowerHA and single root I/O virtualization and virtual network interface card

PowerHA 7.2.x supports both native single root I/O virtualization (SR-IOV) and virtual network interface cards (vNICs). To the cluster, there is no real discernible difference between the two types of interfaces. However, vNIC is required if you want to use Live Partition Mobility (LPM) or Live Update on a cluster node.

Unlike virtual Ethernet, `poll_uplink` does not apply to SR-IOV or vNIC devices primarily because the VIOS does not own the physical adapters or ports; they are assigned to the PowerVM hypervisor. For more information about configuring vNIC devices, see the [vNIC Functionality Guide](#).

Although a `netmon.cf` file is not required, it is a best practice to use one, as described in 12.5, “Understanding the `netmon.cf` file” on page 517.

9.2.4 PowerHA and SAN heartbeat

A SAN-based path is a redundant high-speed path of communication that is established between the hosts by using the SAN fabric that exists in any data center between hosts. Cluster Aware AIX (CAA) provides an extra heartbeat path over SAN or Fibre Channel (FC) adapters. It is not mandatory to set up an FC or SAN-based heartbeat path, but a configured SANComm (sfwcomm, as seen in the `lsccluster -i` output) provides an extra heartbeat path for redundancy.

Important: You can perform LPM on a PowerHA SystemMirror LPAR that is configured with SAN communication. However, when you use LPM, the SAN communication is not automatically migrated to the destination system. Configure the SAN communication on the destination system before you use LPM. For more information, see [Configuring SAN communication with LPM](#).

PowerHA SystemMirror 7.2.x supports a SAN-based heartbeat within a site. The SAN heartbeating infrastructure can be accomplished in many ways:

- ▶ Using real or physical adapters on cluster nodes and enabling the storage framework capability (sfwcomm device) of the HBAs. Currently, FC and serial-attached SCSI (SAS) technologies are supported. For more information about the HBAs and the steps to set up the storage framework communication, see [Setting up cluster SAN communication](#).
- ▶ In a virtual environment that uses NPIV or vSCSI with a VIOS. Enabling the sfwcomm interface requires activating *target mode enabled* (the **tme** attribute) on the real adapters in the VIOS and defining a private VLAN with VLAN ID 3358 for communication between the partitions that contain the sfwcomm interface and VIOS (the VLAN acts as a control channel, such as in a SEA failover). The real adapter on the VIOS must be a supported HBA.
- ▶ Using FC or SAN heartbeat requires the zoning of corresponding FC adapter ports (real FC adapters or virtual FC adapters on the VIOS).

Configure the following two types of zones:

- ▶ Heartbeat zones, which contain VIOS-physical worldwide port names (WWPNs):
 - The VIOS on each machine should be zoned together.
 - The virtual WWPNs of the client LPARs should not be zoned together.
- ▶ Storage zones:
 - Contains the LPAR's virtual WWPNs.
 - Contains the storage controller's WWPNs.

To perform zoning, complete the following steps:

1. Log in to each VIOS (both VIOSs on each managed system). Verify that the FC adapters are available. Capture the WWPN information for zoning.
2. From the client LPAR, capture the WWPNs for the fcsX adapter.
3. Perform the zoning on switch fabrics as follows:
 - a. Zone the LPAR's virtual WWPN to the storage ports on the storage controller that is used for shared storage access.
 - b. Create the zones that contain VIOS physical ports, which will be used for heartbeat.

After the zoning is complete, enable the **tme** attribute. The **tme** attribute for a supported adapter is available only when the minimum AIX level for CAA is installed (AIX 6.1 Technology Level (TL) 6 or later or AIX 7.1 TL0 or later). This action must be performed on all VIOSs.

Complete the following steps:

1. Configure the FC adapters for SAN heartbeating on VIOS:

```
# chdev -l fcsX -a tme=yes -P
```
2. Repeat step 1 for all FC adapters.
3. Set dynamic tracking to yes and FC error recovery to fast_fail:

```
# chdev -l fcscsiX -a dyntrk=yes -a fc_err_recov=fast_fail -P
```
4. Restart the VIOS.
5. Repeat steps 1 - 4 for every VIOS that serves the cluster LPARs.
6. On the HMC, create a virtual Ethernet adapter for each cluster LPAR and VIOS. Set the VLAN ID to 3358. Do not put other VLAN IDs or any other traffic on this interface. Save the LPAR profile.

7. On the VIOS, run the **cfgmgr** command and check for the virtual Ethernet and sfwcomm device by using the **lsvdev** command:

```
#lsvdev -C | grep sfwcomm
```

Here is the output of the **lsvdev** command:

```
sfwcomm0 Available 01-00-02-FF Fibre Channel Storage Framework Communication  
sfwcomm1 Available 01-01-02-FF Fibre Channel Storage Framework Communication
```

8. On the cluster nodes, run the **cfgmgr** command, and check for the virtual Ethernet adapter and sfwcomm with the **lsvdev** command.
9. No other configuration is required at the PowerHA level. When the cluster is configured and running, you can check the status of SAN heartbeat by using the **lscuster -i sfwcin** command.

Demonstration: For a demonstration of creating SAN heartbeating in a virtualized environment, see this [YouTube video](#).

9.3 Resource Optimized High Availability

This section describes the following topics regarding Resource Optimized High Availability (ROHA):

- ▶ Concepts and terminology
- ▶ Planning
- ▶ Configuring ROHA
- ▶ Application Provisioning with DLPAR functions
- ▶ Sample configuration and test scenarios
- ▶ Managing, monitoring, and troubleshooting ROHA

We expect that proper LPAR and DLPAR planning is part of your overall process before implementing any similar configuration. Understanding the following topics is important:

- ▶ The requirements and how to implement them.
- ▶ The overall effect that each decision has on the overall implementation.

9.3.1 Concepts and terminology

The ROHA feature of PowerHA can be configured to manage CPU and memory resource allocations by using dynamic logical partitioning and Capacity on Demand (CoD). CoD resources are composed of Enterprise Pool CoD (EPCoD) and On/Off Capacity on Demand (On/Off CoD) resources.

Dynamic logical partitioning

Dynamic logical partitioning is a facility that provides the capability to logically attach or detach a managed system's resources to and from an LPAR's operating system without restarting the system.

Enterprise Pool Capacity on Demand resources

EPCoD resources are resources that can be moved between systems within the same pool. Physical resources such as CPU and memory are not moved between systems; what is moved is the privilege or right to use or access the resources. This privilege is granted to any server in the pool, thus allowing for flexibility in managing the pool of resources and in allocating the resources where they are most needed.

On/Off Capacity on Demand resources

The On/Off CoD resources are physical resources (processor and memory) that are preinstalled and inactive (unpaid resources). These resources can be temporarily activated through an On/Off CoD license.

Trial Capacity on Demand

Trial Capacity on Demand (Trial CoD) resources are temporary resources that are available for a limited period. When the Trial CoD license is entered into the HMC, the resources are available immediately for use.

By integrating with DLPAR and CoD resources, PowerHA SystemMirror ensures that each node can support the application with reasonable performance at a minimum cost. This approach allows for flexibility in tuning the resource capacity of the LPAR by using the On/Off CoD function to upgrade the capacity when your application requires more resources without having to pay for idle capacity until you need it. Using EPCoD allows for sharing resources across systems in the same enterprise pool whenever they are needed.

Table 9-1 displays all the available types of the CoD offering. Only two of them are dynamically managed and controlled by PowerHA SystemMirror: EPCoD and On/Off CoD.

Table 9-1 CoD offerings and PowerHA

CoD offering	PowerHA SystemMirror 6.1 Standard and Enterprise Edition	PowerHA SystemMirror 7.1 or 7.2 Standard and Enterprise Edition
Enterprise Pool Memory and Processor	No.	Yes, from Version 7.2.
On/Off CoD (temporary) Memory	No.	Yes, from Version 7.1.3 SP2.
On/Off CoD (temporary) Processor	Yes.	Yes.
Utility CoD (temporary) Memory and Processor	Utility CoD is automatically performed at the PHYP/System level. PowerHA cannot play a role in the same system.	
Trial CoD Memory and Processor	Trial CoD is used if available through a DLPAR operation.	
Capacity Upgrade on Demand (CUoD) (permanent) Memory & Processor	CUoD is used if available through a DLPAR operation. PowerHA does not handle this resource directly.	

9.3.2 Planning

This section contains the following topics:

- ▶ Planning for ROHA
- ▶ Prerequisites for using ROHA

Planning for ROHA

Using the ROHA feature entails proper planning of the amount of resources that will be allocated through the HMC. Be familiar with the different types of CoD licenses that are available.

To prepare for a ROHA configuration, the following cluster information must be reviewed:

- ▶ LPAR resources information and resource group (RG) policies information:
 - The amount of memory and CPU resources that the applications that are supported by your cluster require when running on their regular hosting nodes. Under normal running conditions, check how much memory and what number of CPUs each application uses to run with optimum performance on the LPAR node on which its RG is (normally, the home node for the RG).
 - Determine the startup, failover, and fallback policies of the RG that contains the application controller by using the **c1RGinfo** command. This command identifies the LPAR node to which the RG will fall over in a failure.
 - The amount of memory and CPU that is allocated to the LPAR node on which the RG will fall over in a failure. This LPAR node is referred to as a standby node. With these numbers in mind, consider whether the application's performance on the standby node will be impaired, if the application is running with fewer resources.
 - Check on the existing values for the LPAR minimums, maximums, and wanted amounts (resources and memory) specified by using the **1shwres** or **pvmtl** command on the standby node.
- ▶ Estimate the resources that are required for the application:
 - For each standby node that can host an RG, you must estimate the optimal amount of resources (CPU and memory) that this node requires for the application to run successfully. The optimal amount of resources that you identify are specified in PowerHA SystemMirror. PowerHA SystemMirror verifies that the optimal amount is contained within the boundaries of the LPAR.
 - When you specify that an application use resources through the DLPAR operation, PowerHA SystemMirror dynamically activates CoD resources from either EPCoD or On/Off CoD resource pools. When the application no longer requires these extra resources, they are returned to the corresponding free pool.
- ▶ Revise existing pre-event and post-event scripts that were used to allocate DLPAR resources.

- ▶ The following limitations are applicable for ROHA multi-cluster deployments. A multi-cluster deployment is an environment with multiple clusters that are deployed across two different systems. In a multi-cluster deployment environment, each cluster has one node on an active system, and another node on the standby system.
 - You can have up to 10 clusters in a ROHA multi-cluster deployment. For example, you can have 10 LPARs hosting nodes on the active system and 10 LPARs hosting nodes on the standby system. In this example, if a failure occurs on the active system, the 10 LPARs on the standby system independently contact the PowerVM NovaLink or HMC simultaneously to obtain resources.
 - EPCoD is supported, but you must verify that your system has enough available resources to meet resource requirements for all clusters and possible resource conflicts. Plan to allocate an extra 15% of the total resource requirement for all clusters on the system.
 - On/Off CoD is not supported in a multi-cluster deployment.
- ▶ ROHA does *not* support RGs that have a startup policy of Online on All Available Nodes.
- ▶ In PowerHA SystemMirror 7.2.1 SP1 or later, ROHA supports dynamic automatic reconfiguration (DARE). With DARE, you can change CoD resource requirements for application controllers without bringing down a workload. The resource changes that you specify are synchronized across the cluster.
- ▶ In PowerHA SystemMirror 7.2.1 SP1 or later, ROHA supports LPM to migrate partitions from one system to another one. Before you implement LPM, you must verify the following information about the target system, target HMC, and target PowerVM NovaLink:
 - Before you start the LPM process, you must verify and synchronize all nodes in the target system and the source system.
 - Both the source and target systems must be part of the same EPCoD.
 - If the target system is running HMC 8.40 or earlier, you must establish connections for the primary HMC and the backup HMC. If the target system is running HMC 8.50 or later, only a connection to the primary HMC is required.
 - The source and target systems must have a similar number of resources. If the target system has fewer resources than the source system, a failover might occur. Before the LPM process starts, you must verify that the target system has enough available resources.
 - The target HMC and the PowerVM NovaLink must be defined in the PowerHA SystemMirror cluster.
- ▶ In PowerHA SystemMirror 7.2.1 SP1 or later, ROHA supports resources that can be released asynchronously if the source node and the target node are on different systems.

Prerequisites for using ROHA

Here are the requirements to implement ROHA:

- ▶ PowerHA SystemMirror 7.2 Standard Edition or Enterprise Edition
- ▶ AIX level:
 - 7.1 TL3 SP5 or later
 - 7.1 TL4 or later
 - 7.2 or later
- ▶ OpenSSH

- ▶ HMC requirements:
 - To use the EPCoD license, your system must be using HMC 8.8.7 firmware or later.
 - Configure the backup HMC for EPCoD with HA.
 - For the EPCoD User Interface (UI) in HMC, the HMC must have a minimum of 2 GB of memory.

- ▶ Hardware requirements for using an EPCoD license:

- IBM Power7+ processor-based systems: 9117-MMD (IBM Power 770 D model) or 9179-MHD (IBM Power 780 D model) that uses FW780.10 or later.
- IBM Power8 processor-based system: 9119-MME (IBM Power E870) or 9119-MHE (IBM Power E880) that uses FW820 or later.
- IBM Power9 processor-based system: 9080-M9S (IBM Power E980).
- IBM Power10 processor-based system: 9080-HEX (IBM Power E1080).

For more information about Enterprise Pools, see *IBM Power Systems Private Cloud with Shared Utility Capacity: Featuring Power Enterprise Pools 2.0*, SG24-8478.

- ▶ Verify the LPAR node name.

Historically, the HMC LPAR name and AIX hostname had to match. This situation is no longer with case starting with the following levels:

- IBM AIX 7.1 with TL 4 or later, or AIX 7.2 or later.
- Power Firmware SC840 for Enterprise Power8 (Power E870 and Power E880).

- ▶ Verify what DLPAR resources are available and what CoD licenses that you can access.

PowerHA SystemMirror does not identify what resources are available. PowerHA SystemMirror has no control over whether the resources are physically available on IBM Power servers or whether they are unallocated and available. PowerHA SystemMirror provides dynamic allocations only for CPU and memory resources. PowerHA SystemMirror does not support dynamic changes of the I/O slots.

- ▶ Identify the type of CoD function.

Create the EPCoD on the HMC. Enter the license key (also called activation code) for either the EPCoD or On/Off CoD on the HMC.

- ▶ Establish secure connections to the HMC.

PowerHA SystemMirror must communicate securely with the LPAR nodes through HMC. Install Secure Shell (SSH) for PowerHA SystemMirror to access the HMC without entering a username and password. If you want to use SSH for a secure connection, from the HMC, select **Users and Security → Systems and Console Security → Enable Remote Command Execution**. In the Remote Command Execution window, select **Enable remote command execution using the ssh facility**. The AIX operating system must have SSH installed to generate the public and private keys.

Note: PowerHA SystemMirror uses the root user on the cluster node to issue the SSH commands to the HMC. On the HMC system, the commands are run as the hscroot user.

- ▶ Verify HMC access.

Verify that all nodes and LPARs can access the HMC that manages all nodes and LPARs. If the node or LPAR cannot access the HMC, the resource allocation fails.

Note: PowerHA SystemMirror ROHA does not release more resources than the resources that are received from the resource release operation while using EPCoD or On/Off CoD.

9.3.3 Configuring ROHA

The following tasks are used to configure ROHA to a PowerHA cluster:

- ▶ Ensuring correct name resolution
- ▶ Installing SSH on PowerHA nodes
- ▶ Configuring HMC SSH access

Ensuring correct name resolution

One common issue is that name resolution is inconsistent across all systems. If name resolution is not configured correctly, the DLPAR feature cannot be used. The underlying Reliable Scalable Cluster Technology (RSCT) infrastructure expects an identical hostname resolution on all participating nodes. If not, RSCT cannot communicate properly.

Ensure that all nodes and the HMC are configured identically by checking the following list. All systems (all PowerHA nodes and the HMC) must do these tasks.

- ▶ Resolve the participating hostnames and IP addresses, which include reverse name resolution.
- ▶ Use the same type of name resolution, either short or long name resolution. All systems should use the same name resolution order, either local or remote.

To ensure these requirements, check the following files:

- /etc/hosts on all systems
- /etc/netsvc.conf on all AIX nodes
- /etc/host.conf, if applicable, on the HMC

We expect that knowing how to check these files on the AIX systems is common knowledge. However, what is not as well known is how to check the files on the HMC, which is covered in the following sections.

Installing and configuring SSH on PowerHA nodes

To use remote command operations on the HMC, SSH must be installed on the PowerHA nodes. The HMC must be configured to allow access from these partitions. In this section, we cover installing OpenSSH on the AIX cluster nodes.

With different versions of SSH and HMC code, these steps might differ slightly. We documented the processes that we used to successfully implement our environment.

Installing SSH on PowerHA nodes

Before completing the following steps, be sure that you downloaded or copied these packages onto the PowerHA nodes. We chose to get openssh and openssl from the AIX base media and put them in a common installation directory (/usr/sys/inst.images).

You can now install by using `smitty install_all`. The core file sets that you must install and the results of our installation are shown in Example 9-1.

Example 9-1 Installing SSH

```
smitty install_all
  OpenSSH.base                                     ALL
    @ 6.0.0.6103  Open Secure Shell Commands
    @ 6.0.0.6103  Open Secure Shell Server

  openssh.license                                    ALL
    @ 6.0.0.6103  Open Secure Shell License
  openssh.man.en_US                                 ALL
    @ 6.0.0.6103  Open Secure Shell Documentation - US English
  openssh.msg.EN_US                                ALL
    @ 6.0.0.6103  Open Secure Shell Messages - US English (UTF)
  openssl.base                                      ALL
    @ 1.0.1.500   Open Secure Sockets Layer

  openssl.license                                    ALL
    @ 1.0.1.500   Open Secure Socket License
+-----+
[jessica:root] / # lsLpp -L |grep open
  openssh.base.client    6.0.0.6103  C   F  Open Secure Shell Commands
  openssh.base.server    6.0.0.6103  C   F  Open Secure Shell Server
  openssh.license        6.0.0.6103  C   F  Open Secure Shell License
  openssh.msg.en_US      6.0.0.6103  C   F  Open Secure Shell Messages -
  openssl.base           1.0.1.500   C   F  Open Secure Sockets Layer
  openssl.license        1.0.1.500   C   F  Open Secure Socket License
  openssl.man.en_US      1.0.1.500   C   F  Open Secure Sockets Layer
```

Tip: Be sure to select yes in the field to accept the license agreement.

Now that SSH is installed, configure the PowerHA nodes to access the HMC without passwords for remote DLPAR operations.

Configuring HMC SSH access

Here are the high-level steps that we used in our setup to enable SSH access from our PowerHA nodes on HMC V10 R1 M1010:

1. Enable HMC SSH access.
2. Generate SSH keys on the PowerHA nodes.
3. Enable non-password HMC access through the `authorized_keys2` file.

Be sure that the HMC is set up to allow remote operations:

1. In the left navigation area, select **Users and Security**.
2. In the dialog box, select **Systems and Console Security**.
3. In the right frame under Remote Control, select **Enable Remote Command Execution**.
4. Select the **Enable remote command execution using the ssh facility** checkbox (Figure 9-3 on page 387).

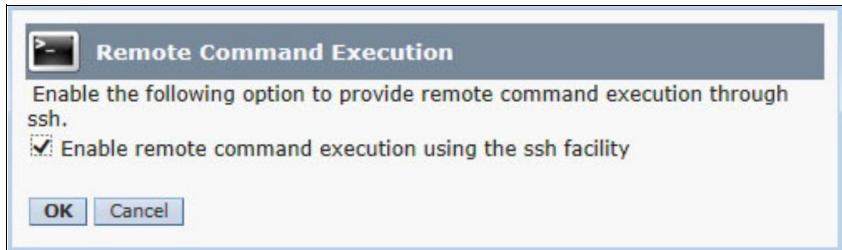


Figure 9-3 Enable remote command execution on the HMC

Note: PowerHA SystemMirror 7.2.5 for AIX or later supports a non-root user to connect with the HMC by using SSH communications. PowerHA SystemMirror 7.2.4 for AIX and earlier supports only hscroot as the HMC username. The following configuration steps assume that hscroot is being used.

Create the SSH directory \$HOME/.ssh for the root user to store the authentication keys. PowerHA runs the SSH remote DLPAR operations as the root user, which by default is /.ssh.

The following steps are what we did for our example:

1. To generate public and private keys, run the following command on each PowerHA node:

```
/usr/bin/ssh-keygen -t rsa
```

The command creates the following files in /.ssh:

```
private key: id_rsa  
public key: id_rsa.pub
```

2. The write bits for both group and other are turned off. Ensure that the private key has a permission of 600.
3. The public key for the HMC must be in the known_hosts file on each PowerHA node, and the public key for each node must be in the HMC. You can ensure this configuration by running **ssh** to the HMC from each PowerHA node. The first time that you run the command, you are prompted to insert the key into the file. Answer yes to continue.

Then, you are prompted to enter a password, which is necessary now because we have not completed the setup yet to allow non-password SSH access, as shown in Example 9-2.

Example 9-2 SSH to HMC

```
[shanley:root] / # ssh hscroot@itsohmc  
The authenticity of host 'itsohmc (192.168.100.2)' can't be established.  
RSA key fingerprint is a9:c3:74:c7:26:10:35:0b:8a:a4:22:77:6e:3c:da:64.  
Are you sure you want to continue connecting (yes/no)? yes  
Warning: Permanently added 'itsohmc,192.168.100.2' (RSA) to the list of known hosts.
```

When using two HMCs, you must repeat this process for each HMC. Also do this task among all member nodes to allow SSH-based operations (**scp**, **sftp**, and **ssh**) among them.

To allow non-password SSH access, we put each PowerHA node's public key into the **authorized_keys2** file on the HMC. You can accomplish this task in more than one way, such as consulting the HMC for information about using **mkauthkeys**.

Here is an overview of the steps that we used:

1. Copy (**scp**) the authorized_keys2 file from the HMC to the local node.
2. Concatenate (**cat**) the public key for each node into the authorized_keys2 file.
3. Repeat on each node.
4. Copy (**scp**) the concatenated file to the HMC /home/hscroot/.ssh.

Here are the detailed steps:

1. Copy the authorized_keys2 file from the HMC. To verify that the authorized_keys2 file is in the .ssh directory on the HMC, run the command that is shown in Example 9-3 from the client.

Example 9-3 Running the command from the client

```
ssh hscroot@itsohmc
hscroot@itsohmc:~> ls -al .ssh/
total 16
drwxr-xr-x 3 root      hmc 4096 2008-10-28 17:48.
drwxr-xr-x 6 hscroot  hmc 4096 2008-11-11 08:14 ..
-rw-r--r-- 1 hscroot  hmc 4070 2009-04-01 17:23 authorized_keys2
drwxrwxr-x 2 ccfw     hmc 4096 2008-10-28 17:48 ccfw
```

In the /.ssh directory, copy the authorized_keys2 file to the local node by running the following command:

```
scp hscroot@itsohmc:~/ssh/authorized_keys2 ./authorized_keys2.hmc
```

2. From /.ssh on each AIX LPAR, copy the public key and rename it to include the local node name as part of the file name. Then, copy, through **scp**, the public key of each machine (jessica and shanley) to one node (cassidy).

Run the **cat** command to create an authorized_keys2 file that contains the public key information for all PowerHA nodes. Then, use **scp** to copy the combined file to the HMC. The commands that are run on each node are shown in Example 9-4.

Example 9-4 Scp authorized_keys2 file to HMC

```
Shanley /.ssh > cp id_rsa.pub id_rsa.pub.shanley
Shanley /.ssh > scp id_rsa.pub.shanley cassidy:/.ssh/id_rsa.pub.shanley

Jessica /.ssh > cp id_rsa.pub id_rsa.pub.jessica
Jessica /.ssh > scp id_rsa.pub.jessica cassidy:/.ssh/id_rsa.pub.jessica

Cassidy /.ssh > cp id_rsa.pub id_rsa.pub.Cassidy
Cassidy /.ssh > cat id_rsa.pub.shanley id_rsa.pub.jessica id_rsa.pub.cassidy
>>authorized_keys2.hmc

[cassidy:root] /.ssh # ls -al
total 72
drwx----- 2 root      system          256 Jun  05 15:52.
drwxr-xr-x 28 root      system        4096 Jun  05 14:20 ..
-rw-r--r--  1 root      system         3194 Jun  05 15:53 authorized.key2.hmc
-rw-r--r--  1 root      system         394 May 16 09:01 authorized_keys2
-rw-----  1 root      system        1679 Jun  05 15:18 id_rsa
-rw-r--r--  1 root      system         394 Jun  05 15:18 id_rsa.pub
-rw-r--r--  1 root      system         394 Jun  05 15:51 id_rsa.pub.cassidy
-rw-r--r--  1 root      system         394 Jun  05 15:52 id_rsa.pub.jessica
-rw-r--r--  1 root      system         394 Jun  05 15:50 id_rsa.pub.shanley
-rw-r--r--  1 root      system        844 Jun  05 15:29 known_hosts
```

```
Cassidy/.ssh > scp authorized.keys2.hmc hscroot@itsohmc:~/ssh/authorized_keys2
hscroot@itsohmc's password:
authorized_keys2
100% 664 0.7KB/s 00:00
```

As you can see in Example 9-4 on page 388, when running the **scp** command to the HMC, you are prompted to enter the password for the hscroot user to copy the combined authorized_key2 file to the HMC.

3. Test whether the no-password access is working from each node by using the **ssh** command, as shown in Example 9-2 on page 387. However, this time, you should arrive at the HMC shell prompt, as shown in Example 9-5.

Example 9-5 Testing the no-password ssh access

```
[shanley:root] /ssh # ssh hscroot@itsohmc
Last login: Thu Jun 5 16:07:50 2014 from 192.168.100.52
```

When each node can use **ssh** to the HMC without a password, then this step is completed and PowerHA verification of the HMC communications succeeds.

PowerHA ROHA configuration by using the System Management Interface Tool panels

To support the ROHA feature, the PowerHA SystemMirror System Management Interface Tool (SMIT) has both menu and **c1mgr** command options. These options include the following functions:

- ▶ HMC configuration
- ▶ Hardware Resource Provisioning for Application Controller
- ▶ Cluster tunable configuration

Figure 9-4 shows a summary of the SMIT menu navigation for all ROHA panels.

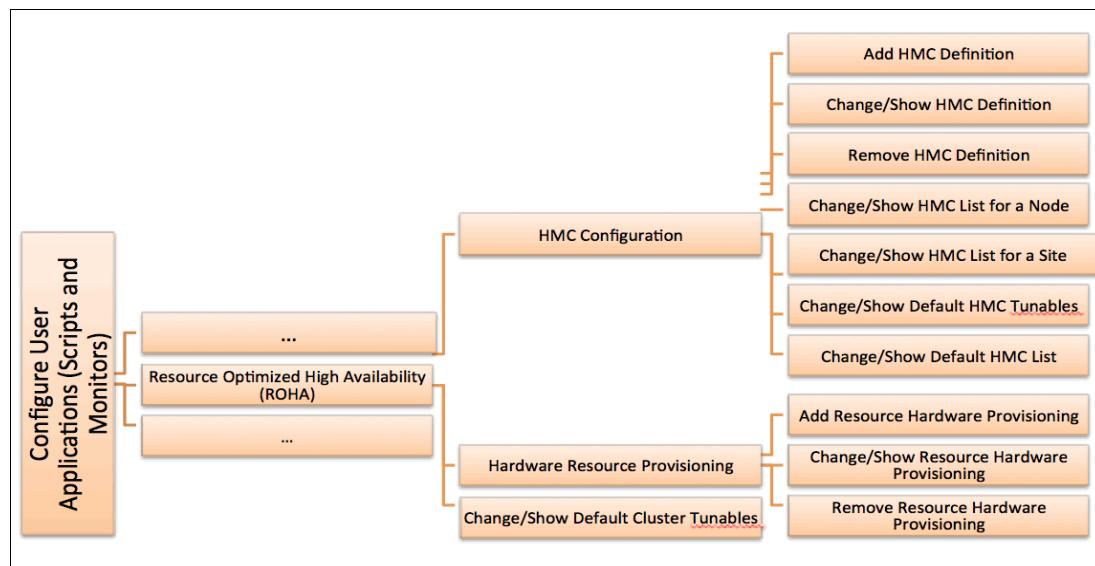


Figure 9-4 ROHA SMIT panels

HMC configuration

To define the HMC configuration to the cluster, complete the following steps:

1. Enter `smit sysmirror`, and select **Cluster Applications and Resources** → **Resources** → **Configure User Applications (Scripts and Monitors)** → **Resource Optimized High Availability** → **HMC Configuration**.
2. The next panel is a menu panel with a title line and seven menu options, as shown in Figure 9-5. Its fast path is `smitty cm_cfg_hmc`.

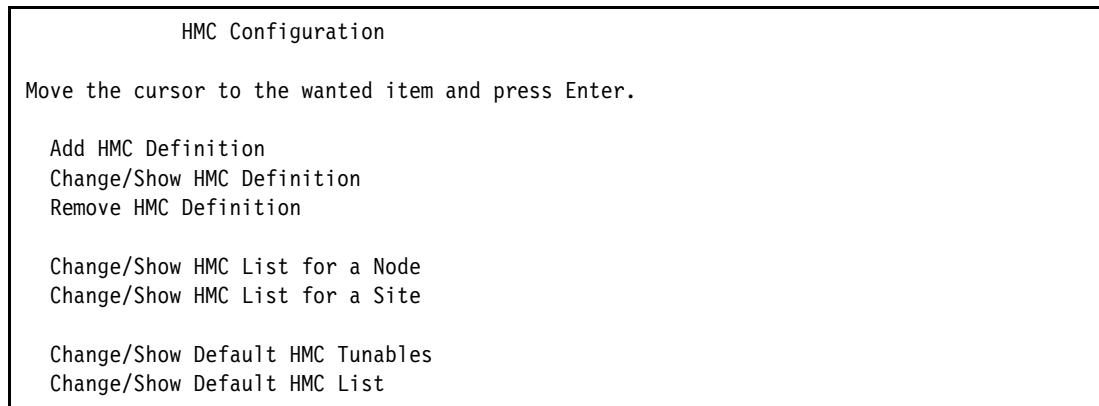


Figure 9-5 HMC configuration menu

Table 9-2 shows the help information for the HMC Configuration menu.

Table 9-2 Context-sensitive help for the HMC Configuration menu

Name and fast path	Context-sensitive help (F1)
Add HMC Definition # smitty cm_cfg_add_hmc	Select this option to add an HMC and its communication parameters and add this HMC to the default list. All the nodes of the cluster use, by default, these HMC definitions to perform DLPAR operations, unless you associate a particular HMC to a node.
Change/Show HMC Definition # smitty cm_cfg_ch_hmc	Select this option to modify or view an HMC hostname and its communication parameters.
Remove HMC Definition # smitty cm_cfg_rm_hmc	Select this option to remove an HMC, and then remove it from the default list.
Change/Show HMC List for a Node # smitty cm_cfg_hmcs_node	Select this option to modify or view the list of an HMC of a node.
Change/Show HMC List for a Site # smitty cm_cfg_hmcs_site	Select this option to modify or view the list of an HMC of a site.
Change/Show Default HMC Tunables # smitty cm_cfg_def_hmc_tun	Select this option to modify or view the HMC default communication tunables.
Change/Show Default HMC List # smitty cm_cfg_def_hmcs	Select this option to modify or view the default HMC list that is used by default by all nodes of the cluster. Nodes that define their own HMC list do not use this default HMC list.

Add HMC Definition menu

Note: Before you add an HMC, you must set up password-less communication from AIX nodes to the HMC. For more information, see “HMC configuration” on page 390.

To add an HMC, select Add HMC Definition. The next panel is a dialog panel with a title dialog header and several dialog command options. Its fast path is `smitty cm_cfg_add_hmc`. Each item has a context-sensitive help window that you can access by pressing F1. You can also see any associated list for each item by pressing F4.

Figure 9-6 shows the menu to add the HMC definition and its entry fields.

Add HMC Definition	
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.	
[Entry Fields]	
* HMC name	[] +
DLPAR operations timeout (in minutes)	[] #
Number of retries	[] #
Delay between retries (in seconds)	[] #
Nodes	[] +
Sites	[] +
User name	[]
Check connectivity between HMC and nodes	Yes

Figure 9-6 Add HMC Definition menu

If the domain name server (DNS) is configured in your environment and can resolve the HMC IP and hostname, then you can press F4 to select an HMC to be added.

Figure 9-7 shows an example of selecting one HMC from the list to perform the add operation.

HMC name		
Move the cursor to the wanted item and press Enter.		
e16hmc1 is 9.3.207.130		
e16hmc3 is 9.3.207.133		
F1=Help Esc+8=Image /=Find	F2=Refresh Esc+0=Exit n=Find Next	F3=Cancel Enter=Do

Figure 9-7 Selecting the HMC to add from the HMC list

Table 9-3 shows the help panel describing all the options that are available for use when adding the HMC definition.

Table 9-3 Context-sensitive help and associated list for the Add HMC Definition menu

Name	Context-sensitive help (F1)	Associated list (F4)
HMC name	Enter the hostname for the HMC. An IP address is also accepted here. Both IP version 4 (IPv4) and IPv6 addresses are supported.	Yes (single-selection). The list is obtained by running the following command: <code>/usr/sbin/rsct/bin/rmcdomainstatus -s ctrmc -a IP</code>
DLPAR operations timeout (in minutes)	Enter a timeout in minutes by using DLPAR commands that you run on an HMC (use the <code>-w</code> parameter). The <code>-w</code> parameter is used with the <code>chhwres</code> command only when allocating or releasing resources. The parameter is adjusted according to the type of resources (for memory, 1 minute per gigabyte is added to this timeout). Setting no value means that you use the default value, which is defined in the Change>Show Default HMC Tunables panel. When <code>-1</code> is displayed in this field, it indicates that the default value is used.	None.
Number of retries	Enter the number of times that an HMC command is retried before the HMC is considered as non-responsive. The next HMC in the list is used after this number of retries fails. Setting no value means that you use the default value, which is defined in the Change>Show Default HMC Tunables panel. When <code>-1</code> is displayed in this field, it indicates that the default value is used.	None.
Delay between retries (in seconds)	Enter a delay in seconds between two successive retries. Setting no value means that you use the default value, which is defined in the Change>Show Default HMC Tunables panel. When <code>-1</code> is displayed in this field, it indicates that the default value is used.	None.
Nodes	Enter the list of nodes that use this HMC.	Yes (multiple-selection). A list of nodes to be proposed can be obtained by running the following command: <code>odmget HACMPnode</code>
Sites	Enter the sites that use this HMC. All nodes of the sites then use this HMC by default, unless the node defines an HMC at its own level.	Yes (multiple-selection). A list of sites to be proposed can be obtained by running the following command: <code>odmget HACMPSite</code>
Check the connectivity among the HMC and the nodes.	Select Yes to check the communication links among the nodes and the HMC.	<Yes> <No>. The default is Yes.

PowerHA SystemMirror also supports entering the HMC IP address to add the HMC. Figure 9-8 shows an example of entering an HMC IP address to add the HMC.

Add HMC Definition	
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.	
* HMC name	[Entry Fields] [9.3.207.130] +
DLPAR operations timeout (in minutes)	[]
Number of retries	[]
Delay between retries (in seconds)	[]
Nodes	[] +
Sites	[] +
User Name	[]
Check connectivity between HMC and nodes	Yes

Figure 9-8 Entering an HMC IP address to add the HMC

Change>Show HMC Definition menu

To show or modify an HMC, select Change>Show HMC Definition. The next panel is an auto-generated list that lists all the existing HMC names, as shown in Figure 9-9. Its fast path is `smitty cm_cfg_ch_hmc`.

HMC name
Move the cursor to the wanted item and press Enter.
e16hmc1
e16hmc3
F1=Help F2=Refresh F3=Cancel
Esc+8=Image Esc+0=Exit Enter=Do
/=Find n=Find Next

Figure 9-9 Selecting an HMC from a list to change or show an HMC configuration

To modify an existing HMC, select it and press Enter. The next panel is the one that is shown in Figure 9-10. The HMC name cannot be changed. To change the name, the HMC definition must be deleted and readded.

Change/Show HMC Definition	
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.	
* HMC name	[Entry Fields] e16hmc1
DLPAR operations timeout (in minutes)	[5] #
Number of retries	[3] #
Delay between retries (in seconds)	[10] #
Nodes	[ITSO_rar1m3_Node1 ITSO_r1r9m1_Node1] +
Sites	[] +
Check connectivity between HMC and nodes	Yes

Figure 9-10 Change/Show HMC Definition menu

Remove HMC Definition menu

To delete an HMC, select Remove HMC Definition. The panel that is shown in Figure 9-11 is the auto-generated list. To remove an existing HMC name, select it from the list and press Enter. Its fast path is `smitty cm_cfg_rm_hmc`.

HMC name		
Move the cursor to the wanted item and press Enter.		
e16hmc1		
e16hmc3		
F1=Help	F2=Refresh	F3=Cancel
Esc+8=Image	Esc+0=Exit	Enter=Do
/=Find	n=Find Next	

Figure 9-11 Selecting an HMC to remove

The next panel is shown in Figure 9-12. Press Enter to remove the HMC definition.

Remove HMC Definition	
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.	
* HMC name	[Entry Fields] e16hmc1

Figure 9-12 Removing an HMC

Change>Show HMC List for a Node menu

To show or modify the HMC list for a node, select Change>Show HMC List for a Node. The next panel is an auto-generated list, as shown in Figure 9-13. Choose the wanted node and press Enter. The fast path to this panel is **smitty cm_cfg_hmc_node**.

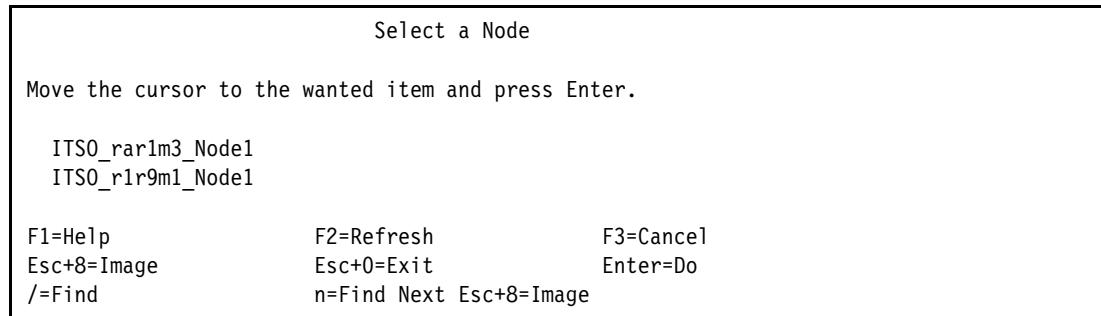


Figure 9-13 Selecting a node to change

To modify an existing node, select it and press Enter. The next panel (Figure 9-14) is a dialog panel with a title dialog header and two dialog command options.

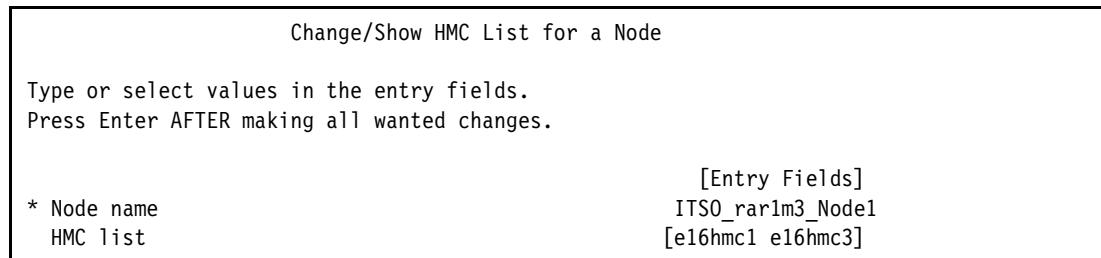


Figure 9-14 Change>Show HMC List for a Node menu

You cannot add or remove an HMC from this list. You can order only the list of the HMCs that are used by the node in the correct precedence order.

Table 9-4 shows the help information for the Change>Show HMC List for a Node menu.

Table 9-4 Context-sensitive help for the Change>Show HMC List for a Node menu

Name and fast path	Context-sensitive help (F1)
Node name	This node name to associate with one or more HMCs.
HMC list	The precedence order of the HMCs that are used by this node. The first in the list is tried first, then the second, and so on. You cannot add or remove any HMC. You can modify only the order of the already set HMCs.

Change>Show HMC List for a Site menu

To show or modify the HMC list for a node, select Change>Show HMC List for a Site. The next panel is an auto-generated list, as shown in Figure 9-15. Select the wanted site and press Enter. Its fast path is **smitty cm_cfg_hmc_site**.

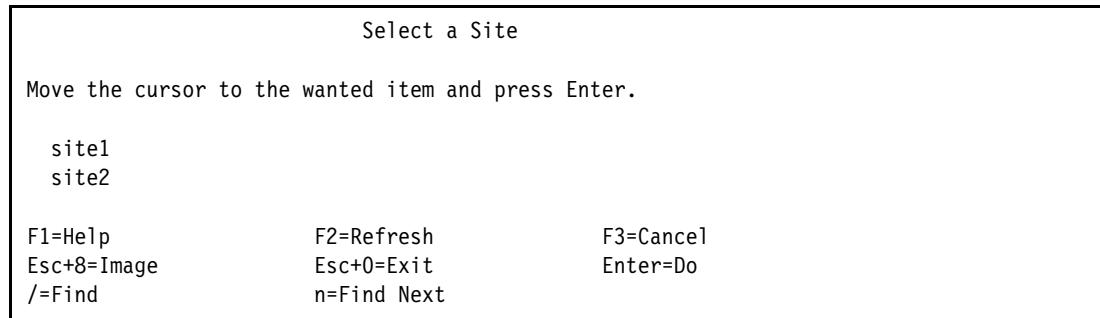


Figure 9-15 Select a Site menu

To modify an existing site, select it and press Enter. The next panel (Figure 9-16) is a dialog panel with a title dialog header and two dialog command options.

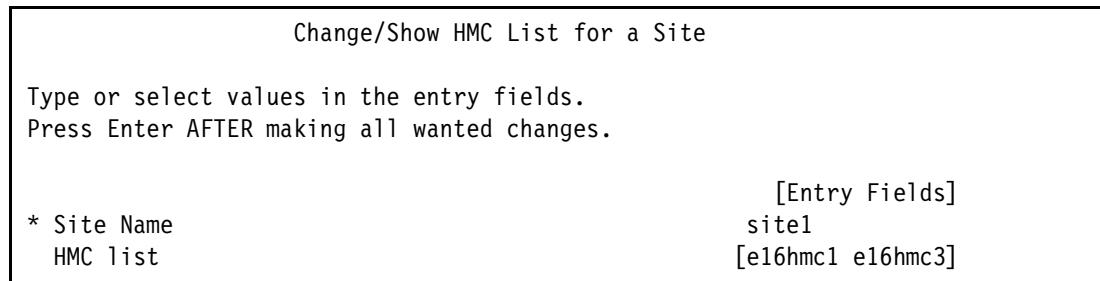


Figure 9-16 Change>Show HMC List for a Site menu

Again, you cannot add or remove an HMC from the list. You can reorder only the HMCs that are used by the site. Table 9-5 shows the help information for the Change>Show HMC List for a Site menu.

Table 9-5 Site and HMC usage list

Name and fast path	Context-sensitive help (F1)
Site name	The site name to associate with one or more HMCs.
HMC list	The precedence order of the HMCs that are used by this site. The first in the list is tried first, then the second, and so on. You cannot add or remove any HMC. You can modify only the order of the already set HMCs.

Change>Show Default HMC Tunables menu

To show or modify the default HMC communication tunables, select Change>Show Default HMC Tunables. The next panel (Figure 9-17) is a dialog panel with a title dialog header and three dialog command options. Its fast path is **smitty cm_cfg_def_hmc_tun**. Each item has an available context-sensitive help window that you can open by pressing F1. Any existing selectable items can be found by pressing F4 in a field.

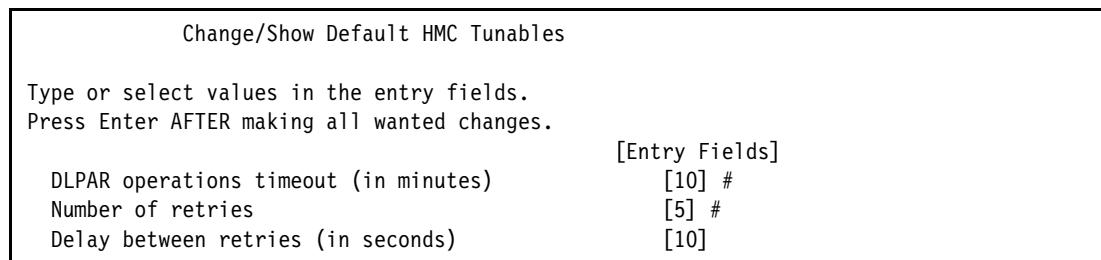


Figure 9-17 Change>Show Default HMC Tunables menu

Change>Show Default HMC List menu

To show or modify the default HMC list, select Change>Show Default HMC List. The next panel (Figure 9-18) is a dialog panel with a title dialog header and one dialog command option. Its fast path is **smitty cm_cfg_def_hmc**s. Each item has an available context-sensitive help window that you can open by pressing F1. Any existing selectable items can be found by pressing F4 in the correct field.

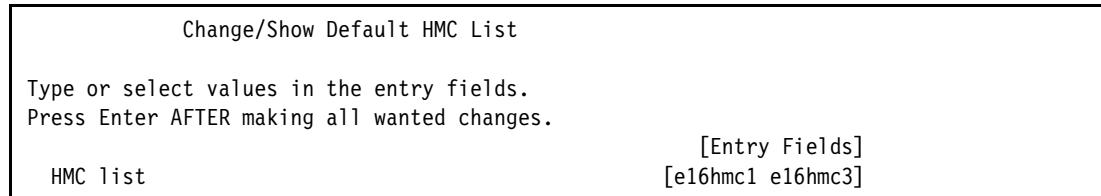


Figure 9-18 Change>Show Default HMC List menu

Hardware resource provisioning for an application controller

To provision hardware for an application controller, complete the following steps:

1. Start **smit sysmirror**, and select **Cluster Applications and Resources → Resources → Configure User Applications (Scripts and Monitors) → Resource Optimized High Availability → Hardware Resource Provisioning for Application Controller**, as shown in Figure 9-19.

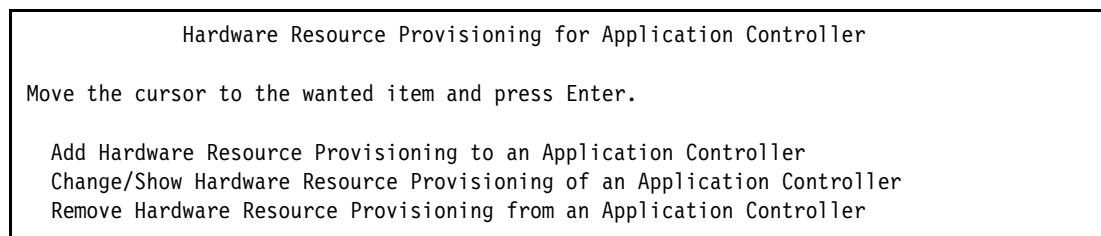


Figure 9-19 Hardware Resource Provisioning for Application Controller menu

2. Select one of the following actions:

- To add an application controller configuration, select Add.
- To change or show an application controller configuration, select Change/Show.
- To remove an application controller configuration, select Remove.

The first time that you select Add or Change/Show, the On/Off CoD Agreement is displayed, as shown in Figure 9-20. However, it is displayed *only* if the user has *not* yet agreed to it. If the user already agreed to it, it is not displayed.

On/Off CoD Agreement menu

If you have not previously accepted the On/Off CoD agreement, a panel opens so that you may accept the agreement, as shown in Figure 9-20.

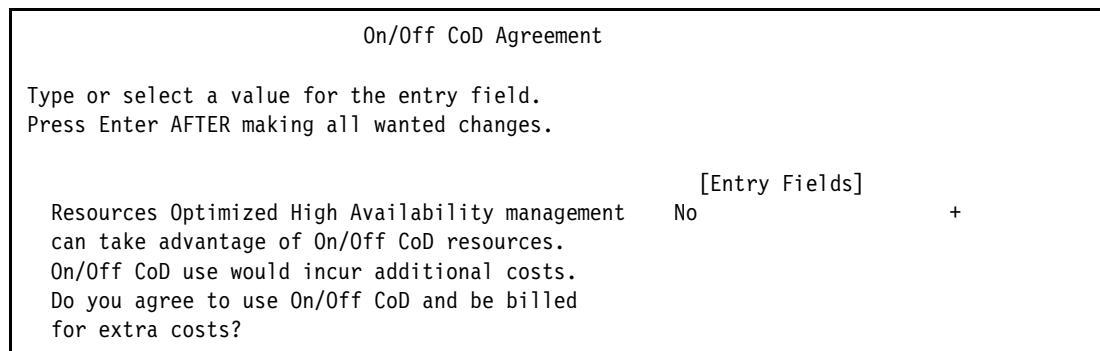


Figure 9-20 On/Off CoD Agreement menu

To accept the On/Off CoD Agreement, complete the following steps:

1. Enter Yes to have PowerHA SystemMirror use On/Off CoD resources to perform DLPAR operations on your nodes.
2. If you agree to use On/Off CoD, you must ensure that you entered the On/Off CoD activation code. The On/Off CoD license key must be entered into HMC before PowerHA SystemMirror can activate this type of resources.
3. In the following cases, keep the default value:
 - If there is only EPCoD, keep the default value of No.
 - If there is not EPCoD or On/Off CoD, PowerHA manages only the server's permanent resources through DLPAR, so keep the default value.

Add Hardware Resource Provisioning to an Application Controller menu

The next panel is an auto-generated list that lists all existing application controllers, as shown in Figure 9-21.

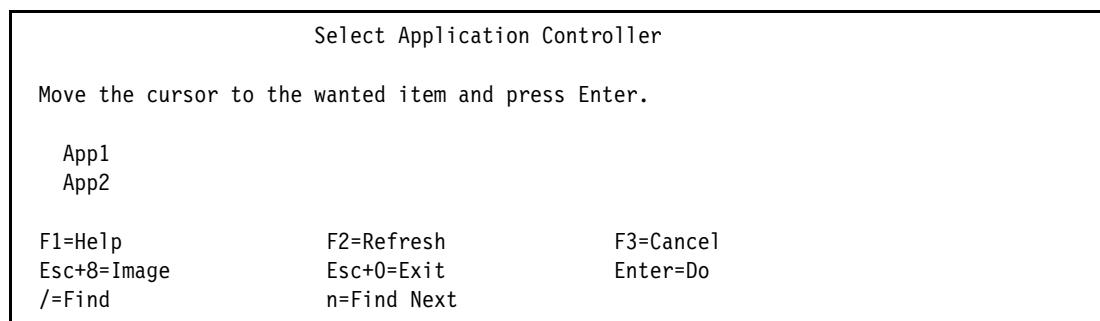


Figure 9-21 Select Application Controller menu

To add hardware resource provisioning for an application controller, the list displays only application controllers that do not already have hardware resource provisioning, as shown in Figure 9-22.

Add Hardware Resource Provisioning to an Application Controller	
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.	
* Application Controller Name	[Entry Fields] App1
Use wanted level from the LPAR profile	No +
Optimal number of gigabytes of memory	[]
Optimal number of dedicated processors	[] #
Optimal number of processing units	[]
Optimal number of virtual processors	[]

Figure 9-22 Add Hardware Resource Provisioning to an Application Controller menu

To modify an existing application controller, select it and press Enter. Each item has an available context-sensitive help window that you can open by pressing F1. Any existing selectable items can be found by pressing F4 in the correct field.

To modify or remove hardware resource provisioning for an application controller, the list displays *only* application controllers that already have hardware resource provisioning.

Table 9-6 shows the help for adding hardware resources.

Table 9-6 Context-sensitive help for adding hardware resource provisioning

Name and fast path	Context-sensitive help (F1)
Application Controller Name	The application controller for which you configure DLPAR and CoD resource provisioning.
Use the wanted level from the LPAR profile	<p>There is no default value. Make one of the following choices:</p> <ul style="list-style-type: none"> ▶ Enter Yes if you want the LPAR hosting your node to reach only the level of resources that is indicated by the wanted level of the LPAR's profile. By selecting Yes, you trust the wanted level of the LPAR profile to fit the needs of your application controller. ▶ Enter No if you prefer to enter exact optimal values for memory, processor (CPU), or both. These optimal values match the needs of your application controller, and you have better control of the level of resources that are allocated to your application controller. ▶ Enter nothing if you do not need to provision any resource for your application controller. <p>For all application controllers that have this tunable set to Yes, the allocation that is performed lets the LPAR reach the LPAR wanted value of the profile.</p> <p>Suppose that you have a mixed configuration, in which some application controllers have this tunable set to Yes, and other application controllers have this tunable set to No with some optimal level of resources that is specified. In this case, the allocation that is performed lets the LPAR reach the wanted value of the profile that is added to the optimal values.</p>

Name and fast path	Context-sensitive help (F1)
Optimal number of gigabytes of memory	<p>Enter the amount of memory that PowerHA SystemMirror attempts to acquire for the node before starting this application controller.</p> <p>This Optimal number of gigabytes of memory value can be set only if the Used wanted level from the LPAR profile value is set to No.</p> <p>Enter the value in multiples of 0.25, 0.5, 0.75, or 1 GB. For example, 1 represents 1 GB or 1024 MB, 1.25 represents 1.25 GB or 1280 MB, 1.50 represents 1.50 GB or 1536 MB, and 1.75 represents 1.75 GB or 1792 MB.</p> <p>If this amount of memory is not satisfied, PowerHA SystemMirror takes RG recovery actions to move the RG with this application to another node.</p> <p>Alternatively, PowerHA SystemMirror can allocate less memory depending on the Start RG even if resources are insufficient cluster tunable.</p>
Optimal number of dedicated processors	<p>Enter the number of processors that PowerHA SystemMirror attempts to allocate to the node before starting this application controller.</p> <p>This attribute is only for nodes running on an LPAR with Dedicated Processing Mode.</p> <p>This Optimal number of dedicated processors value can be set only if the Used wanted level from the LPAR profile value is set to No.</p> <p>If this number of CPUs is not satisfied, PowerHA SystemMirror takes RG recovery actions to move the RG with this application to another node.</p> <p>Alternatively, PowerHA SystemMirror can allocate fewer CPUs depending on the Start RG even if resources are insufficient cluster tunable.</p>
Optimal number of processing units	<p>Enter the number of processing units that PowerHA SystemMirror attempts to allocate to the node before starting this application controller.</p> <p>This attribute is only for nodes running on an LPAR with Shared Processing Mode.</p> <p>This Optimal number of processing units value can be set only if the Used wanted level from the LPAR profile value is set to No.</p> <p>Processing units are specified as a decimal number with two decimal places, 0.01 - 255.99.</p> <p>This value is used only on nodes that support allocation of processing units.</p> <p>If this number of CPUs is not satisfied, PowerHA SystemMirror takes RG recovery actions to move the RG with this application to another node.</p> <p>Alternatively, PowerHA SystemMirror can allocate fewer CPUs depending on the Start RG even if resources are insufficient cluster tunable.</p>
Optimal number of virtual processors	<p>Enter the number of virtual processors that PowerHA SystemMirror attempts to allocate to the node before starting this application controller.</p> <p>This attribute is only for nodes running on an LPAR with Shared Processing Mode.</p> <p>This Optimal number of dedicated or virtual processors value can be set only if the Used wanted level from the LPAR profile value is set to No.</p> <p>If this number of virtual processors is not satisfied, PowerHA SystemMirror takes RG recovery actions to move the RG with this application to another node.</p> <p>Alternatively, PowerHA SystemMirror can allocate fewer CPUs depending on the Start RG even if resources are insufficient cluster tunable.</p>

To modify an application controller configuration, click Change>Show. The next panel is an auto-generated list, as shown in Figure 9-22 on page 399. To modify an existing application controller, select it and press Enter. The next panel is the same dialog panel that is shown in Figure 9-22 on page 399, except the title, which is different.

To delete an application controller configuration, click **Remove**. The next panel is an auto-generated list, as shown previously. To remove an existing application controller, select it and press Enter.

If Use wanted level from the LPAR profile is set to No, then at least the memory (Optimal number of gigabytes of memory) or CPU (Optimal number of dedicated or virtual processors) setting is mandatory.

Change>Show Default Cluster Tunable menu

Start **smit sysmirror**, and select **Cluster Applications and Resources → Resources → Configure User Applications (Scripts and Monitors) → Resource Optimized High Availability → Change>Show Default Cluster Tunables**. The next panel (Figure 9-23) is a dialog panel with a title dialog header and seven dialog command options. Each item has an available context-sensitive help window that you can open by pressing F1. Any existing selectable items can be found by pressing F4 in the wanted field. Its fast path is **smitty cm_cfg_def_cl_tun**.

Change>Show Default Cluster Tunables			
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.			
[Entry Fields]			
Dynamic LPAR			
Always Start Resource Groups	Yes	+ No	+ No
Adjust Shared Processor Pool size if required	No	+ Yes	+ Yes
Force synchronous release of DLPAR resources	No	+ Yes	+ Yes
Enterprise Pool			
Resource Allocation order	Free Pool First	+	
On/Off CoD			
I agree to use On/Off CoD and be billed for extra costs	No	+	
Number of activating days for On/Off CoD requests [30]	#		
F1=Help F5=Reset F9=Shell	F2=Refresh F6=Command F10=Exit	F3=Cancel F7>Edit Enter=Do	F4>List F8=Image

Figure 9-23 Change>Show Default Cluster Tunables menu

Table 9-7 shows the help for the cluster tunables.

Table 9-7 Context-sensitive help for Change/Show Default Cluster Tunables menu

Name and fast path	Context-sensitive help (F1)
Always start Resource Groups	<p>Enter Yes to have PowerHA SystemMirror start RGs even if there are errors in ROHA resources activation. Errors can occur when the total requested resources exceed the LPAR profile's maximum or the combined available resources, or if there is a total loss of HMC connectivity. Thus, the best-can-do allocation is performed.</p> <p>Enter No to prevent starting RGs if any errors occur during ROHA resources acquisition.</p> <p><i>The default is Yes.</i></p>
Adjust Shared Processor Pool size if required	<p>Enter Yes to authorize PowerHA SystemMirror to dynamically change the user-defined Shared Processors Pool boundaries, if necessary. This change can occur only at takeover, and only if CoD resources are activated for the central processing complex (CPC) so that changing the maximum size of a particular Shared Processors Pool is not done to the detriment of other Shared Processors Pools.</p> <p><i>The default is No.</i></p>
Force synchronous release of DLPAR resources	<p>Enter Yes to have PowerHA SystemMirror release CPU and memory resources synchronously. For example, if the client must free resources on one side before they can be used on the other side. By default, PowerHA SystemMirror automatically detects the resource release mode by looking at whether Active and Backup nodes are on the same or different CECs. A best practice is to have an asynchronous release to not delay the takeover.</p> <p><i>The default is No.</i></p>
I agree to use On/Off CoD and be billed for extra costs	<p>Enter Yes to have PowerHA SystemMirror use On/Off CoD to obtain enough resources to fulfill the optimal amount that is requested. Using On/Off CoD requires an activation code to be entered on the HMC and can result in extra costs due to the usage of the On/Off CoD license.</p> <p><i>The default is No.</i></p>
Number of activating days for On/Off CoD requests	<p>Enter the number of activating days for On/Off CoD requests. If the requested available resources are insufficient for this duration, then longest-can-do allocation is performed. Try to allocate the amount of resources that is requested for the longest duration. To do so, consider the overall resources that are available. This number is the sum of the On/Off CoD resources that are already activated but not yet used, and the On/Off CoD resources that are not yet activated.</p> <p><i>The default is 30.</i></p>

PowerHA SystemMirror ROHA verification

The ROHA function enables PowerHA SystemMirror to automatically or manually check for environment discrepancies. The **c1verify** tool has ROHA-related configuration integrity validation checks.

The verification tool can be used to ensure that the environment is correct regarding a ROHA setup. Discrepancies are called out by PowerHA SystemMirror, and the tool helps correct the configuration if possible.

The results appear in the following files:

- ▶ The /var/hacmp/log/c1verify.log file
- ▶ The /var/hacmp/log/autoverify.log file

The user is actively notified of critical errors. A distinction can be made between errors that are raised during configuration and errors that are raised during cluster synchronization.

As a general principal, any problems that are detected at configuration time are presented as warnings instead of errors. Another general principle is that PowerHA SystemMirror checks only what is being configured at configuration time and not the whole configuration. PowerHA SystemMirror checks the whole configuration at verification time.

For example, when adding an HMC, you check only the new HMC (verify that it is pingable, at an appropriate software level, and so on) and not *all* the HMCs. Checking the whole configuration can take some time, and is done at verify and sync time rather than each individual configuration step.

General verification

Table 9-8 shows the general verification list.

Table 9-8 General verification list

Item	Configuration time	Synchronization time
Check that all RG active and standby nodes are on different CPCs, which enable the asynchronous mode of releasing resources.	Info	Warning
This code cannot run on an IBM Power4 processor-based server.	Error	Error

HMC communication verification

Table 9-9 shows the HMC communication verification list.

Table 9-9 HMC communication verification list

Item	Configuration time	Synchronization time
Only one HMC is configured per node.	None	Warning
Two HMCs are configured per node.	None	OK
One node is without an HMC (if ROHA only).	None	Error
Only one HMC per node can be pinged.	Warning	Warning
Two HMCs per node can be pinged.	OK	OK
One node has a non-pingable HMC.	Warning	Error
Only one HMC with password-less SSH communication exists per node.	Warning	Warning
Two HMCs with password-less SSH communication exist per node.	OK	OK
One node exists with a non-SSH accessible HMC.	Warning	Error
Check that all HMCs share the level (the same version of HMC).	Warning	Warning

Item	Configuration time	Synchronization time
Check that all HMCs administer the CPC hosting the current node. Configure two HMCs administering the CPC that is hosting the current node. If not, PowerHA gives a warning message.	Warning	Warning
Check whether the HMC level supports FSP Lock Queuing.	Info	Info

CoD verification

Table 9-10 shows the CoD verification.

Table 9-10 CoD verification

Item	Configuration time	Synchronization time
Check that all CPCs are CoD-capable.	Info	Warning
Check whether CoD is enabled.	Info	Warning

Power Enterprise Pool verification

Table 9-11 shows the enterprise pool verification list.

Table 9-11 Power Enterprise Pool verification

Item	@info	@Sync
Check that all CPCs are Enterprise Pool-capable.	Info	Info
Determine which HMC is the master, and which HMC is the non-master.	Info	Info
Check that the nodes of the cluster are on different pools, which enables the asynchronous mode of releasing resources.	Info	Info
Check that all HMCs are at Version 7.8 or later.	Info	Warning
Check that the CPC has unlicensed resources.	Info	Warning

Resource provisioning verification

Table 9-12 shows the resource provisioning verification information.

Table 9-12 Resource provisioning verification

Item	@info	@Sync
Check that for one node that the total of optimal memory (of RGs on this node) that is added to the profile's minimum does not exceed the profile's maximum.	Warning	Error
Check that for one node that the total of optimal CPU (of RGs on this node) that is added to the profile's minimum does not exceed the profile's maximum.	Warning	Error

Item	@info	@Sync
Check that for one node that the total of optimal PU (of RGs on this node) that is added to the profile's minimum does not exceed the profile's maximum.	Warning	Error
Check that the total processing units do not break the minimum processing units per virtual processor ratio.	Error	Error

9.3.4 Troubleshooting HMC verification errors

Errors might be encountered during verification. These errors are generated for various reasons, some of which are explained here. Although many error messages seem self-explanatory, tips about troubleshooting can help.

Example 9-6 shows an error message that gives good probable causes for a problem. However, the following two actions can help you to discover the source of the problem:

- ▶ Ping the HMC IP address.
- ▶ Use the `ssh hscroot@hmcip` command to the HMC.

If `ssh` is unsuccessful or prompts for a password, it is an indication that SSH was not correctly configured.

Example 9-6 HMC unreachable during verification

ERROR: The HMC with IP label 192.168.100.2 configured on node Cassidy is not reachable. Make sure that the HMC IP address is correct, the HMC is turned on and connected to the network, and the HMC has OpenSSH installed and setup with the public key of node Cassidy.

If the message in Example 9-7 appears by itself, it is normally an indication that access to the HMC is working. However, the particular node's matching LPAR definition is not reporting that it is DLPAR-capable.

Example 9-7 Node not DLPAR-capable verification error

ERROR: An HMC has been configured for node Cassidy, but the node does not appear to be DLPAR capable.

This error might be caused by Resource Monitoring and Control (RMC) not updating properly. Generally, this situation is rare, and usually applies only to IBM Power4 processor-based systems. You can verify manually that the LPAR is DLPAR-capable from the HMC CLI, as shown in Example 9-8.

Example 9-8 Verifying that the LPAR is DLPAR-capable

```
hscroot@hmc3:~> lspartition -dlpar
<#0> Partition:<5*8204-E8A*10FE401, , 9.12.7.5>
      Active:<0>, OS:<, , >, DCaps:<0x0>, CmdCaps:<0x0, 0x0>, PinnedMem:<0>

<#3> Partition:<4*8233-E8B*100C99R, cassidy, 192.168.100.52>
      Active:<1>, OS:<AIX, 7.1, 7100-03-02-1412>, DCaps:<0x2c5f>, CmdCaps:<0x1b,
      0x1b>, PinnedMem:<1035>
```

Note: The HMC command syntax can vary by HMC code level and type.

In partition #0, which is shown in the example, it is *not* DLPAR-capable, as indicated by DCaps :<0x0>. Partition #3 *is* DLPAR-capable. In this case, because partition #0 is not active, it might be a truer indication of why the partition is not currently DLPAR capable.

Also, be sure that RMC communication to HMC (port 657) is working and restart the RSCT daemons on the partitions by running the following commands in this order on the cluster node:

1. /usr/sbin/rsct/install/bin/recfgct
2. /usr/sbin/rsct/bin/rmcctrl -z
3. /usr/sbin/rsct/bin/rmcctrl -A
4. /usr/sbin/rsct/bin/rmcctrl -p

Also, restart the RSCT daemons on HMC the same as described here, but you must first become root from the product engineering shell (**pesh**) and use the hscpe user profile to do so. This action often requires getting a **pesh** password from IBM Support.

During our testing, we ran several events within short periods. At certain points, our LPAR reported that it was no longer DLPAR-capable. Then, after a short period, it reported normally again. We believe that this event occurred because the RMC information became out-of-sync between the LPARs and the HMC, so ultimately it was a timing issue.

9.4 ROHA testing

This section describes two different test scenarios that use ROHA in a cluster. The first example (9.4.1, “Example 1: Setting up a ROHA cluster without On/Off CoD” on page 406) is done by using Enterprise Pools without using On/Off CoD. The second example (9.4.3, “Example 2: Setting up one ROHA cluster with On/Off CoD” on page 427) uses both Enterprise Pools and On/Off CoD.

9.4.1 Example 1: Setting up a ROHA cluster without On/Off CoD

This section describes how to set up a ROHA cluster without On/Off CoD.

Hardware requirements

- ▶ Two IBM Power 770 D model servers, both in one Power Enterprise Pool.
- ▶ One PowerHA SystemMirror cluster with two nodes that are in different servers.
- ▶ The PowerHA SystemMirror cluster manages the server’s available resources and EPCoD mobile resources to automatically satisfy the application’s hardware requirements before it starts.

Hardware topology

The hardware topology is shown in Figure 9-24 on page 407.

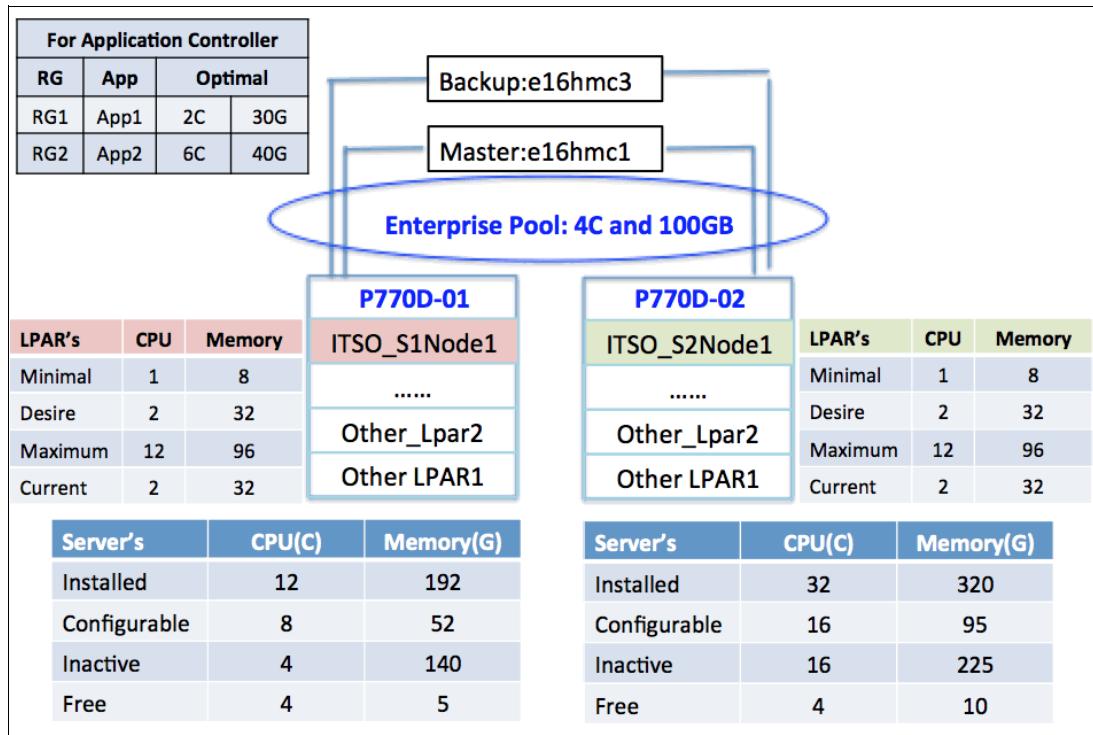


Figure 9-24 Hardware topology for Example 1

The topology includes the following components for the configuration:

- ▶ Two Power 770 D model servers, which are named P770D-01 and P770D-02:
 - P770D-01 has four inactive CPUs, 140 GB of inactive memory, four available CPUs, and 5 GB of available memory.
 - P770D-02 has 16 inactive CPUs, 225 GB of inactive memory, four available CPUs, and 10 GB of available memory.
- ▶ One Power Enterprise Pool with four mobile processors and 100 GB mobile memory resources.
- ▶ The PowerHA SystemMirror cluster includes two nodes, ITSO_S1Node1 and ITSO_S2Node1.
- ▶ This topology also includes the profile configuration for each LPAR.

There are two HMCs to manage the EPCoD, which are named e16hmc1 and e16hmc3. Here, e16hmc1 is the master and e16hmc3 is the backup. There are two applications in this cluster and the related resource requirements.

Cluster configuration

The cluster configuration is shown in Table 9-13.

Table 9-13 Cluster's attributes

Attribute	ITSO_S1Node1	ITSO_S2Node2
Cluster name	ITSO_ROHA_cluster Cluster type: No Site Cluster (NSC)	
Network interface	en0: 10.40.1.218 Netmask: 255.255.254.0 Gateway: 10.40.1.1	en0: 10.40.0.11 Netmask: 255.255.254.0 Gateway: 10.40.1.1
Network	net_ether_01 (10.40.0.0/23)	
CAA	Unicast Primary disk: repdisk1 Backup disk: repdisk2	
Shared VG	shareVG1: hdisk18 shareVG2: hdisk19	shareVG1: hdisk8 shareVG2: hdisk9
Application controller	<ul style="list-style-type: none">▶ App1Controller: /home/bing/app1start.sh /home/bing/app1stop.sh▶ App2Controller: /home/bing/app2start.sh /home/bing/app2stop.sh	
Service IP	10.40.1.61 ITSO_ROHA_service1 10.40.1.62 ITSO_ROHA_service2	
RG	RG1 includes shareVG1, ITSO_ROHA_service1, and App1Controller. RG2 includes shareVG2, ITSO_ROHA_service2, and App2Controller. The node order is ITSO_S1Node1 ITSO_S2Node1. Startup Policy: Online on Home Node Only (OHNO) Failover Policy: Failover To Next Priority Node In The List Fallback Policy: Never Fallback	

ROHA configuration

The ROHA configuration includes the HMC, hardware resource provisioning, and the cluster-wide tunable configuration.

HMC configuration

There are two HMCs to add, as shown in Table 9-14 and Table 9-15 on page 409.

Table 9-14 Configuration of HMC1

Items	Value
HMC name	9.3.207.130 ^a
DLPAR operations timeout (in minutes)	3
Number of retries	2
Delay between retries (in seconds)	5
Nodes	ITSO_S1Node1 ITSO_S2Node1

Items	Value
Sites	N/A
Check connectivity between HMC and nodes?	Yes (default)

- a. Enter one HMC name, not an IP address, or select one HMC and then press F4 to show the HMC list. PowerHA SystemMirror also supports an HMC IP address.

Table 9-15 Configuration of HMC2

Items	Value
HMC name	9.3.207.133 ^a
DLPAR operations timeout (in minutes)	3
Number of retries	2
Delay between retries (in seconds)	5
Nodes	ITSO_S1Node1 ITSO_S2Node1
Sites	N/A
Check connectivity between HMC and nodes?	Yes (default)

- a. Enter one HMC name, not an IP address, or select one HMC and then press F4 to show the HMC list. PowerHA SystemMirror also supports an HMC IP address.

Also, in /etc/hosts, there are resolution details between the HMC IP and the HMC hostname, as shown in Example 9-9.

Example 9-9 The /etc/hosts file for Example 1 and Example 2

```
10.40.1.218 ITSO_S1Node1
10.40.0.11 ITSO_S2Node1
10.40.1.61 ITSO_ROHA_service1
10.40.1.62 ITSO_ROHA_service2
9.3.207.130 e16hmc1
9.3.207.133 e16hmc3
```

Hardware resource provisioning for application controller

There are two application controllers to add, as shown in Table 9-16 and Table 9-17.

Table 9-16 Configuration for AppController1

Items	Value
I agree to use On/Off CoD and be billed for extra costs	No (default)
Application Controller Name	AppController1
Use the wanted level from the LPAR profile	No
Optimal number of gigabytes of memory	30
Optimal number of dedicated processors	2

Table 9-17 Configuration for AppController2

Items	Value
I agree to use On/Off CoD and be billed for extra costs	No (default)
Application Controller Name	AppController2
Use the wanted level from the LPAR profile	No
Optimal number of gigabytes of memory	40
Optimal number of dedicated processors	6

Cluster-wide tunables

All the tunables use the default values, as shown in Table 9-18.

Table 9-18 ROHA cluster tunables

Items	Value
DLPAR Start Resource Groups even if resources are insufficient	No (default)
Adjust Shared Processor Pool size if required	No (default)
Force synchronous release of DLPAR resources	No (default)
I agree to use On/Off CoD and be billed for extra costs	No (default)

Perform the PowerHA SystemMirror Verify and Synchronize Cluster Configuration process after finishing the previous configuration by running `clmgr sync cluster`.

Showing the ROHA configuration

Example 9-10 shows the output of the `clmgr view report roha` command.

Example 9-10 Output of the clmgr view report roha command

```
Cluster: ITSO_ROHA_cluster of NSC type
      Cluster tunables
          Dynamic LPAR
              Start Resource Groups even if resources are insufficient: '0'
              Adjust Shared Processor Pool size if required: '0'
              Force synchronous release of DLPAR resources: '0'
          On/Off CoD
```

```

I agree to use On/Off CoD and be billed for extra costs: '0'
--> don't use On/Off CoD resource in this case
      Number of activating days for On/Off CoD requests: '30'
      Node: ITSO_S1Node1
          HMCs: 9.3.207.130 9.3.207.133
          Managed system: rar1m3-9117-MMD-1016AAP <--this server is P770D-01
          LPAR: ITSO_S1Node1
              Current profile: 'ITSO_profile'
              Memory (GB):           minimum '8' wanted '32' current '32' maximum
              '96'
              Processing mode: Dedicated
              Processors:           minimum '1' wanted '2' current '2' maximum
              '12'
              ROHA provisioning for resource groups
              No ROHA provisioning.
      Node: ITSO_S2Node1
          HMCs: 9.3.207.130 9.3.207.133
          Managed system: r1r9m1-9117-MMD-1038B9P <--this server is P770D-02
          LPAR: ITSO_S2Node1
              Current profile: 'ITSO_profile'
              Memory (GB):           minimum '8' wanted '32' current '32' maximum
              '96'
              Processing mode: Dedicated
              Processors:           minimum '1' wanted '2' current '2' maximum
              '12'
              ROHA provisioning for resource groups
              No ROHA provisioning.

Hardware Management Console '9.3.207.130' <--this HMC is master
Version: 'V8R8.3.0.1'

Hardware Management Console '9.3.207.133' <--this HMC is backup
Version: 'V8R8.3.0.1'

Managed System 'rar1m3-9117-MMD-1016AAP'
  Hardware resources of the managed system
    Installed:   memory '192' GB       processing units '12.00'
    Configurable: memory '52' GB     processing units '8.00'
    Inactive:    memory '140' GB      processing units '4.00'
    Available:   memory '5' GB       processing units '4.00'
  On/Off CoD
--> this server has enabled On/Off CoD, but we don't use them during RG bring online or
offline scenarios because we want to simulate ONLY Enterprise Pool scenarios. Ignore the
On/Off CoD information.
  On/Off CoD memory
    State: 'Available'
    Available: '9927' GB.days
  On/Off CoD processor
    State: 'Running'
    Available: '9944' CPU.days
    Activated: '4' CPUs <-- this 4CPU is assigned to P770D-01 manually
to simulate four free processor resource
    Left: '20' CPU.days
    Yes: 'DEC_2CEC'
  Enterprise pool
    Yes: 'DEC_2CEC' <-- the enterprise pool name
  Hardware Management Console
    9.3.207.130
    9.3.207.133
  Logical partition 'ITSO_S1Node1'
```

```

Managed System 'r1r9m1-9117-MMD-1038B9P'
    Hardware resources of the managed system
        Installed:     memory '320' GB      processing units '32.00'
        Configurable: memory '95' GB     processing units '16.00'
        Inactive:     memory '225' GB     processing units '16.00'
        Available:    memory '10' GB      processing units '4.00'
    On/Off CoD
--> this server has enabled On/Off CoD, but we don't use them during RG bring online or
offline because we want to simulate ONLY Enterprise Pool exist scenarios.
    On/Off CoD memory
        State: 'Available'
        Available: '9889' GB.days
    On/Off CoD processor
        State: 'Available'
        Available: '9976' CPU.days
        Yes: 'DEC_2CEC'
    Enterprise pool
        Yes: 'DEC_2CEC'
    Hardware Management Console
        9.3.207.130
        9.3.207.133
    Logical partition 'ITSO_S2Node1'
        This 'ITSO_S2Node1' partition hosts 'ITSO_S2Node1' node of the NSC cluster
        'ITSO_ROHA_cluster'

Enterprise pool 'DEC_2CEC'
--> shows that there is no EPCoD mobile resource that is assigned to any server
    State: 'In compliance'
    Master HMC: 'e16hmc1'
    Backup HMC: 'e16hmc3'
    Enterprise pool memory
        Activated memory: '100' GB
        Available memory: '100' GB
        Unreturned memory: '0' GB
    Enterprise pool processor
        Activated CPUs: '4'
        Available CPUs: '4'
        Unreturned CPUs: '0'
    Used by: 'rar1m3-9117-MMD-1016AAP'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPUs: '0' CPUs
        Unreturned CPUs: '0' CPUs
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPUs: '0' CPUs
        Unreturned CPUs: '0' CPUs

```

9.4.2 Testing Example 1: Setting up one ROHA cluster without On/Off CoD

Based on the cluster configuration in 9.4.1, “Example 1: Setting up a ROHA cluster without On/Off CoD” on page 406, this section introduces several testing scenarios:

- ▶ Bringing two resource groups online
- ▶ Moving a resource group to another node
- ▶ Restarting with the current configuration after the primary node crashes

Bringing two resource groups online

When PowerHA SystemMirror starts the cluster service on the primary node, ITSO_S1Node1, the two RGs are brought online. The procedure that is related to ROHA is described in Figure 9-25.

There are four steps for PowerHA SystemMirror to acquire resources:

1. Query.
2. Compute.
3. Identify.
4. Acquire.

Query step

PowerHA SystemMirror queries the server, the EPCoD, the LPARs, and the current RG information. The data is shown in yellow in Figure 9-25.

Compute step

In this step, PowerHA SystemMirror computes how many resources are added by using DLPAR. It needs 7C and 46 GB. The purple cells show the process in Figure 9-25.

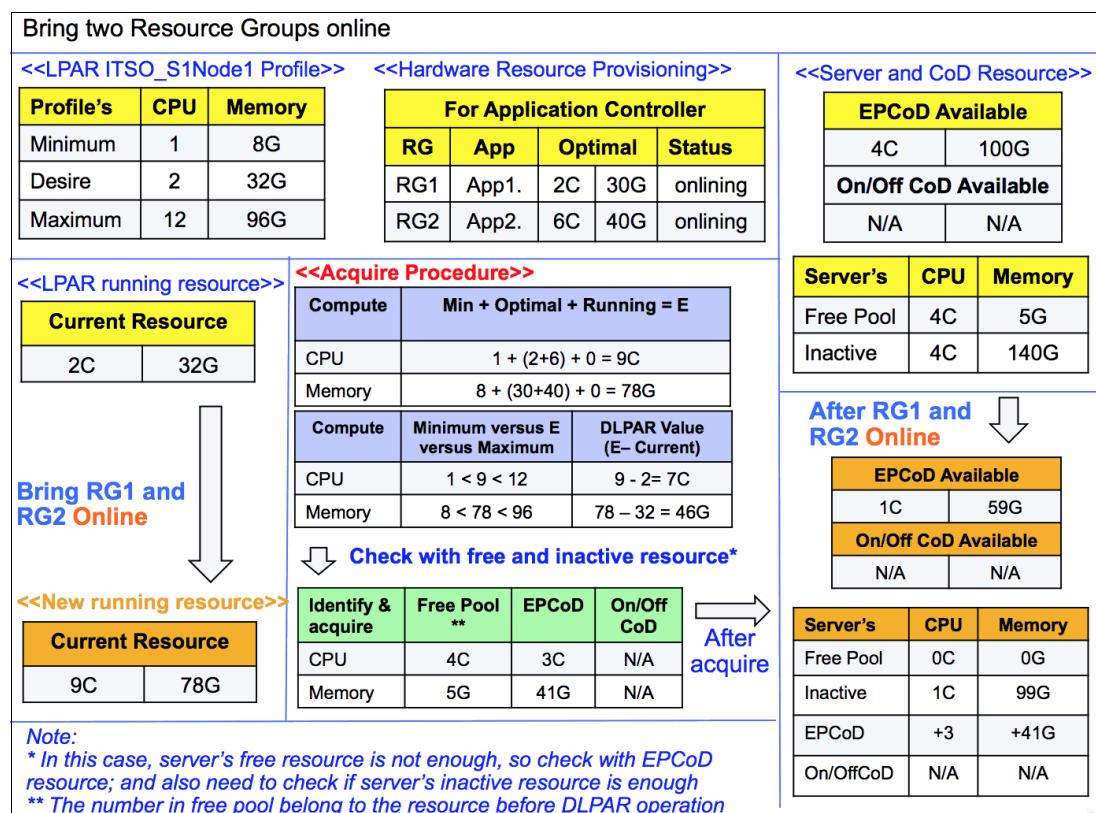


Figure 9-25 Resource acquisition procedure to bring two resource groups online

For example:

- ▶ The expected total CPU number is as follows:
1 (Min) + 2 (RG1 requires) + 6 (RG2 requires) + 0 (running RG requires, there is no running RG) = 9C.
- ▶ Take this value to compare with the LPAR's profile needs less than or equal to the Maximum and more than or equal to the Minimum value.
- ▶ If the requirement is satisfied and takes this value minus the current running CPU, 9 - 2 = 7, you get the CPU number to add through the DLPAR.

Identify and acquire steps

After the compute step, PowerHA SystemMirror identifies how to satisfy the requirements. For CPU, it uses the remaining 4C of this server and 3C from EPCoD. For memory, it uses the remaining 5 GB of this server and 41 GB from EPCoD. The process is shown in the green cells in Figure 9-25 on page 413. For example:

- ▶ There are four CPUs that are available in the server's free pool, so PowerHA SystemMirror reserves them and then needs another three CPUs (7 - 4).
- ▶ There are four mobile CPUs in the EPCoD pool, so PowerHA SystemMirror assigns the three CPUs from EPCoD to this server by using the HMC (by running the **chcodpool** command). Currently, there are seven CPUs in the free pool, so PowerHA SystemMirror assigns all of them to the LPAR (ITSO_S1Node1) by using the DLPAR operation (by using the **chhwres** command).

Note: During this process, PowerHA SystemMirror adds mobile resources from EPCoD to the server's free pool first, and then adds all the free pool's resources to the LPAR by using DLPAR. To describe the process clearly, the free pool means only the available resources of one server before adding the EPCoD resources to it.

The orange cells in Figure 9-25 on page 413 show the results after the resource acquisition, and include the LPAR's running resource, EPCoD, and the server's resource status.

Tracking the hacmp.out log

By reviewing the `hacmp.out` log, we can determine that the acquisition of seven CPUs and 46 GB of memory took 53 seconds, as shown in Example 9-11.

Example 9-11 The `hacmp.out` log shows the resource acquisition process for Example 1

```
# egrep "ROHALOG|Close session|Open session" /var/hacmp/log/hacmp.out
+RG1 RG2:clmanageroh[roha_session_open:162] roha_session_log 'Open session
Open session 22937664 at Sun Nov  8 09:11:39 CST 2015
INFO: acquisition is always synchronous.
==== HACMPProhaparam ODM ====
--> Cluster-wide tunables display
ALWAYS_START_RG      = 0
ADJUST_SPP_SIZE      = 0
FORCE_SYNC_RELEASE   = 0
AGREE_TO_COD_COSTS  = 0
ONOFF_DAYS           = 30
=====
-----+-----+
HMC          |    Version   |
-----+-----+
  9.3.207.130 |    V8R8.3.0.1  |
  9.3.207.133 |    V8R8.3.0.1  |
-----+
```

MANAGED SYSTEM	Memory (GB)	Proc Units	
Name	rar1m3-9117-MMD-1016AAP	Operating	--> Server name
State		/	
Region Size	0.25		
VP/PU Ratio	/	0.05	
Installed	192.00	12.00	
Configurable	52.00	8.00	
Reserved	5.00	/	
Available	5.00	4.00	
Free (computed)	5.00	4.00	--> Free pool resource
LPAR (dedicated)	Memory (GB)	CPU(s)	
Name		ITSO_S1Node1	
State		Running	
Minimum	8.00	1	
Wanted	32.00	2	
Assigned	32.00	2	
Maximum	96.00	12	
ENTERPRISE POOL	Memory (GB)	CPUs	
Name		DEC_2CEC	--> Enterprise Pool Name
State		In compliance	
Master HMC		e16hmc1	
Backup HMC		e16hmc3	
Available	100.00	4	--> Available resource
Unreturned (MS)	0.00	0	
Mobile (MS)	0.00	0	
Inactive (MS)	140.00	4	--> Maximum number to add
TRIAL CoD	Memory (GB)	CPUs	
State	Not Running	Not Running	
Activated	0.00	0	
Days left	0	0	
Hours left	0	0	
ONOFF CoD	Memory (GB)	CPUs	
State	Available	Running	
Activated	0.00	4	--> ignore it
Unreturned	0.00	0	
Available	140.00	4	
Days available	9927	9944	
Days left	0	20	
Hours left	0	2	
OTHER	Memory (GB)	CPUs	
LPAR (dedicated)		ITSO_S2Node1	
State		Running	
ID		13	

Uuid	78E8427B-B157-494A-8711-7B8		
Minimum	8.00	1	
Assigned	32.00	2	

MANAGED SYSTEM	r1r9m1-9117-MMD-1038B9P		
State	Operating		

ENTERPRISE POOL	DEC_2CEC		
Mobile (MS)	0.00	0	

OPTIMAL APPS	Use Wanted	Memory (GB)	CPU	PUs/VPs
App1Controller	0	30.00	2	0.00/0
App2Controller	0	40.00	6	0.00/0
Total	0	70.00	8	0.00/0

===== HACMPdynresop ODM =====

```

TIMESTAMP          = Sun Nov 8 09:11:43 CST 2015
STATE              = start_acquire
MODE               = sync
APPLICATIONS      = App1Controller App2Controller
RUNNING_APPS       = 0
PARTITION          = ITSO_S1Node1
MANAGED_SYSTEM     = rar1m3-9117-MMD-1016AAP
ENTERPRISE_POOL    = DEC_2CEC
PREFERRED_HMC_LIST = 9.3.207.130 9.3.207.133
OTHER_LPAR         = ITSO_S2Node1
INIT_SPP_SIZE_MAX = 0
INIT_DLPAR_MEM    = 32.00
INIT_DLPAR_PROCS  = 2
INIT_DLPAR_PROC_UNITS = 0
INIT_CODPOOL_MEM  = 0.00
INIT_CODPOOL_CPU   = 0
INIT_ONOFF_MEM    = 0.00
INIT_ONOFF_MEM_DAYS = 0
INIT_ONOFF_CPU     = 4
INIT_ONOFF_CPU_DAYS = 20
SPP_SIZE_MAX      = 0
DLPAR_MEM          = 0
DLPAR_PROCS        = 0
DLPAR_PROC_UNITS   = 0
CODPOOL_MEM        = 0
CODPOOL_CPU         = 0
ONOFF_MEM          = 0
ONOFF_MEM_DAYS    = 0
ONOFF_CPU           = 0
ONOFF_CPU_DAYS     = 0

```


===== Compute ROHA Memory =====

--> compute memory process

```

minimal + optimal + running = total <=> current <=> maximum
8.00 + 70.00 + 0.00 = 78.00 <=> 32.00 <=> 96.00 : => 46.00 GB
===== End =====
===== Compute ROHA CPUs =====
--> compute CPU process
minimal + optimal + running = total <=> current <=> maximum
1 + 8 + 0 = 9 <=> 2 <=> 12 : => 7 CPUs
===== End =====

```

```

===== Identify ROHA Memory ====
--> identify memory process
Remaining available memory for partition:      5.00 GB
Total Enterprise Pool memory to allocate:    41.00 GB
Total Enterprise Pool memory to yank:        0.00 GB
Total On/Off CoD memory to activate:         0.00 GB for 0 days
Total DLPAR memory to acquire:                46.00 GB
===== End =====
== Identify ROHA Processor ==
--> identify CPU process
Remaining available PUs for partition:        4.00 Processing Units
Total Enterprise Pool CPUs to allocate:     3.00 CPUs
Total Enterprise Pool CPUs to yank:          0.00 CPUs
Total On/Off CoD CPUs to activate:           0.00 CPUs for 0 days
Total DLPAR CPUs to acquire:                 7.00 CPUs
===== End =====
--> assign EPCoD resource to server
clhmccmd: 41.00 GB of Enterprise Pool CoD have been allocated.
clhmccmd: 3 CPUs of Enterprise Pool CoD have been allocated.
--> assign all resource to LPAR
clhmccmd: 46.00 GB of DLPAR resources have been acquired.
clhmccmd: 7 VPs or CPUs and 0.00 PUs of DLPAR resources have been acquired.
The following resources were acquired for application controllers App1Controller
App2Controller.
DLPAR memory: 46.00 GB      On/Off CoD memory: 0.00 GB      Enterprise Pool memory: 41.00
GB.
DLPAR processor: 7.00 CPUs      On/Off CoD processor: 0.00 CPUs      Enterprise Pool
processor: 3.00 CPUs
INFO: received rc=0.
Success on 1 attempt.
===== HACMPdynresop ODM ====
TIMESTAMP          = Sun Nov 8 09:12:31 CST 2015
STATE              = end_acquire
MODE               = 0
APPLICATIONS       = 0
RUNNING_APPS       = 0
PARTITION          = 0
MANAGED_SYSTEM     = 0
ENTERPRISE_POOL    = 0
PREFERRED_HMC_LIST = 0
OTHER_LPAR          = 0
INIT_SPP_SIZE_MAX  = 0
INIT_DLPAR_MEM     = 0
INIT_DLPAR_PROCS   = 0
INIT_DLPAR_PROC_UNITS = 0
INIT_CODPOOL_MEM   = 0
INIT_CODPOOL_CPU    = 0
INIT_ONOFF_MEM     = 0
INIT_ONOFF_MEM_DAYS = 0
INIT_ONOFF_CPU      = 0
INIT_ONOFF_CPU_DAYS = 0
SPP_SIZE_MAX       = 0
DLPAR_MEM          = 46
DLPAR_PROCS        = 7
DLPAR_PROC_UNITS   = 0
CODPOOL_MEM        = 41
CODPOOL_CPU         = 3
ONOFF_MEM          = 0
ONOFF_MEM_DAYS     = 0
ONOFF_CPU           = 0

```

```
ONOFF_CPU_DAYS      = 0
=====
Session_close:313] roha_session_log 'Close session 22937664 at Sun Nov  8 09:12:32 CST
2015'
```

Important: The contents of the HACMPdynresop Object Data Manager (ODM) changed in PowerHA SystemMirror 7.2.1. Although the exact form changed, the idea of persisting values into HACMPdynresop was kept, so the contents of information that is persisted into HACMPdynresop is subject to change depending on the PowerHA SystemMirror version.

ROHA report update

The **c1mgr view report roha** command output that is shown in Example 9-12 shows updates about the resources of P770D-01 and the Enterprise Pool.

Example 9-12 The update in the ROHA report shows the resource acquisition process for Example 1

```
# c1mgr view report roha
...
Managed System 'rar1m3-9117-MMD-1016AAP' --> the P770D-01 server
  Hardware resources of the managed system
    Installed:   memory '192' GB      processing units '12.00'
    Configurable: memory '93' GB     processing units '11.00'
    Inactive:    memory '99' GB      processing units '1.00'
    Available:   memory '0' GB       processing units '0.00'
...
Enterprise pool 'DEC_2CEC'
  State: 'In compliance'
  Master HMC: 'e16hmc1'
  Backup HMC: 'e16hmc3'
  Enterprise pool memory
    Activated memory: '100' GB
    Available memory: '59' GB
    Unreturned memory: '0' GB
  Enterprise pool processor
    Activated CPUs: '4'
    Available CPUs: '1'
    Unreturned CPUs: '0'
  Used by: 'rar1m3-9117-MMD-1016AAP'
    Activated memory: '41' GB
    Unreturned memory: '0' GB
    Activated CPUs: '3' CPUs
    Unreturned CPUs: '0' CPUs
  Used by: 'r1r9m1-9117-MMD-1038B9P'
    Activated memory: '0' GB
    Unreturned memory: '0' GB
    Activated CPUs: '0' CPUs
    Unreturned CPUs: '0' CPUs
```

Testing summary

The total time to bring the two RGs online is 68 seconds (from 09:11:27 to 09:12:35), which includes the resource acquisition time, as shown in Example 9-13.

Example 9-13 The hacmp.out log shows the total time

```
Nov  8 09:11:27 EVENT START: node_up ITSO_S1Node1
Nov  8 09:11:31 EVENT COMPLETED: node_up ITSO_S1Node1 0
Nov  8 09:11:33 EVENT START: rg_move_fence ITSO_S1Node1 2
Nov  8 09:11:33 EVENT COMPLETED: rg_move_fence ITSO_S1Node1 2 0
```

```

Nov  8 09:11:33 EVENT START: rg_move_acquire ITSO_S1Node1 2
Nov  8 09:11:33 EVENT START: rg_move ITSO_S1Node1 2 ACQUIRE
Nov  8 09:11:34 EVENT START: acquire_service_addr
Nov  8 09:11:34 EVENT START: acquire_aconn_service en0 net_ether_01
Nov  8 09:11:34 EVENT COMPLETED: acquire_aconn_service en0 net_ether_01 0
Nov  8 09:11:35 EVENT START: acquire_aconn_service en0 net_ether_01
Nov  8 09:11:35 EVENT COMPLETED: acquire_aconn_service en0 net_ether_01 0
Nov  8 09:11:35 EVENT COMPLETED: acquire_service_addr 0
Nov  8 09:11:39 EVENT COMPLETED: rg_move ITSO_S1Node1 2 ACQUIRE 0
Nov  8 09:11:39 EVENT COMPLETED: rg_move_acquire ITSO_S1Node1 2 0
Nov  8 09:11:39 EVENT START: rg_move_complete ITSO_S1Node1 2
Nov  8 09:12:32 EVENT START: start_server App1Controller
Nov  8 09:12:32 EVENT START: start_server App2Controller
Nov  8 09:12:32 EVENT COMPLETED: start_server App1Controller 0
Nov  8 09:12:32 EVENT COMPLETED: start_server App2Controller 0
Nov  8 09:12:33 EVENT COMPLETED: rg_move_complete ITSO_S1Node1 2 0
Nov  8 09:12:35 EVENT START: node_up_complete ITSO_S1Node1
Nov  8 09:12:35 EVENT COMPLETED: node_up_complete ITSO_S1Node1 0

```

Moving a resource group to another node

There are two RGs that are running on the primary node (ITSO_S1Node1). Now, we want to move one RG from this node to the standby node (ITSO_S2Node).

In this case, we split this move into two parts: One is the RG offline at the primary node, and the other is the RG online at the standby node.

Resource group offline primary node ITSO_S1Node1

Figure 9-26 describes the offline procedure at the primary node.

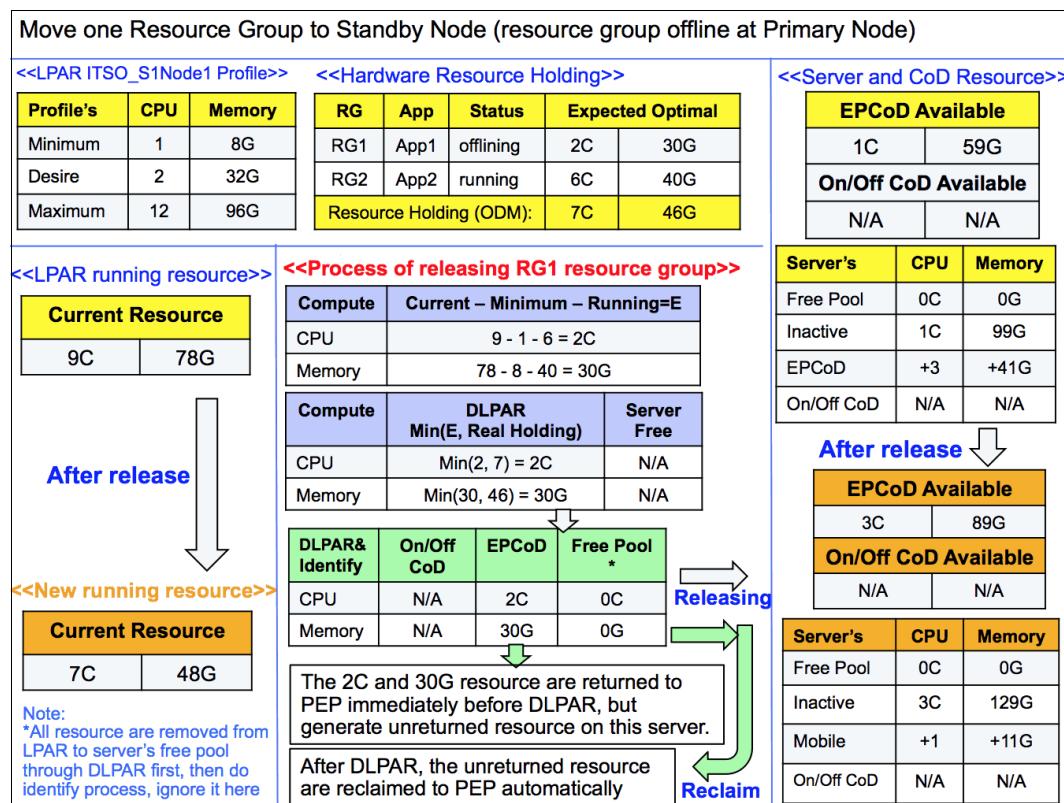


Figure 9-26 Resource group offline procedure at the primary node during the resource group move

The following sections describe the offline procedure.

Query step

PowerHA SystemMirror queries the server, EPCoD, the LPARs, and the current RG information. The data is shown in the yellow cells in Figure 9-26 on page 419.

Compute step

In this step, PowerHA SystemMirror computes how many resources must be removed by using the DLPAR. PowerHA SystemMirror needs 2C and 30 GB. The purple cells show the process, as shown in Figure 9-26 on page 419:

- ▶ In this case, RG1 is released and RG2 is still running. PowerHA calculates how many resources it can release based on whether RG2 has enough resources to run. So, the formula is:
$$9 \text{ (current running)} - 1 \text{ (Min)} - 6 \text{ (RG2 still running)} = 2\text{C}$$

Two CPUs can be released.
- ▶ PowerHA accounts for the fact that sometimes you can adjust your current running resources by using a manual DLPAR operation. For example, you add some resources to satisfy another application that was not started with PowerHA. To avoid removing this resource, PowerHA must check how many resources it allocated before.

The total number is those resources that PowerHA freezes so that the number is not greater than what was allocated before.

So in this case, PowerHA takes the value in the compute step to compare with the real resources this LPAR allocated before. This value is stored in one ODM object database (HACMPdryresop), and the value is 7. PowerHA SystemMirror selects the small one.

Identify and release step

PowerHA SystemMirror identifies how many resources must be released to EPCoD and then releases them to EPCoD asynchronously even though the resources are still in use. This process generates unreturned resources temporarily.

Figure 9-27 shows the dialog boxes that are displayed on the HMC.

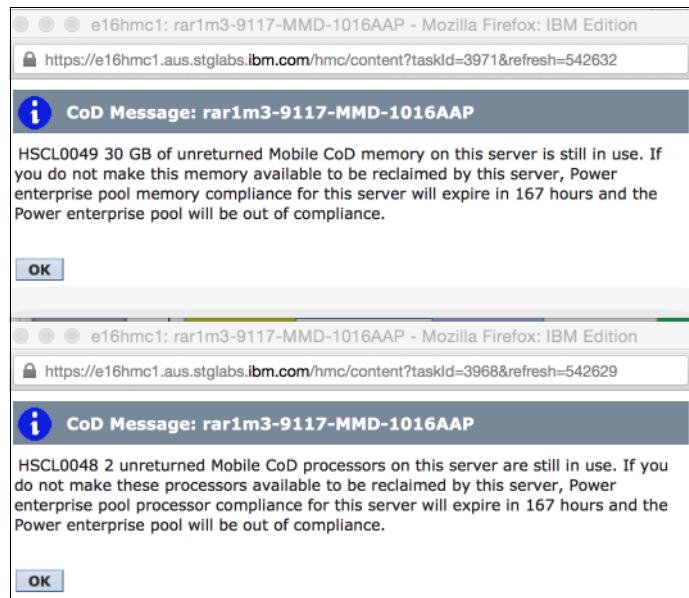


Figure 9-27 HMC message shows that there are unreturned resources that are generated

The unreturned resources can be viewed by using the **clmgr view report roha** command on any of the cluster nodes, as shown in Example 9-14.

Example 9-14 Showing unreturned resources from the AIX command-line interface

```
# clmgr view report roha
...
Enterprise pool 'DEC_2CEC'
    State: 'Approaching out of compliance (within server grace period)'
    Master HMC: 'e16hmc1'
    Backup HMC: 'e16hmc3'
    Enterprise pool memory
        Activated memory: '100' GB
        Available memory: '89' GB -->the 30 GB has been changed to EPCoD available
status
        Unreturned memory: '30' GB -->the 30 GB is marked 'unreturned'
    Enterprise pool processor
        Activated CPUs: '4'
        Available CPUs: '3' --> the 2CPU has been changed to EPCoD available status
        Unreturned CPUs: '2' --> the 2CPU is marked 'unreturned'
    Used by: 'rar1m3-9117-MMD-1016AAP' -->show unreturned resource from server's view
        Activated memory: '11' GB
        Unreturned memory: '30' GB
        Activated CPUs: '1' CPUs
        Unreturned CPUs: '2' CPUs
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPUs: '0' CPUs
        Unreturned CPUs: '0' CPUs
```

From the HMC command-line interface (CLI), you can see the unreturned resources that are generated, as shown in Example 9-15.

Example 9-15 Showing the unreturned resources and the status from the HMC CLI

```
hsroot@e16hmc1:~> lscodpool -p DEC_2CEC --level sys
name=rar1m3-9117-MMD-1016AAP,mtms=9117-MMD*1016AAP,mobile_procs=1,non_mobile_procs=8,unreturned_mobile_procs=2,inactive_procs=1,installed_procs=12,mobile_mem=11264,non_mobile_mem=53248,unreturned_mobile_mem=30720,inactive_mem=101376,installed_mem=196608
name=r1r9m1-9117-MMD-1038B9P,mtms=9117-MMD*1038B9P,mobile_procs=0,non_mobile_procs=16,unreturned_mobile_procs=0,inactive_procs=16,installed_procs=32,mobile_mem=0,non_mobile_mem=97280,unreturned_mobile_mem=0,inactive_mem=230400,installed_mem=327680

hsroot@e16hmc1:~> lscodpool -p DEC_2CEC --level pool
name=DEC_2CEC,ID=026F,state=Approaching out of compliance (within server grace period),sequence_num=41,master_mc_name=e16hmc1,master_mc_mtms=7042-CR5*06K0040,backup_master_mc_name=e16hmc3,backup_master_mc_mtms=7042-CR5*06K0036,mobile_procs=4,avail_mobile_procs=3,unreturned_mobile_procs=2,mobile_mem=102400,avail_mobile_mem=91136,unreturned_mobile_mem=30720
```

Meanwhile, PowerHA SystemMirror triggers one asynchronous process to do the DLPAR remove operation, and it removes 2C and 30 GB of resources from the LPAR into the server's free pool. The log is written in the `/var/hacmp/log/async_release.log` file.

When the DLPAR operation completes, the unreturned resources are reclaimed immediately, and some messages are shown on the HMC in Figure 9-28. The Enterprise Pool's status is changed back to In compliance.

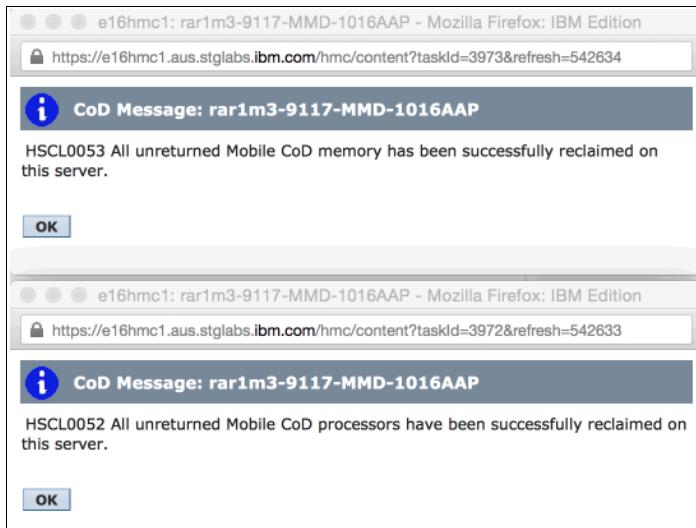


Figure 9-28 The unreturned resources are reclaimed after the DLPAR operation

You can see the changes from the HMC CLI, as shown in Example 9-16.

Example 9-16 Showing the unreturned resources that are reclaimed from the HMC CLI

```
hscroot@e16hmc1:~> lscodpool -p DEC_2CEC --level sys
name=rar1m3-9117-MMD-1016AAP,mtms=9117-MMD*1016AAP,mobile_procs=1,non_mobile_procs
=8,unreturned_mobile_procs=0,inactive_procs=3,installed_procs=12,mobile_mem=11264,
non_mobile_mem=53248,unreturned_mobile_mem=0,inactive_mem=132096,installed_mem=196
608
name=r1r9m1-9117-MMD-1038B9P,mtms=9117-MMD*1038B9P,mobile_procs=0,non_mobile_procs
=16,unreturned_mobile_procs=0,inactive_procs=16,installed_procs=32,mobile_mem=0,no
n_mobile_mem=97280,unreturned_mobile_mem=0,inactive_mem=230400,installed_mem=32768
0
hscroot@e16hmc1:~> lscodpool -p DEC_2CEC --level pool
name=DEC_2CEC, ID=026F, state=In compliance, sequence_num=41, master_mc_name=e16hmc1,
master_mc_mtms=7042-CR5*06K0040, backup_master_mc_name=e16hmc3, backup_master_mc_mt
s=7042-CR5*06K0036, mobile_procs=4,avail_mobile_procs=3, unreturned_mobile_procs=0,m
obile_mem=102400,avail_mobile_mem=91136, unreturned_mobile_mem=0
```

Note: The Approaching out of compliance status is a normal status in the Enterprise Pool, and it is useful when you need extra resources temporarily. The PowerHA SystemMirror RG takeover scenario is one of those cases.

Log information in the hacmp.out file

The hacmp.out log file records the process of releasing the RG, as shown in Example 9-17.

Example 9-17 The hacmp.out log file information about the resource group offline process

```
#egrep "ROHALOG|Close session|Open session" /var/hacmp/log/hacmp.out
...
===== Compute ROHA Memory =====
minimum + running = total <=> current <=> optimal <=> saved
```

```

8.00 + 40.00 = 48.00 <=> 78.00 <=> 30.00 <=> 46.00 : => 30.00 GB
===== End =====
==== Compute ROHA CPUs ====
minimal + running = total <=> current <=> optimal <=> saved
1 +       6 =      7 <=>     9 <=>     2 <=>     7 : =>           2 CPUs
===== End =====
==== Identify ROHA Memory ====
Total Enterprise Pool memory to return back:          30.00 GB
Total On/Off CoD memory to de-activate:                0.00 GB
Total DLPAR memory to release:                         30.00 GB
===== End =====
== Identify ROHA Processor ==
Total Enterprise Pool CPUs to return back:           2.00 CPUs
Total On/Off CoD CPUs to de-activate:                 0.00 CPUs
Total DLPAR CPUs to release:                          2.00 CPUs
===== End =====
clhmccmd: 30.00 GB of Enterprise Pool CoD have been returned.
clhmccmd: 2 CPUs of Enterprise Pool CoD have been returned.
The following resources were released for application controllers App1Controller.
DLPAR memory: 30.00 GB      On/Off CoD memory: 0.00 GB      Enterprise Pool
memory: 30.00 GB.
DLPAR processor: 2.00 CPUs      On/Off CoD processor: 0.00 CPUs
Enterprise Pool processor: 2.00 CPU(s)
Close session 22937664 at Sun Nov  8
09:12:32 CST 2015
..

```

During the release process, the deallocation order is EPCoD, and then the local server's free pool. Because EPCoD is shared between different servers, the standby node running on other servers always needs this resource to bring the RG online in a takeover scenario.

Resources online at the standby node (ITSO_S2Node1)

In this case, the RG that is online on the standby node does not need to wait for the DLPAR to complete on the primary node because it is an asynchronous process. In this process, PowerHA SystemMirror acquires a corresponding resource for the RG.

Note: Before the process of acquiring resources started, the resources (2C and 30 GB) were available in the Enterprise Pool, so they could be used by the standby node.

Figure 9-29 illustrates the resource acquisition process on the standby node ITSO_S2Node1.

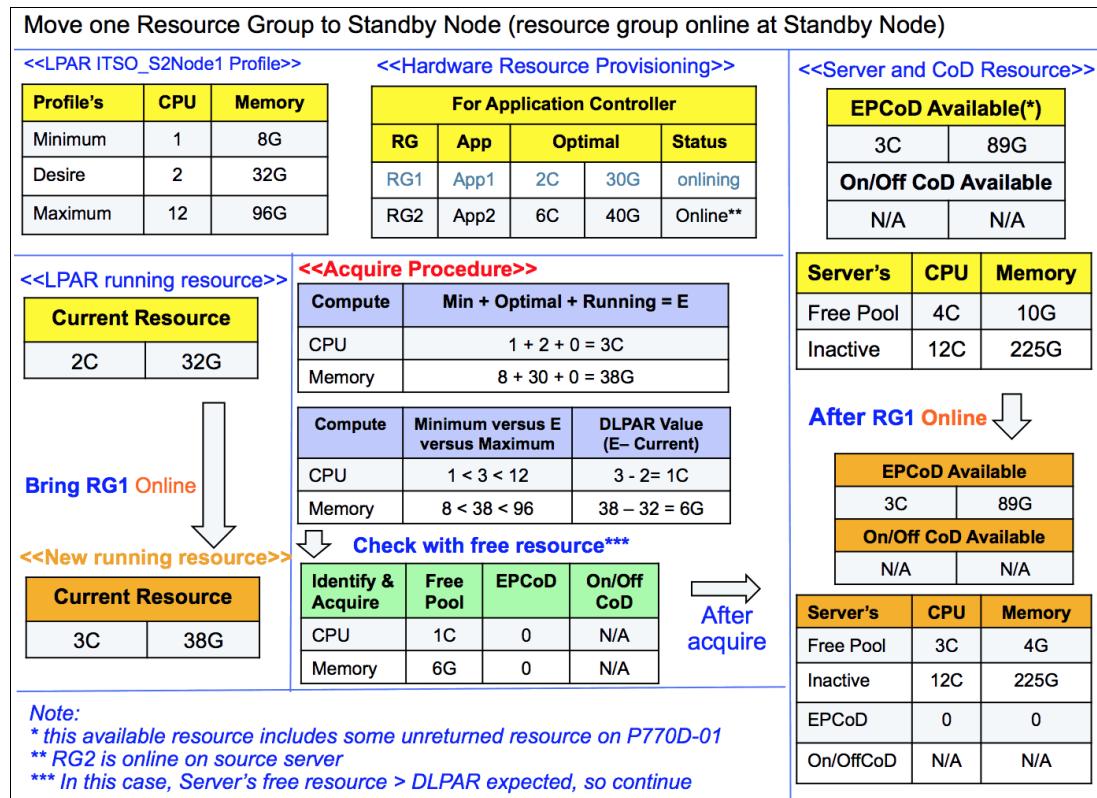


Figure 9-29 The acquisition process on the standby node

This acquisition process differs from the scenario that is described in “Bringing two resource groups online” on page 413. The resources that must be added to the LPAR are one core and 6 GB, which the system’s free pool can satisfy. The server does not need to acquire any resources from EPCoD.

Testing scenario summary

The total time of this RG migration was 28 seconds, from 10:53:15 to 10:53:43.

Removing the resources (2C and 30 GB) from the LPAR to a free pool on the primary node took 257 seconds (10:52:51 - 10:57:08). However, there is no real concern with this time because it is an asynchronous process.

Example 9-18 shows the hacmp.out information about ITSO_S1Node1.

Example 9-18 The key timestamp in hacmp.out on the primary node (ITSO_S1Node1)

```
# egrep "EVENT START|EVENT COMPLETED" hacmp.out
Nov 8 10:52:27 EVENT START: external_resource_state_change ITSO_S2Node1
Nov 8 10:52:27 EVENT COMPLETED: external_resource_state_change ITSO_S2Node1 0
Nov 8 10:52:27 EVENT START: rg_move_release ITSO_S1Node1 1
Nov 8 10:52:27 EVENT START: rg_move ITSO_S1Node1 1 RELEASE
Nov 8 10:52:27 EVENT START: stop_server App1Controller
Nov 8 10:52:28 EVENT COMPLETED: stop_server App1Controller 0
Nov 8 10:52:53 EVENT START: release_service_addr
Nov 8 10:52:54 EVENT COMPLETED: release_service_addr 0
Nov 8 10:52:56 EVENT COMPLETED: rg_move ITSO_S1Node1 1 RELEASE 0
Nov 8 10:52:56 EVENT COMPLETED: rg_move_release ITSO_S1Node1 1 0
```

```

Nov  8 10:52:58 EVENT START: rg_move_fence ITSO_S1Node1 1
Nov  8 10:52:58 EVENT COMPLETED: rg_move_fence ITSO_S1Node1 1 0
Nov  8 10:53:00 EVENT START: rg_move_fence ITSO_S1Node1 1
Nov  8 10:53:00 EVENT COMPLETED: rg_move_fence ITSO_S1Node1 1 0
Nov  8 10:53:00 EVENT START: rg_move_acquire ITSO_S1Node1 1
Nov  8 10:53:00 EVENT START: rg_move ITSO_S1Node1 1 ACQUIRE
Nov  8 10:53:00 EVENT COMPLETED: rg_move ITSO_S1Node1 1 ACQUIRE 0
Nov  8 10:53:00 EVENT COMPLETED: rg_move_acquire ITSO_S1Node1 1 0
Nov  8 10:53:18 EVENT START: rg_move_complete ITSO_S1Node1 1
Nov  8 10:53:19 EVENT COMPLETED: rg_move_complete ITSO_S1Node1 1 0
Nov  8 10:53:50 EVENT START: external_resource_state_change_complete ITSO_S2Node1
Nov  8 10:53:50 EVENT COMPLETED: external_resource_state_change_complete ITSO_S2Node1 0

```

Example 9-19 shows the `async_release.log` file on ITSO_S2Node1.

Example 9-19 The `asyn_release.log` records the DLPAR operation

```

# egrep "Sun Nov| eval LC_ALL=C ssh " async_release.log
Sun Nov  8 10:52:51 CST 2015
+RG1:c1hmccmd[c1hmceexec:3624] : Start ssh command at Sun Nov 8 10:52:56 CST 2015
+RG1:c1hmccmd[c1hmceexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no

$hscroot@9.3.207.130 \'lssyscfg -r sys -m 9117-MMD*1016AAP -F name 2>&1\''
+RG1:c1hmccmd[c1hmceexec:3627] : Return from ssh command at Sun Nov 8 10:52:56 CST 2015
+RG1:c1hmccmd[c1hmceexec:3624] : Start ssh command at Sun Nov 8 10:52:56 CST 2015
+RG1:c1hmccmd[c1hmceexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no

$hscroot@9.3.207.130 \'chhwres -m rar1m3-9117-MMD-1016AAP -p ITSO_S1Node1 -r mem -o r -q
10240 -w 30 2>&1\''
+RG1:c1hmccmd[c1hmceexec:3627] : Return from ssh command at Sun Nov 8 10:54:19 CST 2015
+RG1:c1hmccmd[c1hmceexec:3624] : Start ssh command at Sun Nov 8 10:54:19 CST 2015
+RG1:c1hmccmd[c1hmceexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no

$hscroot@9.3.207.130 \'chhwres -m rar1m3-9117-MMD-1016AAP -p ITSO_S1Node1 -r mem -o r -q
10240 -w 30 2>&1\''
+RG1:c1hmccmd[c1hmceexec:3627] : Return from ssh command at Sun Nov 8 10:55:32 CST 2015
+RG1:c1hmccmd[c1hmceexec:3624] : Start ssh command at Sun Nov 8 10:55:32 CST 2015
+RG1:c1hmccmd[c1hmceexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no

$hscroot@9.3.207.130 \'chhwres -m rar1m3-9117-MMD-1016AAP -p ITSO_S1Node1 -r mem -o r -q
10240 -w 30 2>&1\''
+RG1:c1hmccmd[c1hmceexec:3627] : Return from ssh command at Sun Nov 8 10:56:40 CST 2015
+RG1:c1hmccmd[c1hmceexec:3624] : Start ssh command at Sun Nov 8 10:56:40 CST 2015
+RG1:c1hmccmd[c1hmceexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no

$hscroot@9.3.207.130 \'chhwres -m rar1m3-9117-MMD-1016AAP -p ITSO_S1Node1 -r proc -o r
--procs 2 -w 30 2>&1\''
+RG1:c1hmccmd[c1hmceexec:3627] : Return from ssh command at Sun Nov 8 10:57:08 CST 2015
Sun Nov  8 10:57:08 CST 2015

```

Example 9-20 shows the hacmp.out information about ITSO_S2Node1.

Example 9-20 The key timestamp in hacmp.out on the standby node (ITSO_S1Node1)

```
#egrep "EVENT START|EVENT COMPLETED" hacmp.out
Nov 8 10:52:24 EVENT START: rg_move_release ITSO_S1Node1 1
Nov 8 10:52:24 EVENT START: rg_move ITSO_S1Node1 1 RELEASE
Nov 8 10:52:25 EVENT COMPLETED: rg_move ITSO_S1Node1 1 RELEASE 0
Nov 8 10:52:25 EVENT COMPLETED: rg_move_release ITSO_S1Node1 1 0
Nov 8 10:52:55 EVENT START: rg_move_fence ITSO_S1Node1 1
Nov 8 10:52:55 EVENT COMPLETED: rg_move_fence ITSO_S1Node1 1 0
Nov 8 10:52:57 EVENT START: rg_move_fence ITSO_S1Node1 1
Nov 8 10:52:57 EVENT COMPLETED: rg_move_fence ITSO_S1Node1 1 0
Nov 8 10:52:57 EVENT START: rg_move_acquire ITSO_S1Node1 1
Nov 8 10:52:57 EVENT START: rg_move ITSO_S1Node1 1 ACQUIRE
Nov 8 10:52:57 EVENT START: acquire_takeover_addr
Nov 8 10:52:58 EVENT COMPLETED: acquire_takeover_addr 0
Nov 8 10:53:15 EVENT COMPLETED: rg_move ITSO_S1Node1 1 ACQUIRE 0
Nov 8 10:53:15 EVENT COMPLETED: rg_move_acquire ITSO_S1Node1 1 0
Nov 8 10:53:15 EVENT START: rg_move_complete ITSO_S1Node1 1
Nov 8 10:53:43 EVENT START: start_server App1Controller
Nov 8 10:53:43 EVENT COMPLETED: start_server App1Controller 0
Nov 8 10:53:45 EVENT COMPLETED: rg_move_complete ITSO_S1Node1 1 0
Nov 8 10:53:47 EVENT START: external_resource_state_change_complete ITSO_S2Node1
Nov 8 10:53:47 EVENT COMPLETED: external_resource_state_change_complete ITSO_S2Node1 0
```

Restarting with the current configuration after the primary node crashes

This case introduces the Automatic Release After a Failure (ARAF) process. We simulate a primary node that failed immediately. We do not describe how the RG is online on standby node; we describe only what PowerHA SystemMirror does after the primary node restarts. Assume that we activate this node with the current configuration, which means that this LPAR can still hold the same number of resources as before the crash.

After the primary node restarts, the `/usr/es/sbin/cluster/etc/rc.init` script is triggered by `/etc/inittab` and performs the resource releasing operation. The process is shown in Figure 9-30 on page 427.

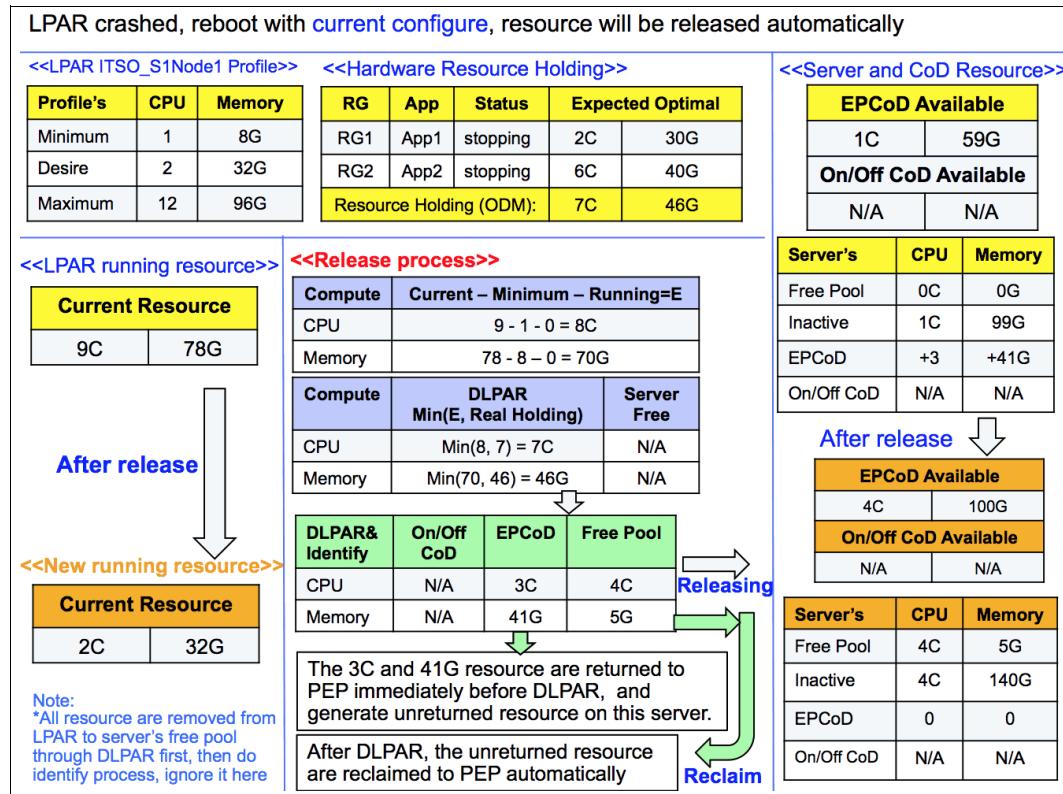


Figure 9-30 Resource release process by using the ARAF process

The process is similar to “Resource group offline primary node ITSO_S1Node1” on page 419. In this process, PowerHA SystemMirror tries to release all the resources that were held by the two RGs before.

Testing summary

If a resource was not released because of a PowerHA SystemMirror service crash or an AIX operating system crash, PowerHA SystemMirror can perform the release operation automatically after the node starts. This operation occurs before you start the PowerHA SystemMirror service by using the `smitty clstart` or the `clmgr start cluster` commands.

9.4.3 Example 2: Setting up one ROHA cluster with On/Off CoD

This section describes how to set up a ROHA cluster with On/Off CoD.

Requirements

- ▶ Two IBM Power 770 D model servers, both in one Power Enterprise Pool. Each server has an On/Off CoD license.
- ▶ One PowerHA SystemMirror cluster with two nodes that are in different servers.
- ▶ The PowerHA SystemMirror cluster manages the server's free resources and EPCoD mobile resources, and On/Off CoD resources to automatically satisfy the application's hardware requirements before it is started.

Hardware topology

Figure 9-31 shows the server and LPAR information for Example 2.

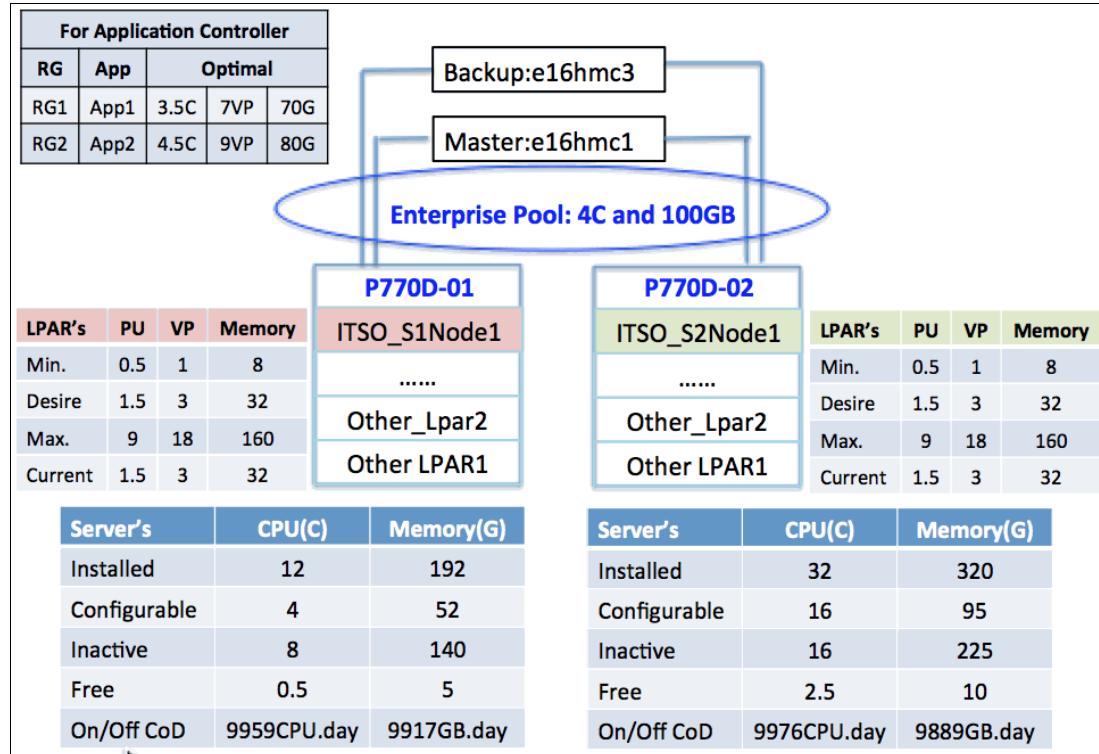


Figure 9-31 Server and LPAR information

The topology includes the following components for configuration:

- ▶ Two Power 770 D model servers that are named P770D-01 and P770D-02.
 - P770D-01 has eight inactive CPUs, 140 GB of inactive memory, 0.5 free CPUs, and 5 GB of available memory.
 - P770D-02 has 16 inactive CPUs, 225 GB of inactive memory, 2.5 free CPUs, and 10 GB of available memory.
 - Each server enabled the On/Off CoD feature.
- ▶ One Power Enterprise Pool with four mobile processors and 100 GB of mobile memory.
- ▶ The PowerHA SystemMirror cluster includes two nodes: ITSO_S1Node1 and ITSO_S2Node1.
- ▶ This topology also includes the profile configuration for each LPAR.

There are two HMCs to manage the EPCoD, which are named e16hmc1 and e16hmc3. Here, e16hmc1 is the master and e16hmc3 is the backup. There are two applications in this cluster and related resource requirements.

Available resources in On/Off CoD

In the example, the resources that we have at the On/Off CoD level are GB.Days or Processor.Days. For example, we can have 600 GB.Days, or 120 Processors.Days in the On/Off CoD pool. The time scope of the activation is determined through a tunable variable that is called Number of activating days for On/Off CoD requests. For more information, see "Change/Show Default Cluster Tunable menu" on page 401.

If the tunable is set to 30, for example, it means that we want to activate the resources for 30 days. So, the tunable allocates 20 GB of memory, and we have 20 GB On/Off CoD only, even if we have 600 GB.Days available.

Cluster configuration

The topology and RG configuration, and HMC configuration are the same, as shown Table 9-13 on page 408.

Hardware resource provisioning for the application controllers

There are two application controllers to add, as shown in Table 9-19 and Table 9-20.

Table 9-19 Configuring AppController1 resources

Items	Value
I agree to use On/Off CoD and be billed for extra costs	Yes
Application Controller Name	AppController1
Use the wanted level from the LPAR profile	No
Optimal number of gigabytes of memory	70
Optimal number of processing units	3.5
Optimal number of virtual processors	7

Table 9-20 Configuring AppController2 resources

Items	Value
I agree to use On/Off CoD and be billed for extra costs	Yes
Application Controller Name	AppController2
Use the wanted level from the LPAR profile	No
Optimal number of gigabytes of memory	80
Optimal number of processing units	4.5
Optimal number of virtual processors	9

Cluster-wide tunables

All the tunables are at the default values, as shown in Table 9-21.

Table 9-21 ROHA cluster tunables

Items	Value
DLPAR Always Start Resource Groups	No (default)
Adjust Shared Processor Pool size if required	No (default)
Force synchronous release of DLPAR resources	No (default)
I agree to use On/Off CoD and be billed for extra costs	Yes
Number of activating days for On/Off CoD requests	30 (default)

This configuration requires that you perform a Verify and Synchronize Cluster Configuration action after changing the previous configuration. Run `c1mgr sync cluster`.

Showing the ROHA configuration

The `clmgr view report roha` command shows the current ROHA data (Example 9-21).

Example 9-21 Showing the ROHA data with the clmgr view report roha command

```
# clmgr view report roha
Cluster: ITSO_ROHA_cluster of NSC type
    Cluster tunables --> Following is the cluster tunables
        Dynamic LPAR
            Start Resource Groups even if resources are insufficient: '0'
            Adjust Shared Processor Pool size if required: '0'
            Force synchronous release of DLPAR resources: '0'
    On/Off CoD
        I agree to use On/Off CoD and be billed for extra costs: '1'
        Number of activating days for On/Off CoD requests: '30'
Node: ITSO_S1Node1 --> Information of ITSO_S1Node1 node
    HMCs: 9.3.207.130 9.3.207.133
    Managed system: rar1m3-9117-MMD-1016AAP
    LPAR: ITSO_S1Node1
        Current profile: 'ITSO_profile'
        Memory (GB): minimum '8' wanted '32' current '32' maximum '160'
        Processing mode: Shared
        Shared processor pool: 'DefaultPool'
        Processing units: minimum '0.5' wanted '1.5' current '1.5' maximum '9.0'
        Virtual processors: minimum '1' wanted '3' current '3' maximum '18'
    ROHA provisioning for resource groups
        No ROHA provisioning.
Node: ITSO_S2Node1 --> Information of ITSO_S2Node1 node
    HMCs: 9.3.207.130 9.3.207.133
    Managed system: r1r9m1-9117-MMD-1038B9P
    LPAR: ITSO_S2Node1
        Current profile: 'ITSO_profile'
        Memory (GB): minimum '8' wanted '32' current '32' maximum '160'
        Processing mode: Shared
        Shared processor pool: 'DefaultPool'
        Processing units: minimum '0.5' wanted '1.5' current '1.5' maximum '9.0'
        Virtual processors: minimum '1' wanted '3' current '3' maximum '18'
    ROHA provisioning for resource groups
        No ROHA provisioning.

Hardware Management Console '9.3.207.130' --> Information of HMCs
    Version: 'V8R8.3.0.1'

Hardware Management Console '9.3.207.133'
    Version: 'V8R8.3.0.1'

Managed System 'rar1m3-9117-MMD-1016AAP' --> Information of P770D-01
    Hardware resources of the managed system
        Installed: memory '192' GB processing units '12.00'
        Configurable: memory '52' GB processing units '4.00'
        Inactive: memory '140' GB processing units '8.00'
        Available: memory '5' GB processing units '0.50'
    On/Off CoD --> Information of On/Off CoD on P770D-01 server
        On/Off CoD memory
            State: 'Available'
            Available: '9907' GB.days
        On/Off CoD processor
            State: 'Available'
            Available: '9959' CPU.days
                Yes: 'DEC_2CEC'
        Enterprise pool
            Yes: 'DEC_2CEC'
    Hardware Management Console
        9.3.207.130
        9.3.207.133
    Shared processor pool 'DefaultPool'
```

```

Logical partition 'ITSO_S1Node1'
    This 'ITSO_S1Node1' partition hosts 'ITSO_S2Node1' node of the NSC cluster
    'ITSO_ROHA_cluster'

Managed System 'r1r9m1-9117-MMD-1038B9P' --> Information of P770D-02
    Hardware resources of the managed system
        Installed: memory '320' GB      processing units '32.00'
        Configurable: memory '95' GB   processing units '16.00'
        Inactive: memory '225' GB     processing units '16.00'
        Available: memory '10' GB     processing units '2.50'
    On/Off CoD --> Information of On/Off CoD on P770D-02 server
        On/Off CoD memory
            State: 'Available'
            Available: '9889' GB.days
        On/Off CoD processor
            State: 'Available'
            Available: '9976' CPU.days
            Yes: 'DEC_2CEC'
    Enterprise pool
        Yes: 'DEC_2CEC'
    Hardware Management Console
        9.3.207.130
        9.3.207.133
    Shared processor pool 'DefaultPool'
    Logical partition 'ITSO_S2Node1'
        This 'ITSO_S2Node1' partition hosts 'ITSO_S2Node1' node of the NSC cluster
        'ITSO_ROHA_cluster'

Enterprise pool 'DEC_2CEC' --> Information of Enterprise Pool
    State: 'In compliance'
    Master HMC: 'e16hmc1'
    Backup HMC: 'e16hmc3'
    Enterprise pool memory
        Activated memory: '100' GB -->Total mobile resource of Pool, does not change during
        resource moving
        Available memory: '100' GB -->Available for assign, changes during resource moving
        Unreturned memory: '0' GB
    Enterprise pool processor
        Activated CPUs: '4'
        Available CPUs: '4'
        Unreturned CPUs: '0'
    Used by: 'rar1m3-9117-MMD-1016AAP'
        Activated memory: '0' GB --> the number that is assigned from EPCoD to server
        Unreturned memory: '0' GB --> the number has been released to EPCoD but not reclaimed,
        need to reclaimed within a period time
        Activated CPUs: '0' CPUs
        Unreturned CPUs: '0' CPUs
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPUs: '0' CPUs
        Unreturned CPUs: '0' CPUs

```

9.4.4 Testing Example 2 with On/Off CoD

Based on the configuration in 9.4.3, “Example 2: Setting up one ROHA cluster with On/Off CoD” on page 427, this section introduces two testing scenarios:

- ▶ Bringing two resource groups online
- ▶ Bringing one resource group offline

Bringing two resource groups online

When PowerHA SystemMirror starts cluster services on the primary node (ITSO_S1Node1), the two RGs go online. The procedure that is related to ROHA is shown in Figure 9-32.

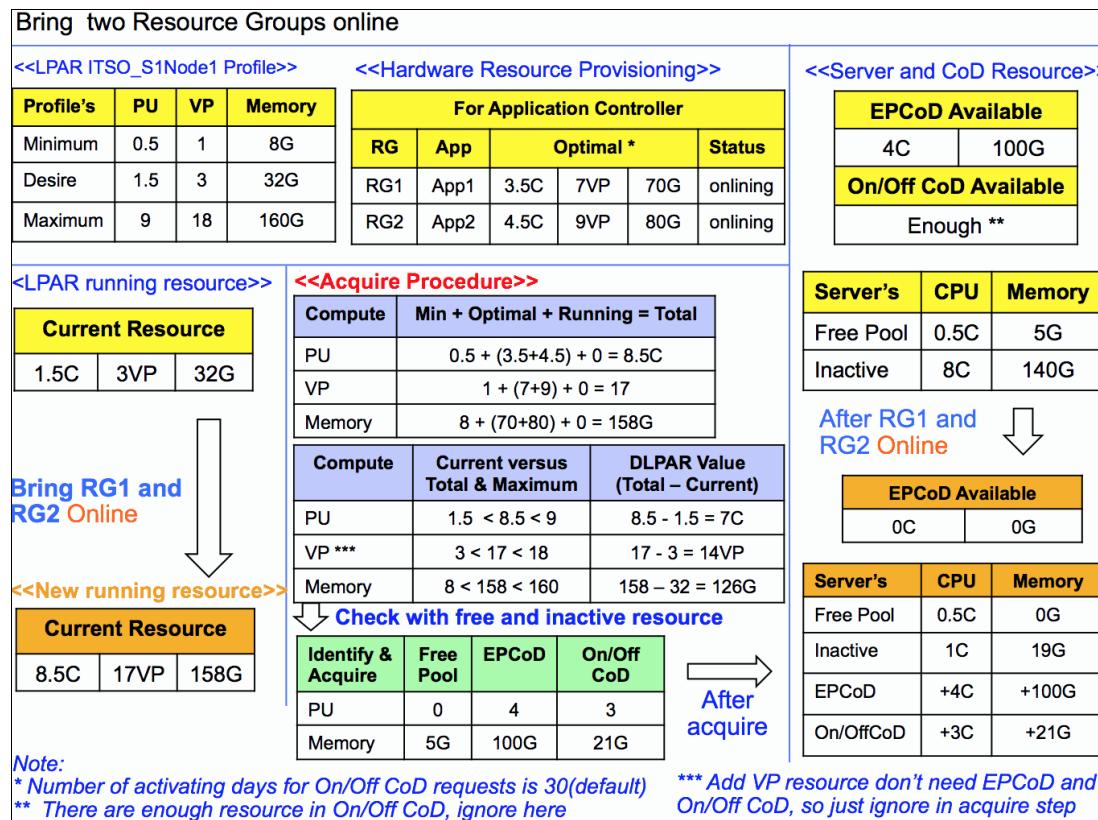


Figure 9-32 The acquisition resource process of Example 2

Query step

PowerHA SystemMirror queries the server, EPCoD, the On/Off CoD, the LPARs, and the current RG information. The data is shown in the yellow cells in Figure 9-32.

For the On/Off CoD resources, we do not display the available resources because there are enough resources in our testing environment:

- ▶ The P770D-01 has 9959 CPU.days and 9917 GB.days.
- ▶ The P770D-02 has 9976 CPU.days and 9889 GB.days.

We display the actual amount that is used.

Compute step

In this step, PowerHA SystemMirror computes how many resources you must add through the DLPAR. PowerHA SystemMirror needs 7C and 126 GB. The purple cells show this process (Figure 9-32 on page 432). We take the CPU resources as follows:

- ▶ The expected total processor unit number is $0.5 \text{ (Min)} + 3.5 \text{ (RG1 requirement)} + 4.5 \text{ (RG2 requirement)} + 0 \text{ (running RG requirement (there is no running RG))} = 8.5\text{C}$.
- ▶ Take this value to compare with the LPAR's profile, which must be less than or equal to the Maximum value and more than or equal to the Minimum value.
- ▶ If this configuration satisfies the requirement, then take this value minus the current running CPU ($8.5 - 1.5 = 7$), which is the number that we want to add to the LPAR through DLPAR.

Identify and acquire step

After the compute step, PowerHA SystemMirror identifies how to satisfy the requirement. For CPU, it gets 4C from EPCoD and 3C from the On/Off CoD. Because the minimum operation unit is 1 for EPCoD and On/Off CoD, even if there is 0.5 CPU in the server's free pool, the requirement is 7, so you leave it in the free pool.

PowerHA SystemMirror gets the remaining 5 GB of this server, all 100 GB from EPCoD, and 21 GB from the On/Off CoD. The process is shown in the green cells in Figure 9-32 on page 432.

Note: During this process, PowerHA SystemMirror adds mobile resources from EPCoD to the server's free pool first, then adds all the free pool's resources to the LPAR through DLPAR. The *free pool* is the available resources of only one server before adding the EPCoD resources to it.

The orange cells in Figure 9-32 on page 432 show the result of this scenario, including the LPAR's running resources, EPCoD, On/Off CoD, and the server's resource status.

Tracking the hacmp.out log

From `hacmp.out`, you know that all the resources (seven CPUs and 126 GB of memory) cost 117 seconds as a synchronous process, as shown in Example 9-22:

22:44:40 → 22:46:37

Example 9-22 The `hacmp.out` log shows the resource acquisition of Example 2

```
===== Compute ROHA Memory =====
minimal + optimal + running = total <=> current <=> maximum
8.00 + 150.00 + 0.00 = 158.00 <=> 32.00 <=> 160.00 : => 126.00 GB
===== End =====
== Compute ROHA PUs/VP(s) ==
minimal + optimal + running = total <=> current <=> maximum
1 + 16 + 0 = 17 <=> 3 <=> 18 : => 14 Virtual Processors
minimal + optimal + running = total <=> current <=> maximum
0.50 + 8.00 + 0.00 = 8.50 <=> 1.50 <=> 9.00 : => 7.00 Processing Units
===== End =====
===== Identify ROHA Memory ====
Remaining available memory for partition: 5.00 GB
Total Enterprise Pool memory to allocate: 100.00 GB
Total Enterprise Pool memory to yank: 0.00 GB
Total On/Off CoD memory to activate: 21.00 GB for 30 days
Total DLPAR memory to acquire: 126.00 GB
===== End =====
== Identify ROHA Processor ==
```

```

Remaining available PUs for partition:           0.50 Processing Units
Total Enterprise Pool CPUs to allocate:        4.00 CPUs
Total Enterprise Pool CPUs to yank:            0.00 CPUs
Total On/Off CoD CPUs to activate:             3.00 CPUs for 30 days
Total DLPAR PUs/VP(s) to acquire:              7.00 Processing Units and 14.00 Virtual
Processors
===== End =====
c1hmccmd: 100.00 GB of Enterprise Pool CoD have been allocated.
c1hmccmd: 4 CPUs of Enterprise Pool CoD have been allocated.
c1hmccmd: 21.00 GB of On/Off CoD resources have been activated for 30 days.
c1hmccmd: 3 CPUs of On/Off CoD resources have been activated for 30 days.
c1hmccmd: 126.00 GB of DLPAR resources have been acquired.
c1hmccmd: 14 VPs or CPUs and 7.00 PUs of DLPAR resources have been acquired.
The following resources were acquired for application controllers App1Controller
App2Controller.
DLPAR memory: 126.00 GB          On/Off CoD memory: 21.00 GB      Enterprise Pool
memory: 100.00 GB.
DLPAR processor: 7.00 PU/14.00 VP   On/Off CoD processor: 3.00 CPUs      Enterprise Pool
processor: 4.00 CPUs

```

ROHA report update

The **c1mgr view report roha** command reports the ROHA data, as shown in Example 9-23. It also has updates about the resources of P770D-01, Enterprise Pool, and On/Off CoD.

Example 9-23 ROHA data after acquiring resources in Example 2

```

# c1mgr view report roha
Cluster: ITSO_ROHA_cluster of NSC type
Cluster tunables
    Dynamic LPAR
        Start Resource Groups even if resources are insufficient: '0'
        Adjust Shared Processor Pool size if required: '0'
        Force synchronous release of DLPAR resources: '0'
    On/Off CoD
        I agree to use On/Off CoD and be billed for extra costs: '1'
        Number of activating days for On/Off CoD requests: '30'
    Node: ITSO_S1Node1
        HMCs: 9.3.207.130 9.3.207.133
        Managed system: rar1m3-9117-MMD-1016AAP
        LPAR: ITSO_S1Node1
            Current profile: 'ITSO_profile'
            Memory (GB):      minimum '8'  wanted '32'  current '158'
            maximum '160'
            Processing mode: Shared
            Shared processor pool: 'DefaultPool'
            Processing units:  minimum '0.5'  wanted '1.5'  current '8.5'
            maximum '9.0'
            Virtual processors: minimum '1'  wanted '3'  current '17'  maximum
            '18'
            ROHA provisioning for 'ONLINE' resource groups
                No ROHA provisioning.
            ROHA provisioning for 'OFFLINE' resource groups
                No 'OFFLINE' resource group.
    Node: ITSO_S2Node1
        HMCs: 9.3.207.130 9.3.207.133
        Managed system: r1r9m1-9117-MMD-1038B9P
        LPAR: ITSO_S2Node1
            Current profile: 'ITSO_profile'
            Memory (GB):      minimum '8'  wanted '32'  current '32'  maximum
            '160'

```

```

        Processing mode: Shared
        Shared processor pool: 'DefaultPool'
        Processing units: minimum '0.5' wanted '1.5' current '1.5'
maximum '9.0'
        Virtual processors: minimum '1' wanted '3' current '3' maximum
'18'
        ROHA provisioning for 'ONLINE' resource groups
            No 'ONLINE' resource group.
        ROHA provisioning for 'OFFLINE' resource groups
            No ROHA provisioning.

Hardware Management Console '9.3.207.130'
Version: 'V8R8.3.0.1'

Hardware Management Console '9.3.207.133'
Version: 'V8R8.3.0.1'

Managed System 'rar1m3-9117-MMD-1016AAP'
    Hardware resources of the managed system
        Installed: memory '192' GB      processing units '12.00'
        Configurable: memory '173' GB      processing units '11.00'
        Inactive: memory '19' GB      processing units '1.00'
        Available: memory '0' GB      processing units '0.50'
    On/Off CoD
        On/Off CoD memory
            State: 'Running'
            Available: '9277' GB.days
            Activated: '21' GB
            Left: '630' GB.days
        On/Off CoD processor
            State: 'Running'
            Available: '9869' CPU.days
            Activated: '3' CPUs
            Left: '90' CPU.days
            Yes: 'DEC_2CEC'
    Enterprise pool
        Yes: 'DEC_2CEC'
    Hardware Management Console
        9.3.207.130
        9.3.207.133
    Shared processor pool 'DefaultPool'
    Logical partition 'ITSO_S1Node1'
        This 'ITSO_S1Node1' partition hosts 'ITSO_S2Node1' node of the NSC cluster
'ITSO_ROHA_cluster'

...
Enterprise pool 'DEC_2CEC'
    State: 'In compliance'
    Master HMC: 'e16hmc1'
    Backup HMC: 'e16hmc3'
    Enterprise pool memory
        Activated memory: '100' GB
        Available memory: '0' GB
        Unreturned memory: '0' GB
    Enterprise pool processor
        Activated CPUs: '4'
        Available CPUs: '0'
        Unreturned CPUs: '0'
Used by: 'rar1m3-9117-MMD-1016AAP'

```

```
Activated memory: '100' GB
Unreturned memory: '0' GB
Activated CPUs: '4' CPUs
Unreturned CPUs: '0' CPUs
Used by: 'r1r9m1-9117-MMD-1038B9P'
Activated memory: '0' GB
Unreturned memory: '0' GB
Activated CPUs: '0' CPUs
Unreturned CPUs: '0' CPUs
```

How to calculate the On/Off CoD consumption

In this case, before bringing the two RGs online, review the remaining resources in On/Off CoD (Example 9-24).

Example 9-24 Remaining resources in On/Off CoD before resource acquisition

```
On/Off CoD memory
  State: 'Available'
  Available: '9907' GB.days
On/Off CoD processor
  State: 'Available'
  Available: '9959' CPU.days
```

After the RG acquisition completes, the status of the On/Off CoD resources is shown in Example 9-25.

Example 9-25 Status of the memory resources

```
On/Off CoD memory
  State: 'Running'
  Available: '9277' GB.days
  Activated: '21' GB
  Left: '630' GB.days
On/Off CoD processor
  State: 'Running'
  Available: '9869' CPU.days
  Activated: '3' CPUs
  Left: '90' CPU.days
```

For the processor resource, PowerHA SystemMirror assigns three processors and the activation day is 30 days, so the total is 90 CPU.Day. ($3 \times 30 = 90$), and the remaining available CPU.Day in the On/Off CoD is 9869 ($9959 - 90 = 9869$).

For memory, PowerHA SystemMirror assigns 21 GB and the activation day is 30 days, so the total is 630 GB.Day. ($21 \times 30 = 630$), and the remaining available GB.Day in On/Off CoD is 9277 ($9907 - 630 = 9277$).

Bringing one resource group offline

This section introduces the process of bringing an RG offline. Figure 9-33 on page 437 shows the overall process.

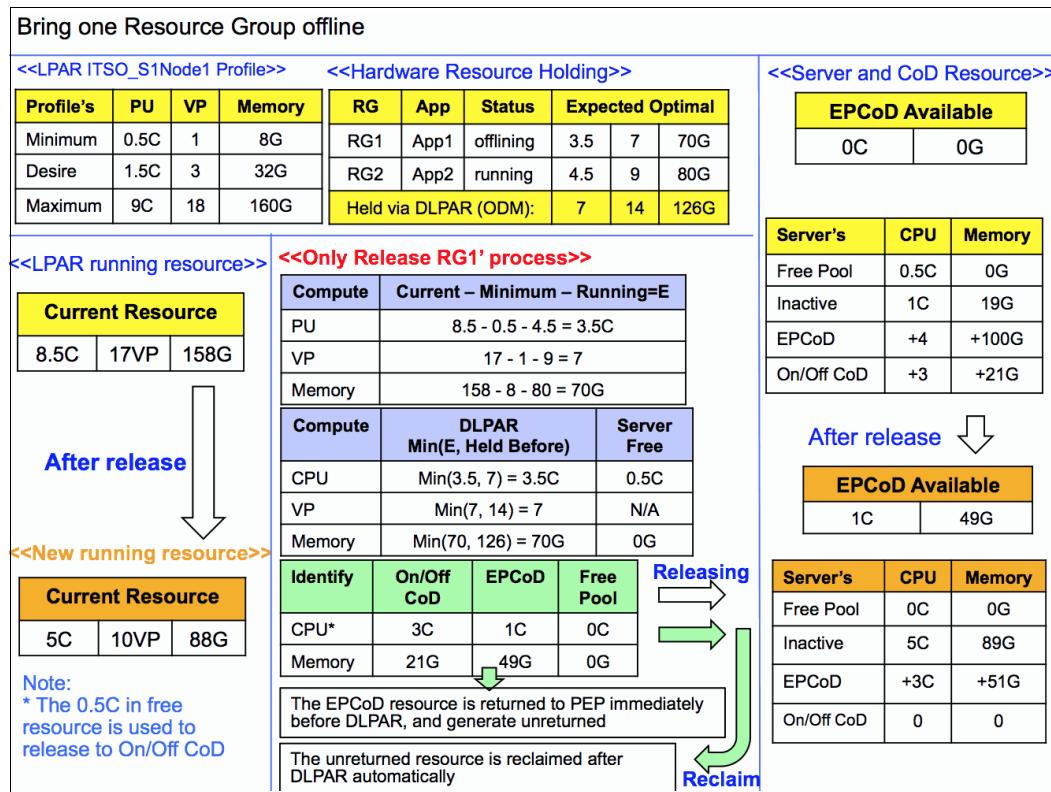


Figure 9-33 Overall release process of Example 2

The process is similar to the one that is shown in “Moving a resource group to another node” on page 419. In the release process, the deallocation order is as follows because there is an extra charge for the On/Off CoD:

1. On/Off CoD
2. EPCoD
3. Server’s free pool

After the release process completes, you can find detailed information about compute, identify, and release processes in the hacmp.out file, as shown in Example 9-26.

Example 9-26 The hacmp.out log information in the release process of Example 2

```
===== Compute ROHA Memory =====
minimum + running = total <=> current <=> optimal <=> saved
8.00 + 80.00 = 88.00 <=> 158.00 <=> 70.00 <=> 126.00 : => 70.00 GB
===== End =====
== Compute ROHA PUs/VP(s) ==
minimal + running = total <=> current <=> optimal <=> saved
1 + 9 = 10 <=> 17 <=> 7 <=> 14 : => 7 Virtual Processors
minimal + running = total <=> current <=> optimal <=> saved
0.50 + 4.50 = 5.00 <=> 8.50 <=> 3.50 <=> 7.00 : => 3.50 Processing Units
===== End =====
==== Identify ROHA Memory ====
Total Enterprise Pool memory to return back: 49.00 GB
Total On/Off CoD memory to de-activate: 21.00 GB
Total DLPAR memory to release: 70.00 GB
===== End =====
== Identify ROHA Processor ==
```

```
Total Enterprise Pool CPUs to return back: 1.00 CPUs
Total On/Off CoD CPUs to de-activate: 3.00 CPUs
Total DLPAR PUs/VP(s) to release: 7.00 Virtual Processors and 3.50
Processing Units
=====
c1hmccmd: 49.00 GB of Enterprise Pool CoD have been returned.
c1hmccmd: 1 CPUs of Enterprise Pool CoD have been returned.
The following resources were released for application controllers App1Controller.
DLPAR memory: 70.00 GB      On/Off CoD memory: 21.00 GB      Enterprise Pool memory: 49.00
GB.
DLPAR processor: 3.50 PU/7.00 VP      On/Off CoD processor: 3.00 CPUs      Enterprise Pool
processor: 1.00 CPUs
```

9.5 Live Partition Mobility

With LPM, you can migrate partitions that are running AIX and Linux operating systems and their hosted applications from one physical server to another one without disrupting the infrastructure services. The migration operation, which takes only several seconds, maintains complete system transactional integrity. The migration transfers the entire system environment, including processor state, memory, attached virtual devices, and connected users.

With LPM, you can eliminate downtime for planned hardware maintenance. For other downtime, such as for required software maintenance or unplanned outages, you can use PowerHA to minimize downtime for those events.

You can use PowerHA within a partition that can be moved with LPM. The combination has been supported for many years. For more information, see [IBM Support page – Support for LPM](#).

However, PowerHA does not use LPM in any way. PowerHA is treated as another application within the partition. Before PowerHA 7.2.0, performing LPM on a PowerHA SystemMirror node in a cluster was a multi-step manual process. Here is an overview of the process:

1. Stop cluster services on the node by using the **Unmanage** option.
2. Disable Dead Man Switch monitoring in the cluster.
3. Perform the LPM.
4. Enable Dead Man Switch monitoring in the cluster.
5. Start cluster services on the node with the **Auto manage** option.

However, with PowerHA 7.2.x, LPM support is integrated to simplify this process by performing similar steps automatically. For more information about this feature, which is known as LPM Node Policy, see 12.3, “Cluster tunables” on page 505.

If your environment has SANcomm defined, then there are more actions that you must take to perform LPM, as described in 9.5.1, “Performing LPM with SANcomm defined” on page 439.

9.5.1 Performing LPM with SANcomm defined

We did this test example to show that it can be done. However, the combination is *only* supported within the guidelines that are listed in the following Important box. This testing was performed before these guidelines were published.

Important: You can perform LPM on a PowerHA SystemMirror LPAR that is configured with SAN communication. However, when you use LPM, the SAN communication is not automatically migrated to the destination system. Configure SAN communication on the destination system before you use LPM. For more information, see the [PowerHA SystemMirror 7.2 for AIX documentation](#).

We used the following scenarios:

- ▶ Target managed systems *do not* have SANComm defined.
- ▶ Target managed systems *do* have SANComm defined and available.

In our scenario, we have a 2-node cluster with nodes jessica and shanley, each on their own managed systems that are named p750_4 and p750_2 respectively. Both systems and nodes are using SANComm. We have a third managed system, p750_3, in which SANComm is *not* configured. However, its VIOS adapters are target-mode-capable and enabled.

In both scenarios, when we first run the migration during the verification process, a warning is displayed, as shown in Figure 9-34. In this case, you can ignore the warning and continue.

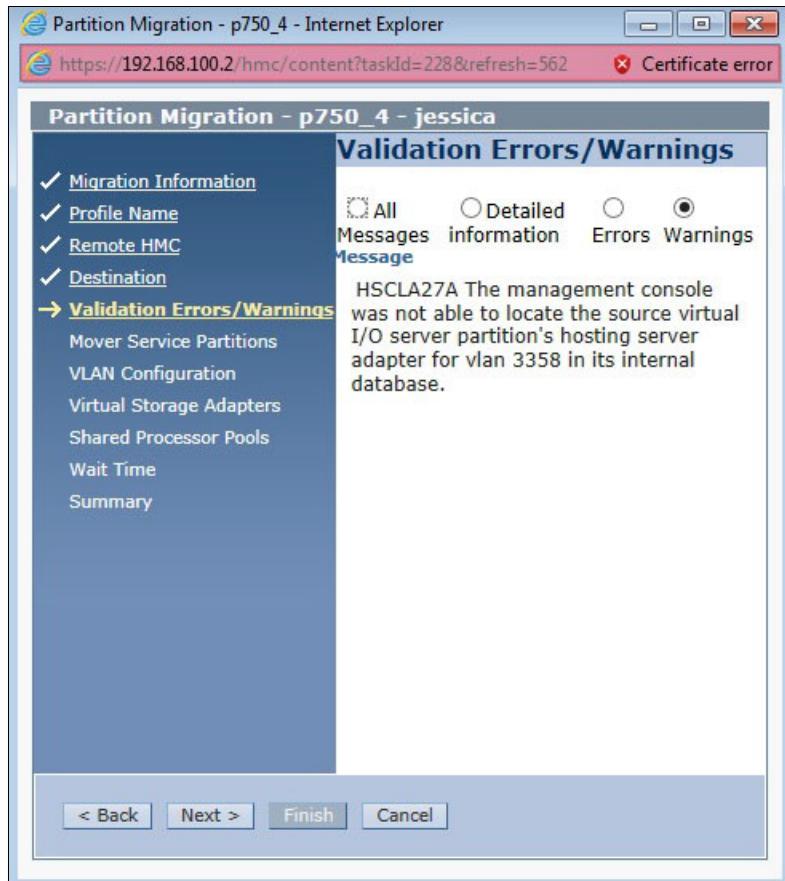


Figure 9-34 LPM warning about SANComm and VLAN3358

The LPM process completes and node jessica is active and running on p750_3. However, SANComm is no longer functioning, as shown by the lack of output from the `lscluster` command output that is shown in Example 9-27.

Example 9-27 The lscluster command output

```
[jessica:root] /utilities # lscluster -m |grep sfw
[jessica:root] /utilities #
```

We add new virtual Ethernet adapters that use VLAN3358 to each VIOS. Then, we run `cfmgr` on each VIOS to configure the sfwcomm device. No further action is required on node jessica because its profile already contains the proper virtual adapter.

The sfwcom devices automatically appear on jessica, as shown in Example 9-28.

Example 9-28 The sfwcom devices automatically appear on jessica

```
[jessica:root] /utilities # lscluster -m |grep sfw
      sfwcom      UP      none      none      none
```

Demonstration: For a demonstration of this exact scenario, see the following [YouTube video](#).

In the second scenario, we repeat the LPM. However, this time, the target system already has both SANComm devices configured on its VIOS and the appropriate virtual Ethernet adapters. During the LPM, we noticed a couple of seconds in which sfwcom registered as being down, but then it automatically came back online.



Extending resource group capabilities

In this chapter, we describe how you can use the PowerHA advanced resource group (RG) capabilities to meet the specific requirements of particular environments.

This chapter contains the following topics:

- ▶ The settling time attribute
- ▶ Serial processing order
- ▶ Node distribution policy
- ▶ Dynamic node priority
- ▶ Delayed fallback timer
- ▶ Resource group dependencies

10.1 Resource group attributes

There are several attributes for RGs that influence the way that they react to different failure scenarios in the cluster. Setting these attributes correctly can help you ensure that the behavior of the RGs in your cluster meets your expectations when your cluster first starts and how they are managed during failover and fallback events.

The attributes that are listed in Table 10-1 can influence the behavior of RGs during startup, failover, and fallback. They are described in the following sections.

Table 10-1 Resource group attribute behavior relationships

Attribute	Startup	Fallover	Fallback
Settling time	Yes	N/A	N/A
Serial processing order	Yes	Yes	Yes
Node distribution policy	Yes	N/A	N/A
Dynamic node priority (DNP)	N/A	Yes	N/A
Delayed fallback timer	N/A	N/A	Yes
Resource group parent-child dependency	Yes	Yes	Yes
Resource group location dependency	Yes	Yes	Yes
Resource group start after/stop after dependency	Yes	Yes	Yes

10.2 The settling time attribute

PowerHA SystemMirror acquires RGs in parallel, but if the settling time or the delayed fallback timer policy is configured for a particular RG, PowerHA SystemMirror delays its acquisition for the duration that is specified in the timer policy.

With the settling time attribute, you can delay the acquisition of an RG so that if a higher priority node joins the cluster during the settling period, the RG is brought online on the higher priority node instead of being activated on the first available node.

10.2.1 Behavior of the settling time attribute

The following characteristics apply to the settling time:

- ▶ If configured, the settling time attribute affects the startup behavior of all offline RGs in the cluster for which you selected the Online on First Available Node (OFAN) startup policy.
- ▶ The only time that the attribute is ignored is when the node that is joining the cluster is the first node in the node list for the RG. In this case, the RG is acquired immediately.
- ▶ If an RG is in the ERROR state, PowerHA waits for the settling time period before attempting to bring the RG online.
- ▶ The current settling time remains active until the RG moves to another node or goes offline. A dynamic automatic reconfiguration (DARE) operation might result in the release and reacquisition of an RG, in which case the new settling time values are effective immediately.

10.2.2 Configuring the settling time for resource groups

To configure the settling time for RGs, complete the following steps:

1. Enter the **smitty sysmirror** fast path, select **Cluster Applications and Resources → Resource Groups → Configure a Resource Group Run-Time Policies → Configure Settling Time for Resource Group**, and press Enter.
2. Enter a field value for Settling Time. Enter any positive integer number into this field. The default is 0, as shown in Figure 10-1.

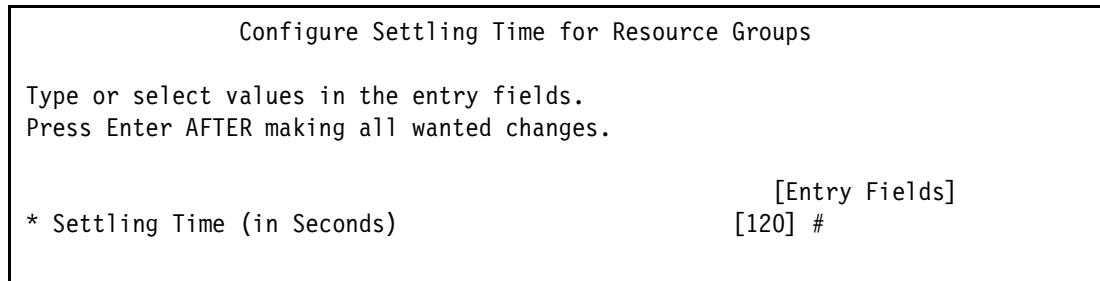


Figure 10-1 System Management Interface Tool panel to configure the settling time

You can also set the field value by running the following command from the CLI:

```
c1mgr modify cluster RG_SETTLING_TIME="120"
```

3. Synchronize the cluster by running **c1mgr sync cluster**.

If this value is set and the node that joins the cluster is not the highest priority node, the RG waits for the duration of the settling time interval. When this time expires, the RG is acquired on the node that has the highest priority among the list of nodes that joined the cluster during the settling time interval.

This approach is valid only for RGs that use the startup policy Online or First Available Node.

10.2.3 Displaying the current settling time

To display the current settling time in a cluster that is already configured, run the **c1mgr -an RG_SETTLING_TIME query cluster** command:

```
#c1mgr -an RG_SETTLING_TIME query cluster  
RG_SETTLING_TIME="120"
```

During the acquisition of the RGs on cluster startup, you can also see the settling time value by running the **c1RGinfo -t** command, as shown in Example 10-1.

Example 10-1 Displaying the RG settling time

#/usr/es/sbin/cluster/utilities/c1RGinfo -t			
Group Name	Group State	Node	Delayed Timers
xsiteGLVMRG	ONLINE	jessica@dallas	120 Seconds
	ONLINE SECONDARY	ashley@fortwor	120 Seconds
newconcRG	ONLINE	jessica	120 Seconds
	OFFLINE	maddi	120 Seconds

Note: A settling time with a nonzero value is displayed only during the acquisition of the RG. The value will be set to 0 after the settling time expires and the RG is acquired by the appropriate node.

10.2.4 Settling time scenarios

To demonstrate how this feature works, we created two settling time scenarios and configured a 2-node cluster by using a single RG. In our scenario, we showed the following characteristics:

- ▶ The settling time period is enforced, and the RG is not acquired on the node startup (while the node is not the highest priority node) until the settling time expires.
- ▶ If the highest priority node joins the cluster during the settling period, then it does not wait for settling time to expire and acquires the RG immediately.

We specified a settling time of 6 minutes and configured an RG that is named SettleRG1 to use the startup policy Online on First Available Node. We set the node list for the RG so that node jessica would failover to node maddi.

For the first test, the following steps demonstrate how we let the settling time expire and how the secondary node acquires the RG:

1. With cluster services inactive on all nodes, define a settling time value of 360 seconds.
2. Synchronize the cluster by running **clmgr sync cluster**.
3. Validate the settling time by running **clmgr** as follows:

```
[jessica:root] / # clmgr -an RG_SETTLING_TIME query cluster  
RG_SETTLING_TIME="120"
```

4. Start the cluster services on node maddi.

We started cluster services on this node because it was the last node in the list for the RG. After starting cluster services, the RG was acquired by node maddi. Running the **clRGinfo -t** command displays the 360 seconds settling time, as shown in Example 10-2.

Example 10-2 Checking the settling time in /var/hacmp/log/hacmp.out

[maddi:root] / # clRGinfo -t			
Group Name	State	Node	Delayed Timers
SettleRG1	OFFLINE	jessica	360 Seconds
	OFFLINE	maddi	360 Seconds

5. Wait for the settling time to expire.

Upon the expiration of the settling time, SettleRG1 was acquired by node maddi. Because the first node in the node list (jessica) did not become available within the settling time period, the RG was acquired on the next node in the node list (maddi). See Figure 10-2 on page 445.

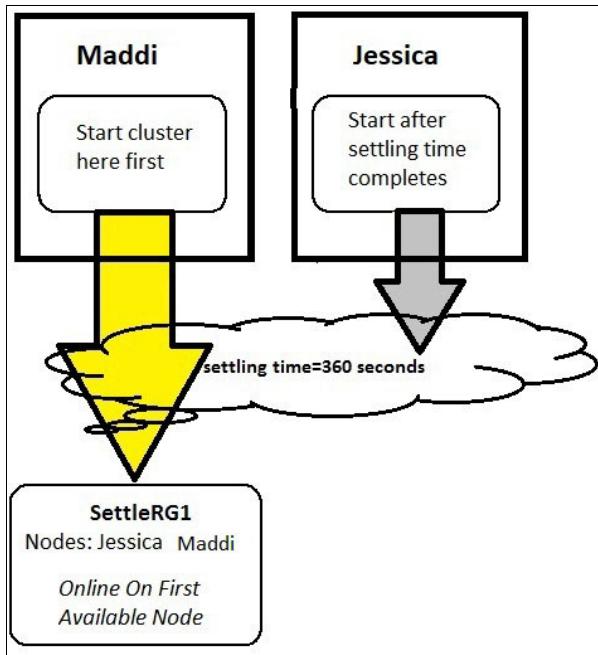


Figure 10-2 Settling time scenario waiting

For the next test scenario, we demonstrate how the primary node starts the RG when the settling time does not expire. Complete the following steps:

1. Repeat steps 1 on page 444 - 4 on page 444.
2. Start the cluster services on node jessica.

After waiting about 2 minutes for the cluster to stabilize on node maddi, we start cluster services on node jessica. The RG is brought online to node jessica, as shown in Figure 10-3.

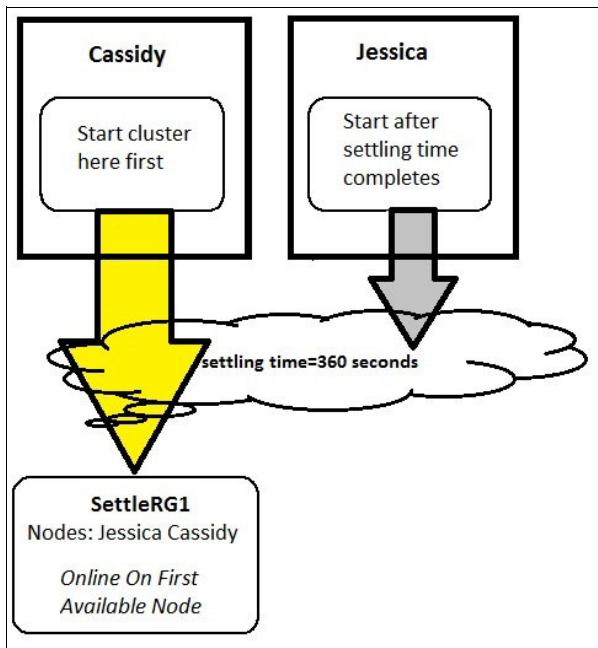


Figure 10-3 Settling time scenario with no waiting

Note: This feature is effective only when cluster services on a node are started. This feature is *not* enforced when Cluster Single Point of Control (C-SPOC) is used to bring an RG online.

10.3 Serial processing order

By default, PowerHA SystemMirror acquires and releases RGs in parallel. One of many of the possible RG behavior policies is to specify an exact acquisition or release order.

Any pair of RGs that do not have the following attributes might be processed in any order, even if one of the RGs of the pair has a relationship (serial order or dependency) with another RG.

The following RG attributes affect the acquisition order:

- ▶ RGs that are configured with a serial acquisition order are acquired in the specified order.
- ▶ RGs that are configured with dependencies with other RGs are acquired in phases. Parent RGs are acquired before child RGs.
- ▶ RGs that are configured as the target for a Start After dependency are acquired before RGs configured as the Source for the dependency.
- ▶ RGs that include Network File System (NFS) mounts are acquired before RGs that do not include NFS mounts.

The following RG attributes affect the releasing order:

- ▶ RGs that are configured with a serial release order are released in that specified order.
- ▶ RGs that are configured with dependencies with other RGs are released in phases. Child RGs are released before parent RGs. RGs that are configured as the target for the Stop After dependency are released before RGs that are configured as the Source for the dependency.
- ▶ RGs that include NFS mounts are released after RGs that do not include NFS mounts.

10.3.1 Configuring a serial (acquisition and release) processing order

To configure a serial processing order, complete the following steps:

1. Enter the **smitty sysmirror** fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Configure Resource Group Run-Time Policies** → **Configure Resource Group Processing Ordering**, and press Enter.
2. Complete both the Acquisition and Release order. Our scenario is shown in Figure 10-4 on page 447. Press Enter twice to confirm.

Although you can press F4 and get an RG list, the RGs are displayed in alphanumeric order. If you chose more than one, they are put into the field in alphanumeric order, which may not be what you want. As a best practice, type in the RG names while using F4 as a reference. The names are separated by a space in the wanted order in each field.

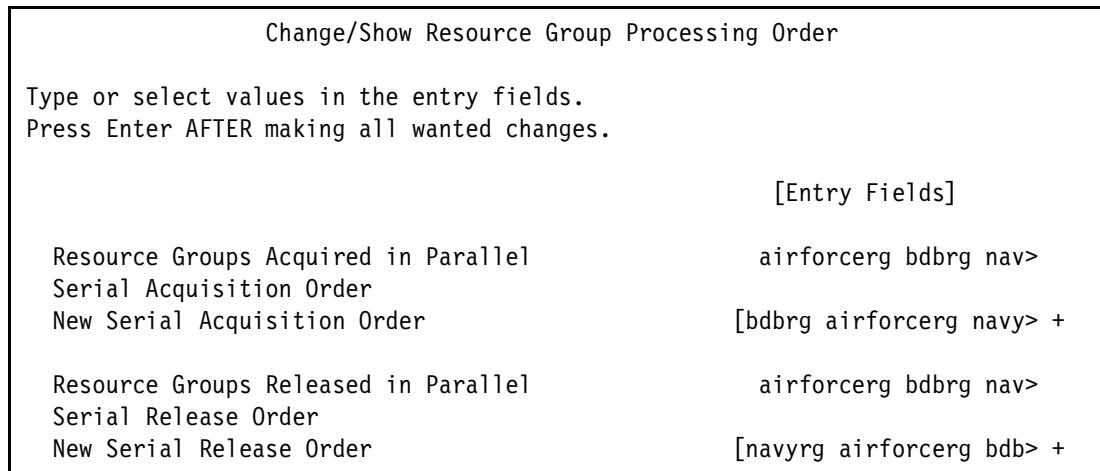


Figure 10-4 Customizing serial resource group processing order

Attention: In PowerHA 7.2.7, the following warning is displayed and should be noted:

WARNING: The RG serial acquisition and serial release ordering will be removed in a future PowerHA SystemMirror release.

3. Synchronize the cluster by running `clmgr sync cluster`.

10.3.2 Serial processing order scenario

Using the configuration that is shown in Figure 10-4, start the cluster and observe the behavior from repetitive running of `c1RGinfo`, as shown in Example 10-3.

Example 10-3 Serial processing startup order results

Group Name	State	Node
bdbrg	ACQUIRING OFFLINE	db2 web
navyrg	OFFLINE OFFLINE	web db2
airforcerg	OFFLINE OFFLINE	db2 web

Group Name	State	Node
bdbrg	ONLINE OFFLINE	db2 web
navyrg	OFFLINE OFFLINE	web db2
airforcerg	ACQUIRING	db2

	OFFLINE	web
<hr/>		
Group Name	State	Node
bdbrg	ONLINE OFFLINE	db2 web
navyrg	ACQUIRING OFFLINE	web db2
airforcerg	ONLINE OFFLINE	db2 web
<hr/>		
Group Name	State	Node
bdbrg	ONLINE OFFLINE	db2 web
navyrg	ONLINE OFFLINE	web db2
airforcerg	ONLINE OFFLINE	db2 web

When the cluster stops, the inverse, based on the configuration, occurs from repetitive execution of **c1RGinfo**, as shown in Example 10-4.

Example 10-4 Serial processing release order results

Group Name	State	Node
<hr/>		
bdbrg	ONLINE OFFLINE	db2 web
navyrg	RELEASING OFFLINE	web db2
airforcerg	ONLINE OFFLINE	db2 web
<hr/>		
Group Name	State	Node
bdbrg	ONLINE OFFLINE	db2 web
navyrg	OFFLINE OFFLINE	web db2
airforcerg	RELEASING OFFLINE	db2 web

Group Name	State	Node
bdbrg	RELEASING OFFLINE	db2 web
navyrg	OFFLINE OFFLINE	web db2
airforcerg	OFFLINE OFFLINE	db2 web
Group Name	State	Node
bdbrg	OFFLINE OFFLINE	db2 web
navyrg	OFFLINE OFFLINE	web db2
airforcerg	OFFLINE OFFLINE	db2 web

10.4 Node distribution policy

One of the startup policies that can be configured for RGs is the Online Using Node Distribution policy.

RGs with this startup policy spread across cluster nodes in such a way that only one RG is acquired by any node during startup. This policy can be used, for example, for distributing CPU-intensive applications on different nodes.

If two or more RGs are offline when a particular node joins the cluster, this policy determines which RG is brought online based on the following criteria and order of precedence:

1. The RG with the fewest participating nodes is acquired.
2. A parent RG is preferred over an RG that does not have any child RGs.

Restriction: When you use the node distribution startup policy, the fallback policy must be set to Never Failback. Otherwise, the following error is displayed:

ERROR: Invalid configuration.

Resource Groups with Startup Policy 'Online Using Distribution Policy' can have Only 'Never Failback' as Fallback Policy.

10.4.1 Configuring a resource group node-based distribution policy

To configure this type of startup policy, complete the following steps:

1. Enter the **smitty sysmirror** fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Add a Resource Group**, and press Enter.
2. Specify an RG name.

3. Select the Online Using Node Distribution Policy startup policy, as shown in Example 10-5.
 4. Select Never Failback for the failback policy as required, and press Enter.
 5. Synchronize the cluster by running `clmgr sync cluster`.

Example 10-5 Configuring a resource group node-based distribution policy

Add a Resource Group (extended)

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

* Resource Group Name	[Entry Fields]	
* Participating Nodes (Default Node Priority)	[bdbrg] [jessica maddi]	
Startup Policy	Online Using Node Dis...	
Fallover Policy	Fallover To Next Prio...	
Fallback Policy	Never Fallback>	
+-----+-----+		
Startup Policy		
Move cursor to the wanted item and press Enter.		
+-----+-----+		
Online on Home Node Only		
Online on First Available Node		
Online Using Node Distribution Policy		
Online On All Available Nodes		
+-----+-----+		
F1=Help	F2=Refresh	F3=Cancel
F8=Image	F10=Exit	Enter=Do
/=Find	n=Find Next	
+-----+-----+		

10.4.2 Node-based distribution scenario

To show how this feature functions and understand the difference between this policy and the Online on Home Node Only (OHNO) policy, we created a node-based distribution scenario and configured a 2-node cluster with three RGs that all use the Online Using Node Distribution policy. The cluster nodes and RGs are shown in Figure 10-5 on page 451. The number of RGs with the Online Using Node Distribution policy is greater than the number of cluster nodes.

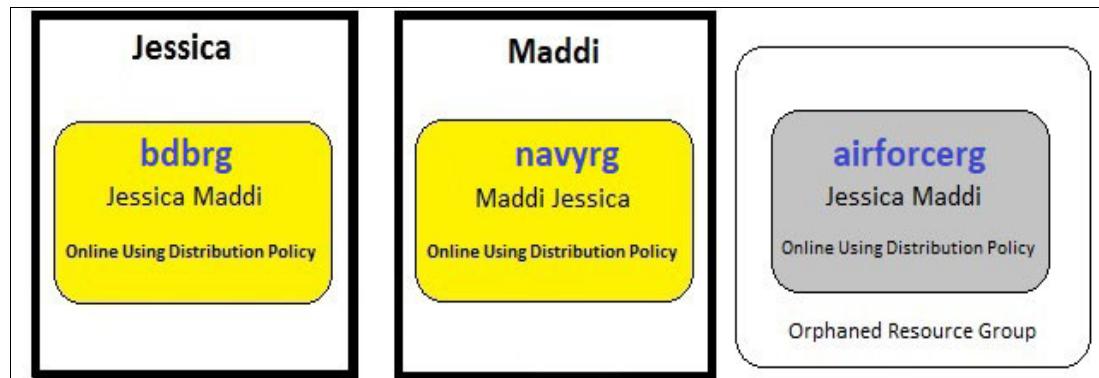


Figure 10-5 *Online Using Node Distribution policy scenario*

For our scenario, we completed the following steps:

1. Start cluster services on node jessica. bdbrg was acquired, as shown in Example 10-6.

Example 10-6 Node jessica starts bdbrg

```
[jessica:root] / # cRGinfo
```

Group	Name	State	Node
bdbrg		ONLINE	jessica
		OFFLINE	maddi
navyrg		OFFLINE	maddi
		OFFLINE	jessica
airforcerg		OFFLINE	jessica
		OFFLINE	maddi

2. Start the cluster services on node maddi. navyrg was acquired, as shown in Example 10-7.

Example 10-7 Node maddi starts navyrg

```
[jessica:root] / # cRGinfo
```

Group	Name	State	Node
bdbrg		ONLINE	jessica
		OFFLINE	maddi
navyrg		ONLINE	maddi
		OFFLINE	jessica
airforcerg		OFFLINE	jessica
		OFFLINE	maddi

airforcerg stays offline, but can be brought online manually through C-SPOC. To do so, run **smitty cspoc**, select **Resource Group and Applications** → **Bring a Resource Group Online**, select airforcerg, and then select the node on which you want to start it.

10.5 Dynamic node priority

The default node priority order for an RG is the order in the participating node list. By implementing a DNP for an RG, you may go beyond the default failover policy behavior and influence the destination of an RG on failover. The two types of dynamic node priorities are as follows:

- ▶ Predefined Resource Monitoring and Control (RMC)-based: These DNPs are included as standard with the PowerHA base product.
- ▶ Adaptive failover: These DNPs are two extra priorities that require customization by the user.

Important: DNP is an available option only to clusters with three or more nodes that participate in the RG.

Predefined RMC-based dynamic node priorities

These priorities are based on the following three RMC preconfigured attributes:

- ▶ `c1_highest_free_mem` (node with the highest percentage of free memory)
- ▶ `c1_highest_idle_cpu` (node with the most available processor time)
- ▶ `c1_lowest_disk_busy` (node with the least busy disks)

The cluster manager queries the RMC subsystem every 3 minutes to obtain the current value of these attributes on each node and distributes them cluster-wide. The interval at which the queries of the RMC subsystem are performed is not user-configurable. During a failover event of an RG with DNP configured, the most recently collected values are used in the determination of the best node to acquire the RG.

For DNP to be effective, consider the following information:

- ▶ DNP cannot be used with fewer than three nodes.
- ▶ DNP cannot be used for Online on All Available Nodes RGs.
- ▶ DNP is most useful in a cluster where all nodes have equal processing power and memory.

Important: The highest free memory calculation is performed based on the amount of paging activity taking place. It does not consider whether one cluster node has less real physical memory than another one.

For more information about how predefined DNP values are used, see 10.5.2, “How predefined RMC-based dynamic node priority functions” on page 455.

Adaptive failover dynamic node priority

Introduced in PowerHA 7.1, you can choose DNP based on a user-defined property by selecting one of the following attributes:

- ▶ `c1_highest_udscript_rc`
- ▶ `c1_lowest_nonzero_udscript_rc`

When you select one of these criteria, you must also provide values for the DNP script path and DNP timeout attributes for an RG. When the DNP script path attribute is specified, the script is invoked on all nodes and return values are collected from all nodes. The failover node decision is made by using these values and the specified criteria.

If you choose the `c1_highest_udscript_rc` attribute, collected values are sorted and the node that returned the highest value is selected as a candidate node to failover. Similarly, if you choose the `c1_lowest_nonzero_udscript_rc` attribute, collected values are sorted and the node that returned the lowest nonzero positive value is selected as a candidate node to fail over. If the return values of the script from all nodes are the same or zero, the default node priority is considered. PowerHA verifies the script existence and the execution permissions during verification.

Demonstration: For a demonstration of user-defined adaptive failover node priority, see the following [YouTube video](#).

10.5.1 Configuring a resource group with a predefined RMC-based DNP policy

When a DNP is set up for an RG, no resources can already be assigned to the RG. Assign the failover policy of DNP when the RG is created. For your RG to use one of the three DNP policies, you must set the failover policy as shown in Example 10-8:

1. Enter the `smitty sysmirror` fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Add a Resource Group**, and press Enter.
2. Set the Failover Policy field to Failover Using Dynamic Node Priority, as shown in Example 10-8. Press Enter.

Example 10-8 Adding a resource group by using DNP

Add a Resource Group (extended)

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

* Resource Group Name	[Entry Fields]
* Participating Nodes (Default Node Priority)	[DNP_test1] [ashley jessica maddi] +
Startup Policy	Online On Home Node 0> +
Failover Policy	Failover Using Dynami> +
Fallback Policy	Fallback To Higher Pr> +

3. Assign the resources to the RG by using the `smitty sysmirror` fast path. Select **Cluster Applications and Resources** → **Resource Groups** → **Change>Show Resources and Attributes for a Resource Group**, select the newly created RG from the list, and press Enter.

4. Select one of the available policies by pressing F4 in the Dynamic Node Priority Policy field, as shown in Figure 10-6.

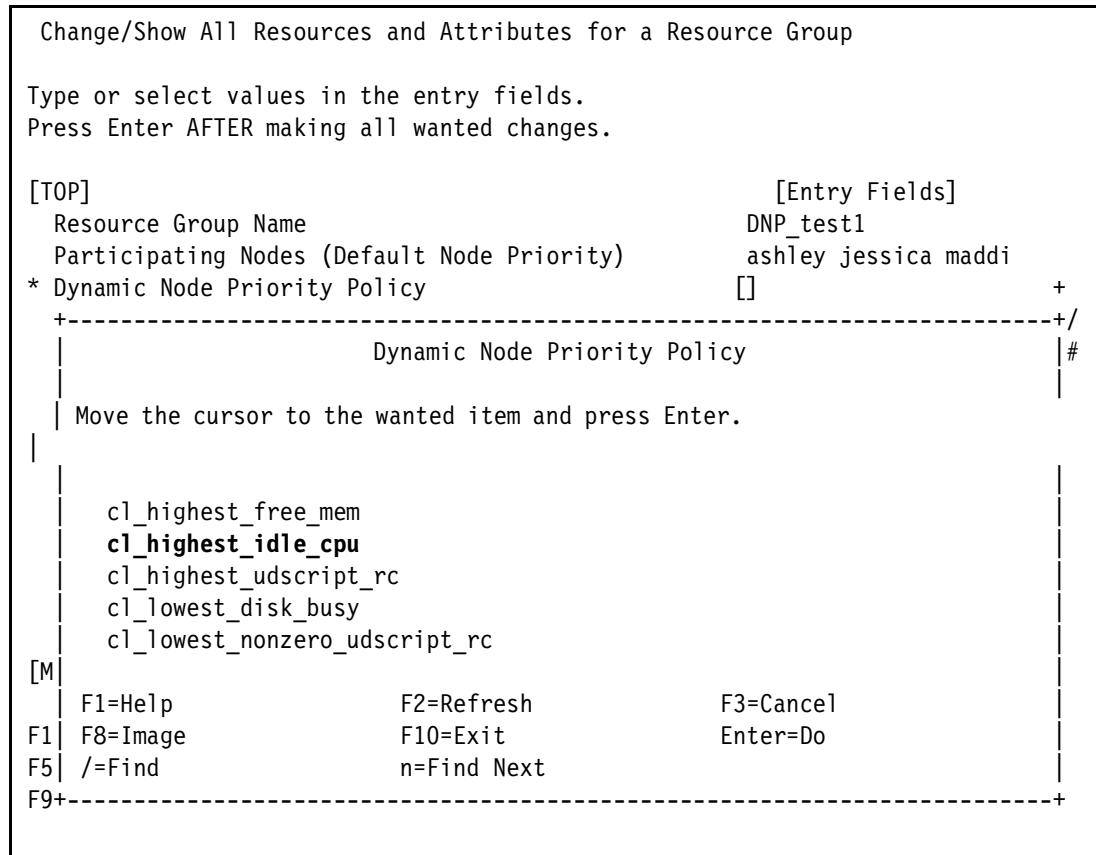


Figure 10-6 Selecting the preferred Dynamic Node Priority Policy

5. Continue selecting the resources that will be part of the RG, as shown in Figure 10-7 on page 455. Press Enter.

Change/Show All Resources and Attributes for a Resource Group		
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.		
[TOP]	[Entry Fields]	
Resource Group Name	DNP_test1	
Participating Nodes (Default Node Priority)	ashley jessica maddi	
* Dynamic Node Priority Policy	[cl_highest_idle_cpu] +	
DNP Script path	[] /	
DNP Script timeout value	[] #	
Startup Policy	Online On Home Node 0>	
Fallover Policy	Fallover Using Dynamical	
Fallback Policy	Fallback To Higher Pr	
Fallback Timer Policy (empty is immediate)	[] +	
Service IP Labels/Addresses	[bdbsvc]	+

Figure 10-7 DNP script path and timeout attributes

6. Synchronize the cluster by running `clmgr sync cluster`.

You can display the current DNP policy for an existing RG, as shown in Example 10-9.

Example 10-9 Displaying the DNP policy for a resource group

```
root@maddi[] odmget -q group=DNP_test1 HACMPresource|more
HACMPresource:
    group = "DNP_test1"
    type = ""
    name = "NODE_PRIORITY_POLICY"
    value = "cl_highest_idle_cpu"
    ID = 1
    monitor_method = ""
```

Notes:

- ▶ Using the information that is retrieved directly from the Object Data Manager (ODM) is for informational purposes only because the format within the stanzas might differ between versions.
- ▶ Hardcoding ODM queries within user-defined applications is not supported and should be avoided.

10.5.2 How predefined RMC-based dynamic node priority functions

C1strmgrES polls the RMC (ctrmc) daemon every 3 minutes and maintains a table that stores the current memory, CPU, and disk I/O state of each node.

The following resource monitors contain the information for each policy:

- ▶ IBM.PhysicalVolume
- ▶ IBM.Host

Each of these monitors can be queried during normal operation by running the commands that are shown in Example 10-10.

Example 10-10 Querying resource monitors

```
root@jessica[] lsrsrc -Ad IBM.Host | grep TotalPgSpFree
    TotalPgSpFree      = 1046264
    PctTotalPgSpFree   = 99.7795
root@jessica[] lsrsrc -Ad IBM.Host | grep PctTotalTimeIdle
    PctTotalTimeIdle   = 92.3674
root@jessica[] lsrsrc -Ap IBM.PhysicalVolume
Resource Persistent Attributes for IBM.PhysicalVolume
resource 1:
    Name          = "hdisk3"
    PVID         = "0x00c472c0 0xde143bdf 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
resource 2:
    Name          = "hdisk7"
    PVID         = "0x00c472c0 0x6f48cdfb 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
resource 3:
    Name          = "hdisk6"
    PVID         = "0x00c472c0 0x6f48ceb0 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
resource 4:
    Name          = "hdisk5"
    PVID         = "0x00f92db1 0xbabc3344 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
resource 5:
    Name          = "hdisk2"
    PVID         = "0x00c472c0 0xde92e337 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
resource 6:
    Name          = "hdisk1"
    PVID         = "0x00c472c0 0x7bcbeb08 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
resource 7:
    Name          = "hdisk4"
    PVID         = "0x00c472c0 0x9f3e94c2 0x00000000 0x00000000"
    ActivePeerDomain = "redbook_cluster"
    NodeNameList   = {"jessica"}
root@maddi[] lsrsrc -Ad IBM.PhysicalVolume
Resource Dynamic Attributes for IBM.PhysicalVolume
resource 1:
    PctBusy     = 1
    RdBlkRate  = 0
    WrBlkRate  = 930
    XferRate   = 4
resource 2:
    PctBusy     = 0
```

```

RdB1kRate = 75
WrB1kRate = 1
XferRate = 1
resource 3:
PctBusy = 0
RdB1kRate = 120
WrB1kRate = 1284
XferRate = 6
resource 4:
PctBusy = 0
RdB1kRate = 0
WrB1kRate = 39
XferRate = 4
resource 5:
PctBusy = 0
RdB1kRate = 0
WrB1kRate = 0
XferRate = 0
resource 6:
PctBusy = 0
RdB1kRate = 0
WrB1kRate = 0
XferRate = 0
resource 7:
PctBusy = 0
RdB1kRate = 0
WrB1kRate = 0
XferRate = 0

```

You can display the current table that is maintained by `clstrmgrES` in an active cluster by running the command that is shown in Example 10-11.

Example 10-11 DNP values that are maintained by the cluster manager

```

[maddi:root] /inst.images # lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#) 8273f4f 43haes/usr/sbin/cluster/hacmprd/main.C, 727, 2238D_aha727,
Aug 24 2022 10:16 PM"
build = "Sep 23 2022 08:32:34 2238D_aha727"
CLversion: 23
local node vrmf is: 7270
cluster fix level is: "0"
i_local_nodeid 2, i_local_siteid -1, my_handle 1
m1_idx[1]=2    m1_idx[2]=1    m1_idx[3]=0
There are 0 events on the Ibcast queue
There are 0 events on the RM Ibcast queue
The following timers are currently active:
Current DNP values:
NodeId 3  NodeName ashley
PgSpFree = 128244  PvPctBusy = 0  PctTotalTimeIdle = 81.898222
NodeId 2  NodeName jessica
PgSpFree = 1045758  PvPctBusy = 0  PctTotalTimeIdle = 99.988795
NodeId 1  NodeName maddi
PgSpFree = 1046217  PvPctBusy = 0  PctTotalTimeIdle = 94.705352
CAA Cluster Capabilities
CAA Cluster services are active

```

```

There are 10 capabilities
Capability 0
    ID: 8 version: 1 flag: 1
    CAA COMDISK capability is defined and globally available
Capability 1
    ID: 9 version: 1 flag: 1
    CAA 4KDISK capability is defined and globally available
Capability 2
    ID: 7 version: 1 flag: 1
    CAA DR capability is defined and globally available
Capability 3
    ID: 6 version: 1 flag: 1
    Sub Cluster Split Merge capability is defined and globally available
Capability 4
    ID: 5 version: 1 flag: 1
    Network Monitor capability is defined and globally available
Capability 5
    ID: 4 version: 1 flag: 1
    Automatic Repository Replacement capability is defined and globally available
Capability 6
    ID: 3 version: 1 flag: 1
    Hostname Change capability is defined and globally available
Capability 7
    ID: 2 version: 1 flag: 1
    Unicast capability is defined and globally available
Capability 8
    ID: 0 version: 1 flag: 1
    IPV6 capability is defined and globally available
Capability 9
    ID: 1 version: 1 flag: 1
    Site capability is defined and globally available
trcOn 0 kTraceOn 0 stopTraceOnExit 0 cdNodeOn 0
Last event run was: JOIN_NODE_CO on node 1 serial 21903
Last admin op event run was: CLRM_START_REQUEST, serial 21901

```

The values in the table are used for the DNP calculation in a failover. If `c1strmgrES` is in the middle of polling the current state when a failover occurs, then the value that was last taken when the cluster was in a stable state is used to determine the DNP.

10.5.3 Configuring a resource group with an adaptive failover DNP policy

When you define DNP for an RG, no resources can already be assigned to the RG. Assign the failover policy of the DNP at the time when the RG is created. For your RG to be able to use one of the DNP policies, you must set the failover policy as shown in Example 10-8 on page 453. Complete the following steps:

1. Enter the `smitty sysmirror` fast path, select **Cluster Applications and Resources → Resource Groups → Add a Resource Group**, and press Enter.
2. Set the Failover Policy field to **Failover Using Dynamic Node Priority**, as shown in Figure 10-8 on page 459.

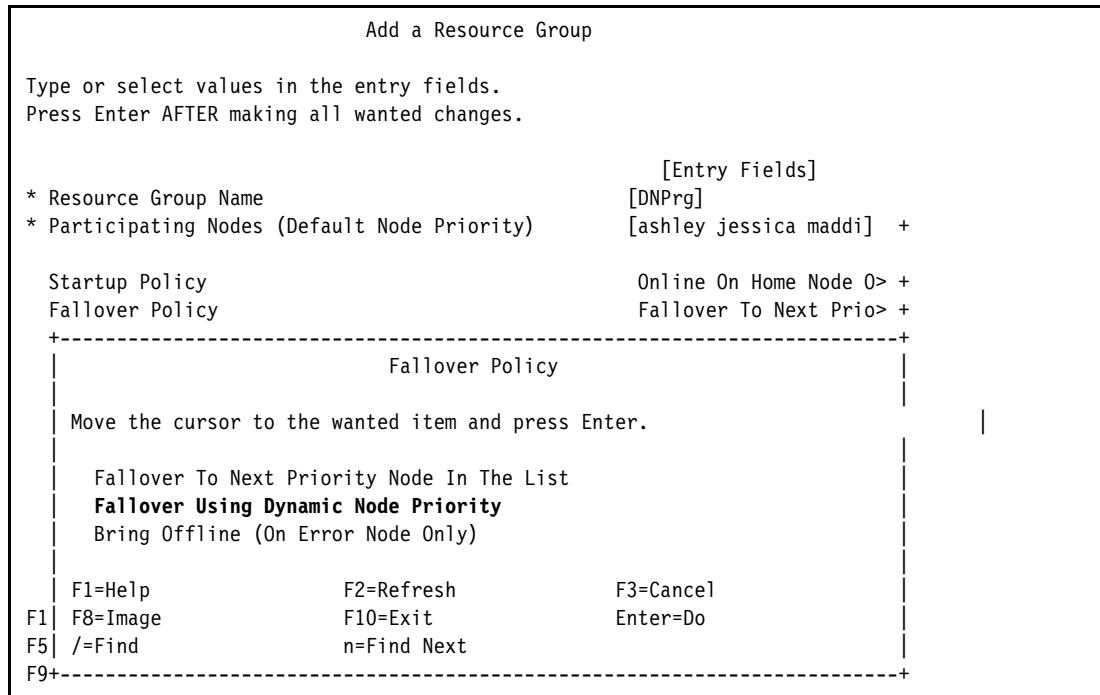


Figure 10-8 Creating a resource group for DNP usage

3. Assign the resources to the RG by running the **smitty sysmirror** fast path, selecting **Cluster Applications and Resources** → **Resource Groups** → **Change>Show Resources and Attributes for a Resource Group**, and selecting the newly created RG from the list. Press Enter.

4. Select one of the two available adaptive policies from the list by pressing F4 in *Dynamic Node Priority Policy*, as shown in Figure 10-9:

- cl_highest_udscript_rc
- cl_lowest_nonzero_udscript_rc

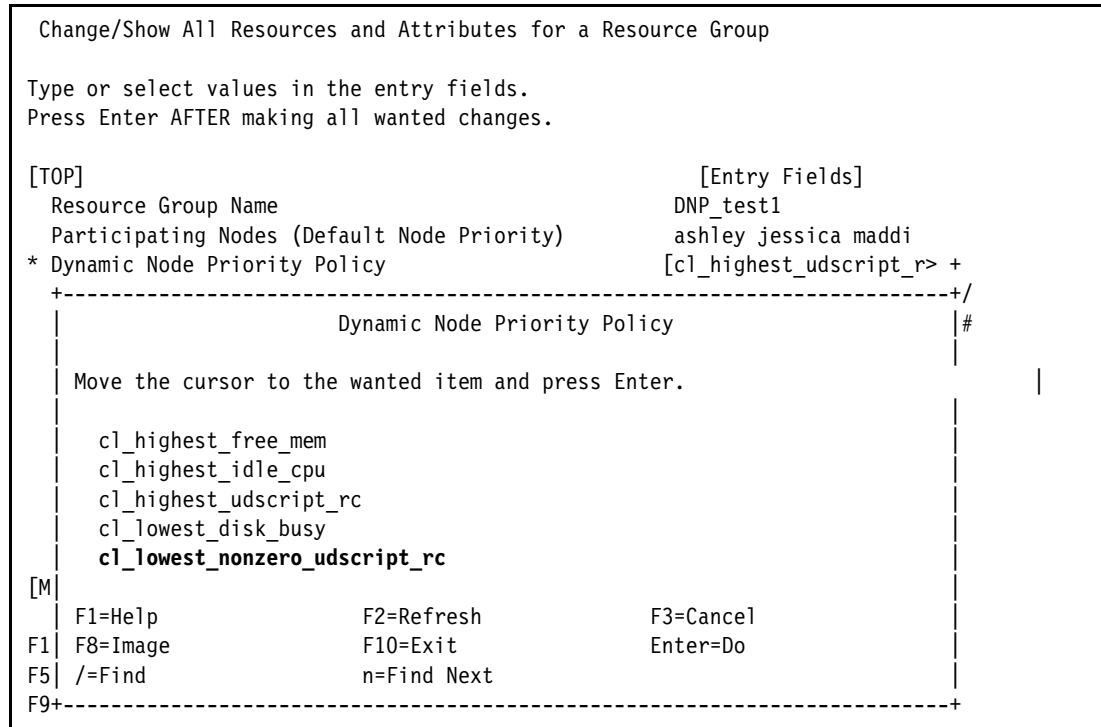


Figure 10-9 Selecting an adapter failover policy

When using one of the two user-defined policies (cl_highest_udscript_rc and cl_lowest_nonzero_udscript_rc), you must also assign a DNP Script path and DNP Script timeout value, as shown in Figure 10-10 on page 461.

Change/Show All Resources and Attributes for a Resource Group

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

[TOP]	[Entry Fields]		
Resource Group Name	DNPrg		
Participating Nodes (Default Node Priority)	ashley jessica maddi		
* Dynamic Node Priority Policy	[cl_lowest_nonzero_uds> +		
DNP Script path	[/HA727/DNP.sh] /		
DNP Script timeout value	[60] #		
Startup Policy	Online On Home Node 0>		
Failover Policy	Failover Using Dynami>		
Fallback Policy	Never Fallback>		
Fallback Timer Policy (empty is immediate)	[] +		
Service IP Labels/Addresses	[bdbsvc] +		
Application Controllers	[] +		
[MORE...28]			
F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell	F10=Exit	Enter=Do	

Figure 10-10 Assigning the DNP script path and timeout attributes to a resource group

Continue selecting the resources that will be part of the RG and press Enter.

5. Synchronize the cluster by running **clmgr sync cluster**.

You can display the current DNP policy for an existing RG, as shown in Example 10-12.

Example 10-12 Displaying a DNP policy for a resource group

```
[jessica:root] /HA727 # odmget -q group=DNPrg HACMPresource|pg
```

```
HACMPresource:
    group = "DNPrg"
    type = ""
    name = "NODE_PRIORITY_POLICY"
    value = "cl_lowest_nonzero_udscript_rc"
    ID = 1
    monitor_method = ""
HACMPresource:
    group = "DNPrg"
    type = ""
    name = "SDNP_SCRIPT_PATH"
    value = "/HA727/DNP.sh"
    ID = 2
    monitor_method = ""
HACMPresource:
    group = "DNPrg"
    type = ""
    name = "SDNP_SCRIPT_TIMEOUT"
    value = "60"
    ID = 3
```

10.5.4 Testing adaptive failover dynamic node priority

We created a 3-node cluster by using the DNP of `c1_lowest_nonzero_udscript_rc`, as shown in Example 10-12 on page 461. The contents of our `DNP.sh` scripts are shown in Example 10-13.

Example 10-13 DNP.sh script contents

```
[maddi:root] /HA727 # clcmd more /HA727/DNP.sh

-----
NODE maddi
-----
#!/bin/ksh
exit 1
-----
NODE ashley
-----
#!/bin/ksh
exit 3
-----
NODE jessica
-----
#!/bin/ksh
exit 2
```

In our test, we started the cluster and the RG came online on node `ashley`, as designated by the startup policy. The starting RG state is shown in Example 10-14.

Example 10-14 Starting resource group location

```
# c1RGinfo
-----
Group Name          Group State    Node
-----
DNPrg              ONLINE        ashley
                  OFFLINE       jessica
                  OFFLINE       maddi
```

Although our default node priority list has `jessica` listed next because we are using DNP with the *lowest* return code (RC) when a failover occurs, it falls over to node `maddi`. So, we perform a hard stop on `ashley` by running `reboot -q`, and the RG activates on node `maddi`, as shown in Example 10-15.

Example 10-15 DNP resource group location after the first failure

```
# c1RGinfo
-----
Group Name          Group State    Node
-----
DNPrg              OFFLINE       ashley
                  OFFLINE       jessica
                  ONLINE        maddi
```

Demonstration: For a demonstration about this exact failover, albeit with a different PowerHA version and node names, see this [YouTube video](#).

On successful reintegration of the original failed node (ashley), the RG remains on node maddi because the fallback policy was set to Never Fallback. We repeat the previous test by failing the current hosting node, maddi, by running `reboot -q`. This time, the node fails over to node jessica instead of node ashley because the DNP value is lower, as shown in Example 10-16.

Example 10-16 DNP resource location after a second failure

Group Name	Group State	Node
DNPrg	OFFLINE	ashley
	ONLINE	jessica
	OFFLINE	maddi

10.6 Delayed fallback timer

With this feature, you can configure the fallback behavior of an RG to occur at one of the predefined recurring times: daily, weekly, monthly, and yearly. Alternatively, you can specify a particular date and time. This feature can be useful for scheduling fallbacks to occur during off-peak business hours. Figure 10-11 shows how the delayed fallback timers can be used.

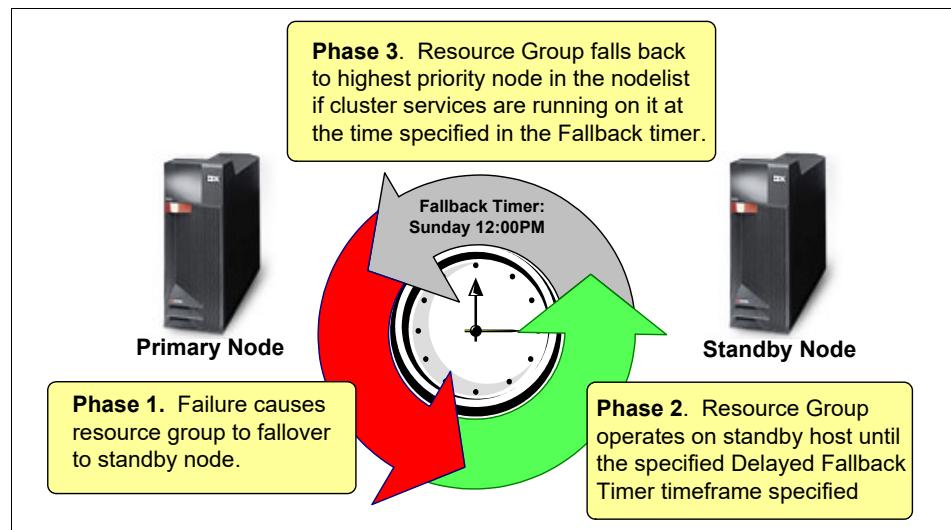


Figure 10-11 Delayed fallback timer usage

Consider a simple scenario with a cluster having two nodes and an RG. In a node failure, the RG falls over to the standby node. The RG remains on that node until the fallback timer expires. If cluster services are active on the primary node then, the RG falls back to the primary node. If the primary node is not available at that moment, the fallback timer is reset and the fallback is postponed until the fallback timer expires again.

10.6.1 Delayed fallback timer behavior

When using delayed fallback timers, consider the following items:

- ▶ The delayed fallback timer applies only to RGs that have the fallback policy set to Fallback To Higher Priority Node In The List.
- ▶ If there is no higher priority node that is available when the timer expires, the RG remains online on the current node. The timer is reset, and the fallback is retried when the timer expires again.
- ▶ If a specific date is used for a fallback timer and at that moment there is no higher priority node, the fallback will not be rescheduled.
- ▶ If an RG that is part of an Online on the Same Node dependency relationship has a fallback timer, the timer applies to all RGs that are part of the Online on the Same Node dependency relationship.
- ▶ When you use the Online on the Same Site dependency relationship and a fallback timer is used for an RG, the timer must be identical for all RGs that are part of the same dependency relationship.

10.6.2 Configuring delayed fallback timers

To configure a delayed fallback policy, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources → Resource Groups → Configure Resource Group Run-Time Policies → Configure Delayed Fallback Timer Policies → Add a Delayed Fallback Timer Policy**, and then press Enter.
2. Select one of the following options:
 - Daily
 - Weekly
 - Monthly
 - Yearly
 - Specific Date
3. Specify the following data (see Example 10-17):
 - Name of Fallback Policy
 - Specify the name of the policy by using no more than 32 characters. Use alphanumeric characters and underscores only. Do not use a leading numeric value or any reserved words.
 - Policy-specific values
 - Based on the previous selection, enter values that are suitable for the policy that is selected.

Example 10-17 Creating a fallback timer

Configure Daily Fallback Timer Policy

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

* Name of the Fallback Policy
* HOUR (0-23)
* MINUTES (0-59)

[Entry Fields]
[daily515]
[17] #
[15]

To assign a fallback timer policy to an RG, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Cluster Applications and Resources → Resource Groups → Change>Show Resources and Attributes for a Resource Group**. Select an RG from the list and press Enter.
2. Press F4 to select one of the policies that were configured in steps 1 on page 464 - 3 on page 464. The display is similar to Example 10-18.
3. Select a fallback timer policy from the list and press Enter.
4. Add any extra resources to the RG and press Enter.
5. Run verification and synchronization on the cluster to propagate the changes to all cluster nodes.

Example 10-18 Assigning a fallback timer policy to a resource group

Change>Show All Resources and Attributes for a Resource Group

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

[TOP]	[Entry Fields]
Resource Group Name	FBtimerRG
Participating Nodes (Default Node Priority)	jessica maddi ashley
Startup Policy	Online on Home Node Only
Fallover Policy	Fallover To Next Priority Node I>
Fallback Policy	Fallback To Higher Priority Node>
Fallback Timer Policy (empty is immediate)	[daily515] +
Service IP Labels/Addresses	[dallaserv] +
Application Controllers	[dummyapp] +

10.6.3 Displaying delayed fallback timers in a resource group

You can display existing fallback timer policies for RGs by using the **clshowres** command, as shown in Example 10-19.

Example 10-19 Displaying resource groups with fallback timers

```
[maddi:root] /utilities # clshowres -g FBtimerRG|egrep -i "resource group|timer"
Resource Group Name          FBtimerRG
Delayed Fallback Timer      daily515
```

An alternative is to query the HACMPTimer object class, as shown in Example 10-20.

Example 10-20 Displaying fallback timers by using ODM queries

```
[jessica:root] / # odmget HACMPTimer
```

```
HACMPTimer:
  policy_name = "daily515"
  recurrence = "daily"
  year = -3800
  month = 0
  day_of_month = 1
  week_day = 0
  hour = 17
  minutes = 30
```

Demonstration: For a demonstration of this exact scenario, see the following [YouTube video](#).

10.7 Resource group dependencies

Having large business environments that accommodate more sophisticated business solutions is common. Complex applications often contain multiple modules that rely on the availability of various resources. Highly available applications that have a multitiered architecture can use PowerHA capabilities to ensure that all required resources remain available and are started in the proper order. PowerHA can include components that are used by an application in RGs and establish RG dependencies that accurately reflect the logical relationships between application components.

For example, a database must be online before the application server is started. If the database goes down and falls over to a different node, the RG that contains the application server also is brought down and back up on any of the available cluster nodes. If the failover of the database RG is not successful, then both RGs (database and application) are put offline.

To understand how PowerHA can be used to ensure high availability (HA) of multitiered applications, understand the following concepts:

- ▶ Parent RG

The parent RG is the first RG to be acquired during the RG acquisition. This RG does not have any other RG as a prerequisite. Here, you should include application components or modules that *do not rely* on the presence of other components or modules.

- ▶ Child RG

A child RG depends on a parent RG. This type of RG assumes the existence of another RG. Here, you should include application components or modules that *do rely* on the availability of other components or modules.

A child RG cannot and will not be brought online unless the parent RG is online. If the parent RG is put offline, the child RG will also be put offline.

- ▶ Parent-child dependency

A parent-child dependency allows binding RGs in a hierarchical manner. There can be only three levels of dependency for RGs. An RG can act both as a parent and a child. You cannot specify circular dependencies among RGs. You can also configure a location dependency between RGs to control the colocation of your RGs.

- ▶ Location dependency

RG location dependency gives you the means to ensure that certain RGs always will be online on the same node or site, or that certain RGs always will be Online On Different Nodes or sites.

- ▶ Start after dependency

This dependency means that the target RG must be online on any node in the cluster before a source (dependent) RG can be activated on a node. There is no dependency when releasing RGs and the groups are released in parallel.

- ▶ Stop after dependency

In this type of dependency, the target RG must be offline on any node in the cluster before a source (dependent) RG can be brought offline on a node. There is no dependency when acquiring RGs, and the groups are acquired in parallel.

10.7.1 Resource group parent-child dependency

You can configure parent-child dependencies between RGs to ensure that RGs are processed properly during cluster events.

Planning for parent-child resource group dependencies

When you plan to use parent-child RG dependencies, consider these factors:

- ▶ Carefully plan which RGs will contain which application component. Ensure that application components that rely on the availability of other components are placed in various RGs. The RG parent-child relationship should reflect the logical dependency between application components.
- ▶ A parent-child relationship can span up to three levels.
- ▶ No circular dependencies should exist between RGs.
- ▶ An RG can act as a parent for an RG and as a child for another RG.
- ▶ Plan for application monitors for each application that you are planning to include in a child or parent RG.
- ▶ For an application in a parent RG, configure a monitor in the monitoring startup mode. After the parent RG is online, the child RGs also is brought online.

Configuring a resource group parent-child dependency

To configure parent-child RG dependency, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Configure Resource Group Run-Time Policies** → **Configure Dependencies between Resource Groups** → **Configure Parent/Child Dependency** → **Add Parent/Child Dependency between Resource Groups**, and press Enter.
2. Complete the fields as follows:
 - Parent Resource Group
Select the parent RG from the list. During RG acquisition, the parent RG is brought online before the child RG.
 - Child Resource Group
Select the child RG from the list and press Enter. During RG release, PowerHA takes the child RG offline before the parent RG. PowerHA prevents you from specifying a circular dependency.
3. Synchronize the cluster by running **clmgr sync cluster**.

10.7.2 Resource group location dependency

You can configure location dependencies between RGs to control the location of RGs during cluster events. With PowerHA, you can configure the following types of RG location dependencies:

- ▶ Online on the Same Node dependency
- ▶ Online on the Same Site dependency
- ▶ Online on Different Nodes dependency

You can combine RG parent-child and location dependencies.

Planning for an Online on the Same Node dependency

When you plan to use Online on the Same Node dependencies, consider these factors:

- ▶ All RGs that have an Online on the Same Node dependency relationship must have the same node list and the participating nodes must be listed in the same order.
- ▶ Both concurrent and nonconcurrent RGs are allowed.
- ▶ You can have more than one Online on the Same Node dependency relationship in the cluster.
- ▶ All nonconcurrent RGs in the same Online on the Same Node dependency relationship must have identical startup, failover, and fallback policies:
 - Online Using Node Distribution Policy is not allowed as a startup policy.
 - If Dynamic Node Priority Policy is used as the failover policy, all RGs in the dependency must use the same DNP policy.
 - If one RG has a fallback timer that is configured, the timer also applies to the RGs that take part in the dependency relationship. All RGs must have an identical fallback time setting.
 - If one or more RGs in the Online on the Same Node dependency relationship fail, cluster services try to place all RGs on the node that can accommodate all RGs that are online plus one or more failed RGs.

Configuring an Online on the Same Node location dependency

To configure an Online on the Same Node RG dependency, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Configure Resource Group Run-Time Policies** → **Configure Dependencies between Resource Groups** → **Configure Online on the Same node Dependency** → **Add Online on the Same Node Dependency Between Resource Groups**, and select the RGs that will be part of that dependency relationship.

To have RGs activated on the same node, they must have identical participating node lists.

2. Propagate the change across all cluster nodes by verifying and synchronizing your cluster.

Planning for an Online On Different Nodes dependency

When you configure RGs in an Online On Different Nodes dependency relationship, you assign priorities to each RG in case there is contention for a particular node at any point. You can assign High, Intermediate, and Low priorities. Higher priority RGs take precedence over lower priority RGs on startup, failover, and fallback.

When you plan to use Online on Different Nodes dependencies, consider these factors:

- ▶ Only one Online On Different Nodes dependency is allowed per cluster.
- ▶ Each RG must have a different home node for startup.

- ▶ When using this policy, a higher priority RG takes precedence over a lower priority RG during startup, failover, and fallback:
 - If an RG with High priority is online on a node, no other RG that is part of the Online On Different Nodes dependency can be put online on that node.
 - If an RG that is part of the Online On Different Nodes dependency is online on a cluster node and an RG that is part of the Online On Different Nodes dependency and has a higher priority falls over or falls back to the same cluster node, the RG with a higher priority is brought online. The RG with a lower priority RG is taken offline or migrated to another cluster node if available.
 - RGs that are part of the Online On Different Nodes dependency and have the same priority cannot be brought online on the same cluster node. The precedence of RGs that are part of the Online On Different Nodes dependency and have the same priority is determined by alphabetical order.
 - RGs that are part of the Online On Different Nodes dependency and have the same priority do not cause each other to be moved from a cluster node after a failover or fallback.
 - If a parent-child dependency is used, the child RG cannot have a priority higher than its parent.

Configuring an Online on Different Node location dependency

To configure an Online on Different Node RG dependency, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Configure Resource Group Run-Time Policies** → **Configure Dependencies between Resource Groups** → **Configure Online on Different Nodes Dependency** → **Add Online on Different node Dependency between Resource Groups**, and press Enter.
2. Complete the following fields and press Enter:
 - High Priority Resource Groups

Select the RGs that will be part of the Online On Different Nodes dependency and should be acquired and brought online *before* all other RGs.

On fallback and failover, these RGs are processed simultaneously and brought online on different cluster nodes before any other RGs. If different cluster nodes are unavailable for failover or fallback, then these RGs, having the same priority level, can remain on the same node.

The highest relative priority within this set is the RG listed first.
 - Intermediate Priority Resource Groups

Select the RGs that will be part of the Online On Different Nodes dependency and should be acquired and brought online *after* High priority RGs and *before* Low-priority RGs.

On fallback and failover, these RGs are processed simultaneously and brought online on different target nodes before Low-priority RGs. If different target nodes are unavailable for failover or fallback, these RGs, having the same priority level, can remain on the same node. The highest relative priority within this set is the RG that is listed first.

- Low-Priority Resource Groups

Select the RGs that will be part of the Online On Different Nodes dependency and that should be acquired and brought online *after* all other RGs. On fallback and failover, these RGs are brought online on different target nodes after the all higher priority RGs are processed.

Higher priority RGs moving to a cluster node can cause these RGs to be moved to another cluster node or be taken offline.

3. Continue configuring runtime policies for other RGs or verify and synchronize the cluster by running `clmgr sync cluster`.

Planning for an Online on the Same Site dependency

When you plan to use Online on the Same site dependencies, consider these factors:

- ▶ All RGs in an Online on the Same Site dependency relationship must have the same Inter-Site Management policy. However, they might have different startup, failover, and fallback policies. If fallback timers are used, they must be identical for all RGs that are part of the Online on the Same Site dependency.
- ▶ The fallback timer does not apply to moving an RG across site boundaries.
- ▶ All RGs in an Online on the Same Site dependency relationship must be configured so that the nodes that can own the RGs are assigned to the same primary and secondary sites.
- ▶ The Online Using Node Distribution policy is supported.
- ▶ Both concurrent and nonconcurrent RGs are allowed.
- ▶ You can have more than one Online on the Same Site dependency relationship in the cluster.
- ▶ All RGs that have an Online on the Same Site dependency relationship are required to be on the same site, although some of them might be in an OFFLINE or ERROR state.
- ▶ If you add an RG that is part of an Online on the Same Node dependency to an Online on the Same Site dependency, you must add all other RGs that are part of the Online on the Same Node dependency to the Online on the Same Site dependency.

Configuring an Online on the Same Site Location dependency

To configure an Online on the Same Site RG dependency, complete the following steps:

1. Use the `smitty sysmirror` fast path, select **Cluster Applications and Resources** → **Resource Groups** → **Configure Resource Group Run-Time Policies** → **Configure Dependencies between Resource Groups** → **Configure Online on the Same Site Dependency** → **Add Online on the Same Site Dependency Between Resource Groups**, and press Enter.
2. Select from the list the RGs to be put online on the same site. During acquisition, these RGs are brought online on the same site according to the site and the specified node startup policy for the RGs. On fallback or failover, the RGs are processed simultaneously and brought online on the same site.
3. Synchronize the cluster by running `clmgr sync cluster`.

10.7.3 Start After and Stop After dependencies

Dependency policies also include a Start After dependency and a Stop After dependency.

Start After dependency

In this type of dependency, the target RG must be online on any node in the cluster before a source (dependent) RG can be activated on a node. There is no dependency when releasing RGs and the groups are released in parallel.

Here are the guidelines and limitations:

- ▶ An RG can serve as both a target and a source RG, depending on which end of a dependency link that it is placed.
- ▶ You can specify three levels of dependencies for RGs.
- ▶ You cannot specify circular dependencies between RGs.
- ▶ This dependency applies only at the time of RG acquisition. There is no dependency between these RGs during RG release.
- ▶ A source RG cannot be acquired on a node until its target RG is fully functional. If the target RG does not become fully functional, the source RG goes into an OFFLINE DUE TO TARGET OFFLINE state. If you notice that an RG is in this state, you might need to troubleshoot which resources might need to be brought online manually to resolve the RG dependency.
- ▶ When an RG in a target role falls over from one node to another node, there will be no effect on the RGs that depend on it.
- ▶ After the source RG is online, any operation (bring offline, move RG) on the target RG will not affect the source RG.
- ▶ A manual RG move or bring RG online on the source RG is not allowed if the target RG is offline.

To configure a Start After RG dependency, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Cluster Applications and Resources** → **Resource Groups** → **Configure Resource Group Run-Time Policies** → **Configure Dependencies between Resource Groups** → **Configure Start After Resource Group Dependency** → **Add Start After Resource Group Dependency**.
2. Choose the appropriate RG to complete each field:
 - Source Resource Group
Select the source RG from the list and press Enter. PowerHA SystemMirror prevents you from specifying circular dependencies. The source RG depends on services that another RG provides. During RG acquisition, PowerHA SystemMirror acquires the target RG on a node before the source RG is acquired.
 - Target Resource Group
Select the target RG from the list and press Enter. PowerHA SystemMirror prevents you from specifying circular dependencies. The target RG provides services that another RG depends on. During RG acquisition, PowerHA SystemMirror acquires the target RG on a node before the source RG is acquired. There is no dependency between source and target RGs during release.

Stop After dependency

In this type of dependency, the target RG must be offline on any node in the cluster before a source (dependent) RG can be brought offline on a node. There is no dependency when acquiring RGs, and the groups are acquired in parallel.

Here are the guidelines and limitations:

- ▶ An RG can serve as both a target and a source RG, depending on which end of a dependency link it is placed.
- ▶ You can specify three levels of dependencies for RGs.
- ▶ You cannot specify circular dependencies between RGs.
- ▶ This dependency applies only at the time of RG release. There is no dependency between these RGs during RG acquisition.
- ▶ A source RG cannot be released on a node until its target RG is offline.
- ▶ When an RG in a source role falls over from one node to another node, first the target RG is released and then the source RG is released. Afterward, both RGs are acquired in parallel, assuming that there is no start after or parent-child dependency between these RGs.
- ▶ A manual RG move or bringing an RG offline on the source RG is not allowed if the target RG is online.

To configure a Stop After RG dependency, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Cluster Applications and Resources → Resource Groups → Configure Resource Group Run-Time Policies → Configure Dependencies between Resource Groups → Configure Stop After Resource Group Dependency → Add Start After Resource Group Dependency**.
2. Choose the appropriate RG to complete each field:
 - Source Resource Group: Select the source RG from the list and press Enter. PowerHA SystemMirror prevents you from specifying circular dependencies. The source RG stops only after the target RG is offline. During the RG release process, PowerHA SystemMirror releases the target RG on a node before releasing the source RG. There is no dependency between source and target RGs during acquisition.
 - Target Resource Group: Select the target RG from the list and press Enter. PowerHA SystemMirror prevents you from specifying circular dependencies. The target RG provides services on which another RG depends. During the RG release process, PowerHA SystemMirror releases the target RG on a node before releasing the source RG. There is no dependency between source and target RGs during acquisition.

10.7.4 Combining various dependency relationships

When combining multiple dependency relationships, consider the following information:

- ▶ Only one RG can belong to both an Online on the Same Node dependency relationship and an Online on Different Nodes dependency relationship.
- ▶ If an RG belongs to both an Online on the Same Node dependency relationship and an Online on Different Node dependency relationship, then all other RGs than are part of the Online of Same Node dependency have the same priority as the common RG.
- ▶ Only RGs with the same priority that are part of an Online on Different Nodes dependency relationship can be part of an Online on the Same Site dependency relationship.

10.7.5 Displaying resource group dependencies

You can display RG dependencies by using the `clrgdependency` command, as shown in Example 10-21.

Example 10-21 Displaying resource group dependencies

```
[jessica:root] / #clrgdependency -t PARENT_CHILD -s1
# Parent           Child
rg_parent          rg_child
```

An alternative approach is to query the `HACMPrg_loc_dependency` and `HACMPrgdependency` object classes, as shown in Example 10-22.

Example 10-22 Displaying resource group dependencies by using ODM queries

```
[jessica:root] / # odmget HACMPrgdependency

HACMPrgdependency:
  ID = 0
  group_parent = "rg_parent"
  group_child = "rg_child"
  dependency_type = "PARENT_CHILD"
  dep_type = 0
  group_name = ""
root@maddi [] odmget HACMPrg_loc_dependency

HACMPrg_loc_dependency:
  ID = 1
  set_id = 1
  group_name = "rg_same_node2"
  priority = 0
  loc_dep_type = "NODECOLLOCATION"
  loc_dep_sub_type = "STRICT"

HACMPrg_loc_dependency:
  ID = 2
  set_id = 1
  group_name = "rg_same_node_1"
  priority = 0
  loc_dep_type = "NODECOLLOCATION"
  loc_dep_sub_type = "STRICT"

HACMPrg_loc_dependency:
  ID = 4
  set_id = 2
  group_name = "rg_different_node1"
  priority = 1
  loc_dep_type = "ANTICOLLOCATION"
  loc_dep_sub_type = "STRICT"

HACMPrg_loc_dependency:
  ID = 5
  set_id = 2
  group_name = "rg_different_node2"
  priority = 2
```

```
loc_dep_type = "ANTICOLLOCATION"
loc_dep_sub_type = "STRICT"
```

Note: Using information that is retrieved directly from the ODM is for informational purposes only because the format within the stanzas might change with updates or new versions.

Hardcoding ODM queries within user-defined applications is not supported and should be avoided.

10.7.6 Resource group dependency scenario

In the following example, we combine both the parent-child dependency along with Online On Different Nodes. We have a 3- and 3-RG configuration, as shown in Example 10-23. This combination requires the startup policy to be Online On First Available.

Example 10-23 Three resource groups base configuration

Resource Group Name	PCrg1
Participating Node Names	jessica ashley maddi
Startup Policy	Online on First Available Node
Fallover Policy	Fallover To Next Priority Node
In The List	
Fallback Policy	Never Fallback
Resource Group Name	PCrg2
Participating Node Names	jessica ashley maddi
Startup Policy	Online on First Available Node
Fallover Policy	Fallover To Next Priority Node
In The List	
Fallback Policy	Never Fallback
Resource Group Name	PCrg3
Participating Node Names	jessica ashley maddi
Startup Policy	Online on First Available Node
Fallover Policy	Fallover To Next Priority Node
In The List	
Fallback Policy	Never Fallback

We configure the first parent RG as PCrg3, and the child as PCrg2. We also make another parent RG of PCrg2 and the child as PCrg1. The PCrg2 RG is *both* a parent and child RG, as shown in Example 10-24. This relationship is a three-level nested one.

Example 10-24 Parent-child relationships among three resource groups

```
# clrgdependency -t PARENT_CHILD -sl
#Parent           Child
PCrg3            PCrg2
PCrg2            PCrg1
```

When using the Online On Different Nodes dependency, you must assign a priority to each RG. With this combination to prevent any RG from becoming orphaned because of a failure, there should be one more node than there are RGs. The priorities are shown in Figure 10-12 on page 475.

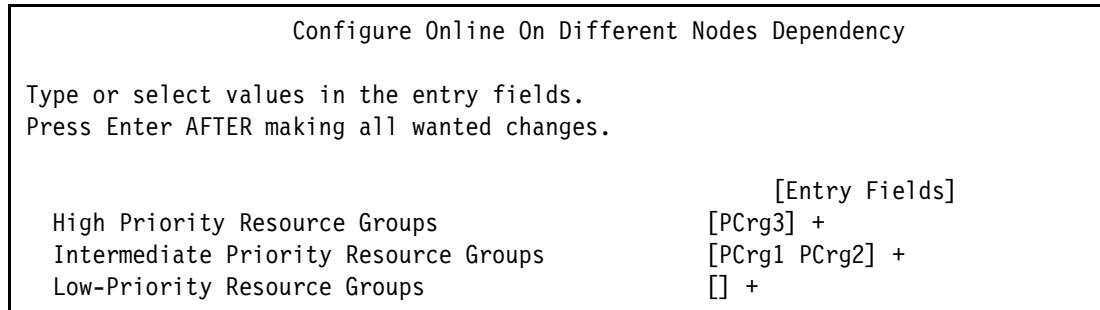


Figure 10-12 Resource group priorities

We start node *jessica* first, and because PCrg3 is the main parent RG, it comes online. Then, we start node *ashley* and it acquires PCrg2 RG because it is the child of PCrg3 and the parent of PCrg1. Then, we start node *maddi* last, and it acquires PCrg1. All three of these states are shown in Example 10-25.

Example 10-25 Resource group locations after the cluster startup

```
# c1RGinfo
-----
Group Name          Group State    Node
-----
PCrg1              OFFLINE       ashley
                  OFFLINE due to p jessica
                  OFFLINE       maddi

PCrg2              OFFLINE       ashley
                  OFFLINE due to l jessica
                  OFFLINE       maddi

PCrg3              OFFLINE       ashley
                  ONLINE        jessica
                  OFFLINE       maddi

# c1RGinfo
-----
Group Name          Group State    Node
-----
PCrg1              OFFLINE due to l ashley
                  OFFLINE due to l jessica
                  OFFLINE       maddi

PCrg2              ONLINE        ashley
                  OFFLINE       jessica
                  OFFLINE       maddi

PCrg3              OFFLINE       ashley
                  ONLINE        jessica
                  OFFLINE       maddi

# c1RGinfo
-----
```

PCrg1	OFFLINE OFFLINE ONLINE	ashley jessica maddi
PCrg2	ONLINE OFFLINE OFFLINE	ashley jessica maddi
PCrg3	OFFLINE ONLINE OFFLINE	ashley jessica maddi

When you fail node jessica by running **reboot -q**, there is a cascading effect because of the combination of dependencies that is used. Essentially, *all* RGs come offline at some point. PCrg1 is taken offline because it is both the lowest child and has the lowest priority. PCrg2 temporarily comes offline because it is a child of PCrg3. PCrg3 is acquired by node ashley, which was previously hosting the lowest RG of PCrg1. Then, PCrg2 is reacquired by node maddi. Although the restart of PCrg2 is not clearly depicted in the output, the result is shown Example 10-26.

Example 10-26 Initial failure results with dependencies

# clRGinfo	-----	
Group Name	Group State	Node
PCrg1	OFFLINE due to l OFFLINE OFFLINE due to l	ashley jessica maddi
PCrg2	OFFLINE OFFLINE ONLINE	ashley jessica maddi
PCrg3	ONLINE OFFLINE OFFLINE	ashley jessica maddi

To expand on the test scenario, we restart cluster services on jessica and acquired the previously orphaned RG PCrg1, as shown in Example 10-27.

Example 10-27 Resource group state after reintegration

# clRGinfo	-----	
Group Name	Group State	Node
PCrg1	OFFLINE ONLINE OFFLINE	ashley jessica maddi
PCrg2	OFFLINE OFFLINE ONLINE	ashley jessica maddi
PCrg3	ONLINE OFFLINE OFFLINE	ashley jessica maddi

We now fail node *maddi*, which results in PCrg1 going offline. PCrg2 is acquired by node *jessica*, and PCrg3 is left in place, as shown in Example 10-28.

Example 10-28 Second failure results with dependencies

#	c1RGinfo	
Group Name	Group State	Node
PCrg1	OFFLINE due to 1	glvm3
	OFFLINE due to 1	jessica
	OFFLINE	maddi
PCrg2	OFFLINE	glvm3
	ONLINE	jessica
	OFFLINE	maddi
PCrg3	ONLINE	glvm3
	OFFLINE	jessica
	OFFLINE	maddi



Customizing resources and events

In this chapter, we show how you can use PowerHA to recognize and react to cluster events. PowerHA has features to help you modify and adjust cluster behavior in response to specific events according to the requirements of your environment.

This chapter contains the following topics:

- ▶ Overview of cluster events
- ▶ System events
- ▶ User-defined resources and types
- ▶ Writing scripts for custom events
- ▶ Pre-event and post-event commands
- ▶ Automatic error notification

11.1 Overview of cluster events

When a cluster event occurs, the Cluster Manager runs the event script that corresponds to that event. While the event script is processed, a series of sub-event scripts might be run. PowerHA provides a script for each event and sub-event. The default scripts are in the /usr/es/sbin/cluster/events directory. By default, the Cluster Manager calls the corresponding event script for a specific event. You can specify extra specific actions to perform when an event occurs. You can customize the handling of an event for your cluster by using the following features:

- ▶ Pre-event and post-event processing

You can customize event processing according to the requirements of your environment by specifying commands or user-defined scripts that are run before or after a specific event is run by the Cluster Manager.

- ▶ Event notification

You can specify a command or a user-defined script to provide notification that an event is about to happen or occurred. This command is run once before processing the event and again as the last step of event processing.

- ▶ Event recovery and retry

You can specify a command that attempts to recover from an event failure. This command is run only if the script event fails. After the recovery command runs, the event script runs again. You can also specify a counter that represents the maximum number of times that the cluster event can fail. If the cluster script still fails after the last attempt, then the cluster manager declares the failure of that event.

- ▶ Cluster automatic error notification

You can use an AIX error logging feature to detect hardware and software errors that by default are not monitored by cluster services and trigger an appropriate response action.

- ▶ Customizing event duration

PowerHA issues a warning each time a cluster event takes more time to complete than a specified timeout period. You can customize the time of the period for a cluster event to complete before a warning is issued.

- ▶ Defining new custom events

You can define new custom cluster events.

11.2 System events

PowerHA SystemMirror 7.1 introduced system events. These events are handled by a subsystem that is named clevmgrdES. The rootvg system event allows for the monitoring of any loss of access to the rootvg volume group (VG). By default, in a loss of access, the event logs an entry in the system error log and restarts the system. The option can be changed in the System Management Interface Tool (SMIT) menu to log only an event entry and not restart the system.

To check or change the response to losing access to rootvg, complete the following steps:

1. Enter **smitty sysmirror**, select **Custom Cluster Configuration → Events → System Events → Change>Show Event Response**, select ROOTVG from the list, and press Enter.
2. Highlight Response and press F4 to open the following list:

Log event and reboot	As the description implies, this response both logs the event and initiates a node restart. It is the default value.
Only log the event	This response logs the event only as implied.
3. Highlight Active and press F4 to open the following list:

Yes	The system event is monitored and reacts based on the response criteria that is selected. This response is the default value, and it is considered a best practice.
No	The system event is not monitored, and the response that is selected is irrelevant.
4. On completing the selections that are shown in Figure 11-1, press Enter.
5. Synchronize the cluster by running **clmgr sync cluster**.

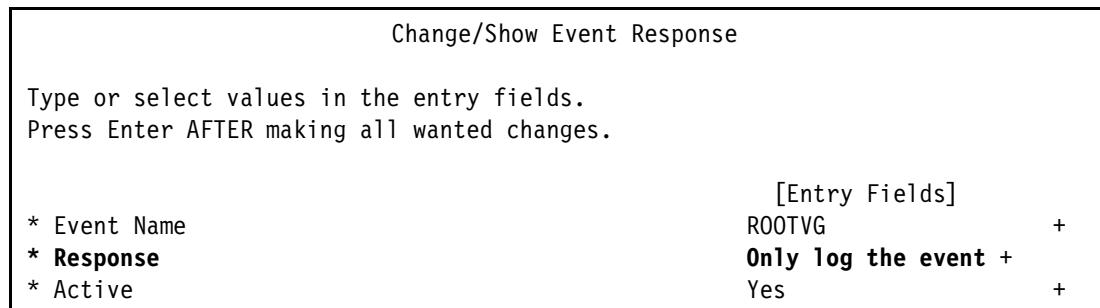


Figure 11-1 The rootvg system event

11.3 User-defined resources and types

This option was introduced in PowerHA 7.1 so that users can add their own resource type and specify how and where PowerHA processes the resource type. Using this feature involves the following high-level steps:

1. Create a user-defined resource type.
2. Assign the new resource type to a resource.
3. Add the user-defined resource into a resource group (RG).

Note: The options of when and where to process the resource are not as granular as using custom events. However, they are suitable for most requirements.

11.3.1 Creating a user-defined resource type

To create a user-defined resource type, use these steps:

1. Run **smitty sysmirror** and select **Custom Cluster Configuration → Resources → Configure User Defined Resources and Types → Configure User-Defined Resource Types → Add a User-Defined Resource Type**.
2. Complete the fields in the Add a User-Defined Resource Type panel. Figure 11-2 shows an example. After creating the resource type, add it to a resource.

Add a User-Defined Resource Type

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

* Resource Type Name	[Entry Fields] [specialrestype]
* Processing Order	[FIRST] +
Verification Method	[]
Verification Type	[Script] +
* Start Method	[/HA727/custom.sh start]
* Stop Method	[/HA727/custom.sh stop]
Monitor Method	[]
Cleanup Method	[]
Restart Method	[]
Failure Notification Method	[]
Required Attributes	[]
Optional Attributes	[]
Description	[Test for Redbooks]

Figure 11-2 Creating a user-defined resource type

The fields are as follows:

► **Resource Type Name**

The symbolic name for a user-defined resource type. You can enter ASCII text up to 64 characters, including alphanumeric characters and underscores.

► **Processing Order**

Specify the processing order that you want to use to process the user-defined resources. Press F4 to view a list of all existing resource types, and select one from the list:

FIRST
WPAR
VOLUME_GROUP
FILE_SYSTEM
SERVICEIP
TAPE
APPLICATION

PowerHA SystemMirror processes the user-defined resources at the beginning of the resource acquisition order if you choose FIRST. If you select any other value, for example, VOLUME_GROUP, the user-defined resources are acquired after varying on the VGs, and they are released after varying off the VGs.

► **Verification Method**

Specify a verification method that is invoked by the cluster verification process. Provide the verification checks so that before you start the cluster services, the user-defined resources are verified to avoid failures during cluster operation.

► **Verification Type**

Specify the type of verification method to use. The verification method can be either a script or a library. Select Script if the type of verification method that you want to use is a script. Select Library if the type of verification method that you want to use is a library that was built with an application programming interface (API).

► **Start Method**

Enter the name of the script and its full path name (followed by arguments) that is called by the cluster event scripts to start the user-defined resource. Use a maximum of 256 characters. This script must be in the same location on each cluster node that might start the server. However, the contents of the script might differ.

► **Stop Method**

Enter the full path name of the script that is called by the cluster event scripts to stop the user-defined resource. Use a maximum of 256 characters. This script must be in the same location on each cluster node that might stop the resource. However, the contents of the script might differ.

► **Monitor Method**

Enter the full path name of the script that is called by the cluster event scripts to monitor the user-defined resource. Use a maximum of 256 characters. This script must be in the same location on each cluster node that might monitor the monitor. However, the contents of the script might differ.

► **Cleanup Method**

Optional: Specify a resource cleanup script to call when a failed user-defined resource is detected before calling the restart method. The default for the cleanup script is the stop script that is defined when the user-defined resource type was set up. If you are changing the monitor mode to use only in the startup monitoring mode, the method that is specified in this field does not apply, and PowerHA SystemMirror ignores values that are entered in this field.

Note: With monitoring, the resource stop script might fail because the resource is already stopped when this script is called.

► **Restart Method**

The default restart method is the resource start script that was defined when the application controller was set up. You can specify a different method here. If you change the monitor mode to be used only in the startup monitoring mode, the method that is specified in this field does not apply, and PowerHA SystemMirror ignores values that are entered in this field.

► **Failure Notification Method**

Define a notify method to run when the user-defined resource fails. This custom method runs during the restart process and during notify activity. If you are changing the monitor mode to be used only in the startup monitoring mode, the method that is specified in this field does not apply, and PowerHA SystemMirror ignores values that are entered in this field.

- ▶ Required Attributes

Specify a list of attribute names, with each name separated by a comma. These attributes must be assigned with values when you create the user-defined resource, for example, Rattr1,Rattr2. The purpose of the attributes is to store resource-specific attributes, which can be used in the different methods that are specified in the resource type configuration.

- ▶ Optional Attributes

Specify a list of attribute names, with each name separated by a comma. These attributes might be assigned with values when creating the user-defined resource, for example, Oattr1,Oattr2. The purpose of the attributes is to store resource-specific attributes, which can be used in the different methods that are specified in the resource type configuration.

- ▶ Description

Provide a description of the user-defined resource type.

11.3.2 Creating a user-defined resource

To create a user-defined resource, complete the following steps:

1. Run **smitty sysmirror** and select **Custom Cluster Configuration → Resources → Configure User-Defined Resources and Types → Configure User Defined Resource → Add a User-Defined Resource**.
2. Choose a previously defined resource type from the list.
3. Complete the remaining fields. An example is shown in Figure 11-3.

Add a User-Defined Resource

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

* Resource Type Name	[Entry Fields]
* Resource Name	specialrestype
Attribute data	[shawnsresource]
	[]

Figure 11-3 Creating a resource for a previously created resource type

The fields are as follows:

- ▶ Resource Type Name

This name was chosen from the menu of previously created resource types.

- ▶ Resource Name

Enter an ASCII text string that identifies the resource. You use this name to refer to the resource when you define resources during node configuration. The resource name can include alphanumeric characters and underscores. A maximum of 64 characters is allowed.

Note: The resource name must be unique across the cluster. When you define a VG as a user-defined resource for a Peer-to-Peer Remote Copy (PPRC) configuration or a HyperSwap configuration, the resource name must match the VG.

- ▶ Attribute data

Specify a list of attributes and values in the form of attribute=value, with each pair separated by a space, as in the following example:

```
Rattr1="value1" Rattr2="value2" Oattr1="value3"
```

When you are done, you must add the resource to an RG for it to be used.

11.3.3 Adding a user-defined resource to a resource group

To create a user-defined RG, complete the following steps:

1. Run **smitty sysmirror** and select **Cluster Applications and Resources → Resources Groups → Change>Show Resources and Attributes for a Resource Group**.
2. Choose an RG from the list.
3. Scroll down to the bottom until you find and select User-Defined Resources.
4. Press F4 to generate a list of previously created resources. Scroll to the one you want and press Enter. An example is shown in Figure 11-4.

Change/Show All Resources and Attributes for a Resource Group		
Type or select values in the entry fields. Press Enter AFTER making all wanted changes.		
[MORE...28]	[Entry Fields]	
Network For NFS Mount	[]	+
Tape Resources	[]	+
Raw Disk PVIDs	[]	+
Raw Disk UUIDs/hdisks	[]	+
Disk Error Management?	no	+
Primary Workload Manager Class	[]	+
Secondary Workload Manager Class	[]	+
Miscellaneous Data	[]	+
WPAR Name	[]	+
User-Defined Resources	[shawnsresource]	+

Figure 11-4 Adding a user-defined resource into a resource group

5. On completion, synchronize the cluster for the new resource to be used by running **clmgr sync cluster**.

11.4 Writing scripts for custom events

Customizing cluster events requires writing scripts. Consider these best practices:

- ▶ Test all possible input parameters.
- ▶ Test all conditional branches, for example, all “if” and “case” branches.
- ▶ Handle all error (nonzero) exit codes.
- ▶ Provide a correct return value: 0 for success, and any other number for failure.
- ▶ Terminate within a reasonable amount of time.
- ▶ Test the scripts thoroughly because they can affect the behavior of your cluster. Consider that if your script fails, your cluster fails too.
- ▶ Be sure that a recovery program can recover from an event failure, otherwise the cluster fails.
- ▶ Store your scripts in a convenient location.
- ▶ The name and location of scripts must be identical on all cluster nodes. However, the content of the scripts might be different.
- ▶ Thoroughly document your scripts.
- ▶ Set the execute bit for all scripts.
- ▶ Synchronization does not copy pre-event and post-event script content from one node to another node. Copy pre-event and post-event scripts on all cluster nodes. You can also use file collections to keep them in sync, assuming that the scripts are identical on each node.

Important: The cluster will not continue processing events until the custom pre-event or post-event script finishes running. If a problem with the scripts are encountered, it can lead to a CONFIG_TOO_LONG, and an RG ERROR state.

11.5 Pre-event and post-event commands

For all predefined events, you can define a pre-event, a post-event, a notification method, and a recovery command:

- ▶ Pre-event script
 - This script runs *before* the cluster event is run.
- ▶ Post-event script
 - This script runs *after* the cluster event is run.
- ▶ Notify method
 - The notification method runs before and after the cluster event. It usually sends a message to the system administrator about an event starting or completing.

11.5.1 Parallel processed resource groups and pre-event and post-event scripts

RGs, by default, are processed in parallel unless you specify a customized serial processing order for all or some of the RGs in the cluster. When RGs are processed in parallel, fewer cluster events occur in the cluster, and the number of particular cluster events for which you can create customized pre-event or post-event scripts is reduced.

Only the following events *occur* during parallel processing of RGs:

- ▶ node_up
- ▶ node_down
- ▶ acquire_svc_addr
- ▶ acquire_takeover_addr
- ▶ release_svc_addr
- ▶ release_takeover_addr
- ▶ start_server
- ▶ stop_server

The following events *do not occur* during parallel processing of RGs:

- ▶ get_disk_vg_fs
- ▶ release_vg_fs
- ▶ node_up_local
- ▶ node_up_remote
- ▶ node_down_local
- ▶ node_down_remote
- ▶ node_up_local_complete
- ▶ node_up_remote_complete
- ▶ node_down_local_complete
- ▶ node_down_remote_complete

Always be attentive to the list of events when you upgrade from an older version and choose parallel processing for some of the pre-existing RGs in your configuration.

Note: When you try to adjust the default behavior of an event script, always use pre-event or post-event scripts. Do not modify the built-in event script files. This option is not supported or safe because these files can be modified without notice when applying fixes or performing upgrades.

11.5.2 Configuring pre-event or post-event scripts

You can define multiple customized pre-event and post-event scripts for a cluster event.

To define a pre-event or post-event script, create a custom event and then associate the custom event with a cluster event as follows:

1. Write and test your event script carefully. Ensure that you copy the file to all cluster nodes under the same path and name.
2. Define the custom event:
 - a. Run `smitty sysmirror` fast path and select **Custom Cluster Configuration** → **Events** → **Cluster Events** → **Configure Pre/Post-Event Commands** → **Add a Custom Cluster Event**.

- b. Complete the following information:
 - Cluster Event Name: The name of the event.
 - Cluster Event Description: A short description of the event.
 - Cluster Event Script Filename: The full path of the event script.
3. Connect the custom event with the pre/post-event cluster event:
 - a. Run `smitty sysmirror` fast path and select **Custom Cluster Configuration** → **Events** → **Cluster Events** → **Change>Show Pre-Defined Events**.
 - b. Select the event that you want to adjust.
 - c. Enter the following values:
 - Notify Command (optional): The full path name of the notification command, if any.
 - Pre-event Command (optional): The name of a previously created custom cluster event that you want to run as a pre-event. You can choose from the custom cluster event list that was previously defined.
 - Post-event Command (optional): The name of a previous custom cluster event that you want to run as a post-event. You can choose from the custom cluster event list that was previously defined.
 - Fail event if pre- or post-event fails (Yes or No): By default, the exit status that is returned by these commands is ignored and does not affect the exit status of the main event. If you select yes for this option, any nonzero exit status from a pre- or post-event command is treated like a failure of the main event. Furthermore, if the pre-event command fails, the main event and post events are not called. If the main event fails, the post event is not called. In all cases, the notify command is called after a failure.
4. Verify and synchronize the cluster by running `clmgr sync cluster`.

Tips:

- ▶ You can use cluster file collection feature to ensure that custom event files are propagated automatically to all cluster nodes.
- ▶ If you use pre-event and post-event scripts to ensure proper sequencing and correlation of resources that are used by applications running on the cluster, consider simplifying or eliminating them by specifying parent-child dependencies between RGs.

11.6 Automatic error notification

By default, PowerHA monitors only cluster nodes, networks, and network adapters. However, in your environment, there might be other events that should be monitored, and the cluster behavior must be modified because of these events.

PowerHA provides a SMIT interface to the AIX error notification function. Use this function to detect an event that is not specifically monitored by the PowerHA (for example, a disk adapter failure) and trigger a response to this event.

Before you configure automatic error notification, a valid cluster configuration must be in place.

Automatic error notification applies to selected hard, unrecoverable error types such as ones that are related to disks or disk adapters. This utility does not support media errors, recovered errors, or temporary errors.

Enabling automatic error notification assigns one of two error notification methods for all error types as follows:

- ▶ The unrecoverable errors pertaining to resources that are determined to represent a single point of failure (SPOF) are assigned the **c1_failover** method and trigger a failover.
- ▶ All other noncritical errors are assigned to the **c1_logerror** method, and an error entry is logged against the `hacmp.out` file.

PowerHA automatically configures error notifications and recovery actions for several resources and error types, which include these items:

- ▶ All disks in the rootvg VG.
- ▶ All disks in cluster VGs, concurrent VGs, and file systems.
- ▶ All disks that are defined as cluster resources.

11.6.1 Disk monitoring consideration

In addition, PowerHA can monitor mirrored and non-mirrored VGs regardless of the disk type. When the loss of quorum is detected, an `LVM_SA_QUORCLOSE` entry is logged in the error log. PowerHA can initiate a takeover for the RG that contains the VG. This action is called selective failover on VG loss, and it is enabled by default. For more information, see [Selective failover on volume group loss](#).

11.6.2 Setting up automatic error notification

PowerHA can add automatic error notifications on all nodes. Automatic error notification methods are added automatically during cluster verification and synchronization.

To set up automatic error notifications, use the `smitty sysmirror` fast path and select **Problem Determination Tools** → **PowerHA SystemMirror Error Notification** → **Configure Automatic Error Notification** → **Add Error Notify Methods for Cluster Resources**.

Note: You cannot configure automatic error notification while the cluster is running, as shown by the following output:

```
jordan: HACMP clstrmgr must be down to run this command.  
jordan: cdsh: c1_rsh: (RC=1) /usr/es/sbin/cluster/cspoc/cexec c1_errnotify -a  
jessica: HACMP clstrmgr must be down to run this command.  
jessica: cdsh: c1_rsh: (RC=1) /usr/es/sbin/cluster/cspoc/cexec c1_errnotify -a
```

11.6.3 Listing automatic error notification

To list automatic error notifications that are configured in your cluster, complete the following steps:

1. Run the **smitty sysmirror** fast path and select **Problem Determination Tools → PowerHA SystemMirror Error Notification → Configure Automatic Error Notification.**
2. Select List Error Notify Methods for Cluster Resources.

The result is similar to the output in Example 11-1

Example 11-1 Sample list of automatic error notifications

COMMAND STATUS		
Command: OK	stdout: yes	stderr: no
Before command completion, additional instructions may appear below.		
jessica :		
jessica : HACMP Resource		Error Notify Method
jessica :		
jessica : hdisk0	/usr/es/sbin/cluster/diag/cl_failover	
jessica : hdisk1	/usr/es/sbin/cluster/diag/cl_logerror	
jessica : hdisk2	/usr/es/sbin/cluster/diag/cl_logerror	
jordan:		
jordan: HACMP Resource		Error Notify Method
jordan:		
jordan: hdisk0	/usr/es/sbin/cluster/diag/cl_failover	
jordan: hdisk1	/usr/es/sbin/cluster/diag/cl_logerror	
jordan: hdisk2	/usr/es/sbin/cluster/diag/cl_logerror	
F1=Help	F2=Refresh	F3=Cancel
F8=Image	F9=Shell	F10=Exit
n=Find Next		/=Find

11.6.4 Removing automatic error notifications

To remove automatic error notifications, complete the following steps:

1. Run the **smitty sysmirror** fast path and select **Problem Determination Tools → PowerHA SystemMirror Error Notification → Configure Automatic Error Notification → Remove Error Notify Methods for Cluster Resources.**
2. Press Enter to confirm.

11.6.5 Using error notification

The Administering PowerHA SystemMirror menu contains lists with hardware errors that are handled by the cluster automatic error notification utility. It also contains a list of hardware errors that are not handled by the cluster automatic error notification utility.

With PowerHA, you can customize the error notification method for other devices and error types and define a specific notification method rather than using one of the two automatic error notification methods.

To add a notify method, complete the following steps:

1. Run the **smitty sysmirror** fast path and select **Problem Determination Tools → PowerHA SystemMirror Error Notification → Add a Notify Method**.
2. Define the notification object:
 - Notification Object Name: A user-supplied name that uniquely identifies the error notification object.
 - Persist across system restart:
 - Yes: The error notification persists after system restart.
 - No: The error notification is used until the next system restart.
 - Process ID for use by Notify Method: The error notification is sent on behalf of the selected process ID. If you specify a nonzero process ID here, set Persist across system restart to No.
 - Select Error Class:
 - None: Choose this value to ignore this entry.
 - All: Match all error classes.
 - Hardware: Match all hardware errors,
 - Software: Match all software errors.
 - Errlogger: Operator notifications and messages from the **errlogger** program.
 - Select Error Type:
 - None: Choose this value to ignore this entry.
 - All: Match all error types.
 - PEND: Impending loss of availability.
 - PERF: Performance degradation.
 - PERM: Permanent errors.
 - TEMP: Temporary errors.
 - UNKN: Unknown error type.
 - Match Alertable errors: This field is intended to be used by alert agents of a system management application. If you do not use such applications, leave this field set to None.
 - None: Choose this value to ignore this entry.
 - All: Alert all errors.
 - True: Match alertable errors.
 - False: Match non-alertable errors.
 - Select Error Label: Select the error label from the list. There usually are over a thousand options. See the short description of the error labels in the `/usr/include/sys/errids.h` file.
 - Resource Name: The name of the failing resource. For the hardware error class, it is the device name. For the software errors class, it is the name of the failing executable. Select All to match all resource types.
 - Resource Class: For the hardware resource class, it is the device class. It is not applicable for software errors. Specify All to match all resource classes.
 - Resource Type: The type of the failing resource. For hardware error class, the device type by which a resource is known in the device object. Specify All to match all resource classes.

- Notify Method: The full-path name of the program to be run when an error is logged that matches any of the defined criteria that are listed in step 2 on page 491. You can pass the following variables to the executable:
 - \$1: Error log sequence number.
 - \$2: Error identifier.
 - \$3: Error class.
 - \$4: Error type.
 - \$5: Alert flag.
 - \$6: Resource name of the failing device.
 - \$7: Resource type of the failing device.
 - \$8: Resource class of the failing device.
 - \$9: Error log entry label.

3. Press Enter to create the error notification object.

After an error notification is defined, PowerHA offers the means to emulate it. You can emulate an error log entry with a selected error label. The error label is listed in the error log and the notification method is run by **errdemon**.

To emulate a notify method, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Problem Determination Tools** → **PowerHA SystemMirror Error Notification** → **Emulate Error Log Entry**.
2. Select the error label or the notify method name from the list. Only notify methods that have an error label that is defined are listed.
3. SMIT shows the error label, notification object name, and the notify method, as shown in Figure 11-5. Press Enter to confirm error log entry emulation.

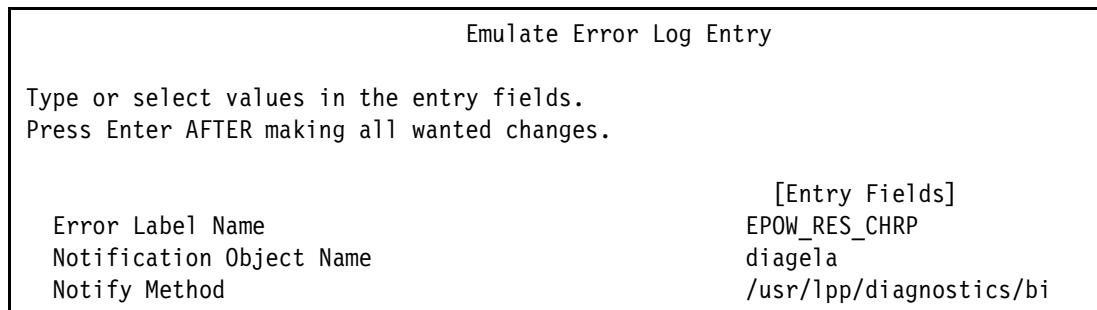


Figure 11-5 Error log emulation

11.6.6 Customizing the event duration

Cluster events run asynchronously and can complete in different amounts of time. Because PowerHA has no means to detect whether an event script has not hung, it runs a `config_too_long` event each time the processing of an event exceeds a certain amount of time. For such events, you can customize the amount of time that cluster services wait for an event to complete before issuing the `config_too_long` warning message.

Cluster events can be divided into two classes as follows:

- ▶ Fast events

These events do not include acquiring or releasing resources and normally complete in a shorter amount of time. For fast events, the time that PowerHA waits before issuing a warning is equal to Event Duration Time.

- ▶ Slow events

These events involve acquiring and releasing resources or using application server start and stop scripts. Slow events might complete in a longer amount of time. With customizing event duration time for slow events, you avoid receiving unnecessary system warnings during normal cluster operation. For slow events, the total time before receiving a config_too_long warning message is set to the sum of Event-only Duration Time and Resource Group Processing Time.

To change the total event duration time before receiving a config_too_long warning message, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Custom Cluster Configuration → Events → Cluster Events → Change>Show Time Until Warning**.
2. Complete these fields:
 - Max. Event-only Duration (in seconds)
The maximum time (in seconds) to run a cluster event. The default is 180 seconds.
 - Max. Resource Group Processing Time (in seconds)
The maximum time (in seconds) to acquire or release an RG. The default is 180 seconds.
 - Total time to process a Resource Group event before a warning is displayed
The total time for the Cluster Manager to wait before running the config_too_long script. The default is 6 minutes. This field is the sum of the two other fields and is not editable.
3. Press Enter to complete the changes.
4. Verify and synchronize the cluster to propagate the changes by running **c1mgr sync cluster**.

11.6.7 Defining new events

With PowerHA, you can define your own cluster events that run specified recovery programs. The events that you define can be related to Resource Monitoring and Control (RMC) resources. An RMC resource refers to an instance of a physical or logical entity that provides services to some other component of the system. Resources can refer to both hardware and software entities. For example, a resource might be a physical disk (IBM.PhysicalVolume) or a running program (IBM.Program). Resources of the same type are organized in classes. You can obtain information regarding resources and resource classes by using the **1srcsrdef** command. The AIX resource monitor generates events for operating system-related resource conditions.

For more information about resources and RMC, see the [IBM Reliable Scalable Cluster Technology \(RSCT\) website](#).

Recovery programs

A recovery program consists of a sequence of recovery command specifications with the following format:

- ```
:node_set recovery_command expected_status NULL
```
- ▶ **node\_set**: The set of nodes on which the recovery program runs. Use one of the following values:
    - all: The recovery command runs on all nodes.
    - event: The node on which the event occurred.
    - other: All nodes except the one on which the event occurred.
  - ▶ **recovery\_command**: A string (delimited by quotation marks) that specifies a full path to the executable program. The command cannot include any arguments. Any executable program that requires arguments must be a separate script. The recovery program must have the same path on all cluster nodes. The program must specify an exit status.
  - ▶ **expected\_status**: Integer status to return when the recovery command completes successfully. The cluster manager compares the actual status that is returned against the expected status. A mismatch indicates unsuccessful recovery. If you specify the character X in the expected status field, the cluster manager skips the comparison.
  - ▶ **NULL**: Not used. Included for future functions.

Multiple recovery command specifications can be separated by the **barrier** command. All recovery command specifications before a barrier start in parallel. When a node encounters a **barrier** command, all nodes must reach the same barrier before the recovery program resumes.

To define your new event, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Custom Cluster Configuration** → **Events** → **Cluster Events** → **User-Defined Events** → **Add Custom User-Defined Events**.
2. Complete these fields:
  - Event name: The name of the event.
  - Recovery program path: The full path of the recovery program.
  - Resource name: The RMC resource name as returned from **lsrcdef** command.
  - Selection String: An SQL expression that includes attributes of the resource instance.
  - Expression: Relational expression between dynamic resource attributes. When the expression evaluates true, it generates an event.
  - Reram expression: An expression that is used to generate an event that alternates with an original event expression until it is true, and then the reram expression is used until it is true, then the event expression is used, and so on. Usually, the logical inverse or complement of the event expression.
3. Press Enter to create the error notification object.

An example of defining a user-defined event is shown in Example 11-2.

*Example 11-2 Defining a user-defined event*

---

Add a Custom User-Defined Event

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

|                         |                       |
|-------------------------|-----------------------|
| * Event name            | [Entry Fields]        |
| * Recovery program path | [user_defined_event]  |
| * Resource name         | [/user_defined.rp]    |
| * Selection string      | [IBM.FileSystem]      |
| * Expression            | [name = "/var"]       |
| Rearm expression        | [PercentTotUsed > 70] |
|                         | [PercentTotUsed < 50] |

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

---

We used the following recovery program:

```
#Recovery Program for user-defined event call 1
event "/usr/ha/trigger_user_defined_event.sh" 0 NULL
```

The /usr/ha/trigger\_user\_defined\_event.sh script can perform any form of notification, such as writing to a log file or sending an email, using SMS, or using an SNMP trap.

For more information about user-defined events, see [IBM PowerHA SystemMirror for AIX Standard Edition Version 7.2: Planning PowerHA SystemMirror](#).





# Network considerations

In this chapter, we describe several network options that are available within PowerHA. Some of these features include the service IP distribution policy and the automatic creation of the `c1hosts` file.

This chapter contains the following topics:

- ▶ Multicast considerations
- ▶ Distribution preference for service IP aliases
- ▶ Cluster tunables
- ▶ Site-specific service IP labels
- ▶ Understanding the `netmon.cf` file
- ▶ Using `poll_uplink`
- ▶ Understanding the `c1hosts` file

## 12.1 Multicast considerations

The PowerHA SystemMirror 7.1 Standard Edition High Availability solution introduced clustering by using multicast-based (IP-based multicast) communication between the nodes or hosts in the cluster. Multicast-based communication provides for optimized communication methods to exchange heartbeats, and also enables clustering software to communicate critical events, cluster coordination messages, and more, in one-to-many methods instead of communication by one-to-one between the hosts.

With PowerHA SystemMirror 7.2, unicast communications are always used between sites in a linked cluster. Within a site, you can select unicast (the default) or multicast communications.

Multicast communication is an established mode of communication in the world of Internet Protocol network communication. However, sometimes the network switches that are used in the communication path must be reviewed and enabled for multicast traffic to flow between the cluster hosts through them. This document explains some of the network setup aspects that might need to be reviewed before the PowerHA SystemMirror 7.1 cluster is deployed.

**Note:** PowerHA 7.1.0 - 7.1.2 require multicast within a site. PowerHA 7.1.3 introduced unicast as an option, making multicast optional.

PowerHA uses a cluster health management layer that is embedded as part of the operating system. It is called Cluster Aware AIX (CAA). CAA uses kernel-level code to exchange heartbeats over a network, a storage area network (SAN) fabric (when correct Fibre Channel (FC) adapters are deployed), and disk-based messaging through the central repository.

CAA uses a multicast IP-based network communication mechanism to communicate among the various nodes in the cluster within a site. Administrators can manually configure a multicast address to use for cluster communication or allow PowerHA SystemMirror and CAA to choose a multicast address.

If multicast is selected during the initial cluster configuration, the multicast traffic must flow among the cluster hosts in the data center before cluster formation can be attempted. Plan to test and verify the multicast traffic flow among the “would-be” cluster nodes before attempting to create the cluster. Review the guidelines in the following sections to test the multicast packet flow among the hosts.

### 12.1.1 Multicast concepts

Multicasting is a form of addressing where a group of hosts forms a group and exchanges messages. A multicast message that is sent by one host in the group is received by all hosts in the group. There is efficient cluster communication when messages must be sent to all nodes in the cluster. For example, a cluster member might need to notify the remaining nodes about a critical event, so they send a single multicast packet with the relevant information.

#### Network switches

Hosts communicate over a network fabric that might consist of many switches and routers. A switch connects separate hosts and network segments so that network traffic can be sent to the correct place. A switch refers to a multiport network bridge that processes and routes data at the data link layer (Layer 2) of the OSI model. Some switches can also process data at the network layer (Layer 3).

Typically, a data center networking environment consists of hosts that are connected through a network fabric that consists of Ethernet or cabling and switches. Often, switches are interconnected to form the fabric among the hosts. When switches cascade, multicast packets must flow from the host in the cluster to a switch and then through the other switches to reach the destination host in the cluster. Because switches review multicast packets differently compared to regular network communication, switch-to-switch communication might not occur for multicast packets if the setup is incorrect. Multicast *must* be enabled on the switches and with any forwarding, if applicable.

## Internet Group Management Protocol

Internet Group Management Protocol (IGMP) is a communications protocol that enables host receivers to inform a multicast router (IGMP querier) of the host's intention to receive particular multicast traffic. This protocol runs between a router and hosts:

- ▶ A router can ask hosts if they need a particular multicast stream (IGMP query).
- ▶ Hosts can respond to the router if they seek a particular multicast stream (IGMP reports).

The IGMP communication protocol is used by the hosts and the adjacent routers on IP networks to interact and establish rules for multicast communication, especially to establish multicast group membership. Switches that feature IGMP snooping derive useful information by observing these IGMP transactions between the hosts and routers, which enable the switches to correctly forward the multicast packets when needed to the next switch in the network path.

## IGMP snooping

IGMP snooping is an activity that is performed by the switches to track the IGMP communications packet exchanges and filter the multicast packets. Switches monitor the IGMP traffic and allow out the multicast packets only when necessary. The switch typically builds an IGMP snooping table that has a list of all the ports that have requested a particular multicast group and uses this table to allow or disallow the multicast packets to flow.

## Multicast routing

The network entities that forward multicast packets by using special routing algorithms are referred to as *mouters*. Also, router vendors might implement multicast routing (see the router vendor's documentation and guidance). Hosts and other network elements implement m routers and allow the multicast network traffic to flow. Some traditional routers also support multicasting packet routing.

When switches are cascaded, or chained, setting up the switch to forward the packets might be necessary to implement mrouting. However, this approach might be one of the possible approaches to solving multicast traffic flow issues in the environment. For more information about setting up the switches for multicast traffic, see the switch vendor's documentation and guidance.

## 12.1.2 Multicast guidelines

These guidelines can help with the multicast setup in your environment. However, they are generic in nature, and the configuration of the switches depends on your network environment, switch type, and capabilities.

### Multicast testing

Do not attempt to create the cluster by using multicast until you verify that multicast traffic flows without interruption between the nodes that are part of the cluster. Clustering will not continue if the **mping** test fails. If problems occur with multicast communication in your network environment, contact the network administrator and review the switches that are involved and the setup that is needed. After the setup is complete, retest the multicast communication.

**The mping test:** One of the simplest methods to test end-to-end multicast communication is to use the **mping** command, which is available on AIX. On one node, start the **mping** command in receive mode, and then use the **mping** command to send packets from another host. If multiple hosts are part of the cluster, test end-to-end **mping** communication from each host to the other hosts.

The **mping** command can be invoked with a particular multicast address; otherwise, it chooses a default multicast address. A test for our cluster is shown in Example 12-1.

*Example 12-1 Using mping to test*

---

```
[jessica:root] / # mping -v -r -c 5 -a 228.168.100.51
mping version 1.1
Connecting by using IPv4.
Listening on 228.168.100.51/4098:
Replies to mping from 192.168.100.52 bytes=32 seqno=0 ttl=1
Replies to mping from 192.168.100.52 bytes=32 seqno=0 ttl=1
Replies to mping from 192.168.100.52 bytes=32 seqno=1 ttl=1
Replies to mping from 192.168.100.52 bytes=32 seqno=1 ttl=1
Replies to mping from 192.168.100.52 bytes=32 seqno=2 ttl=1

[maddi:root] / # mping -v -s -c 5 -a 228.168.100.51
mping version 1.1
Connecting by using IPv4.
mpinging 228.168.100.51/4098 with ttl=1:

32 bytes from 192.168.100.51 seqno=0 ttl=1 time=0.356 ms
32 bytes from 192.168.100.51 seqno=0 ttl=1 time=0.433 ms
32 bytes from 192.168.100.51 seqno=0 ttl=1 time=0.453 ms
32 bytes from 192.168.100.51 seqno=0 ttl=1 time=0.471 ms
32 bytes from 192.168.100.51 seqno=1 ttl=1 time=0.366 ms
32 bytes from 192.168.100.51 seqno=1 ttl=1 time=0.454 ms
32 bytes from 192.168.100.51 seqno=1 ttl=1 time=0.475 ms
32 bytes from 192.168.100.51 seqno=1 ttl=1 time=0.493 ms
32 bytes from 192.168.100.51 seqno=2 ttl=1 time=0.367 ms
32 bytes from 192.168.100.51 seqno=2 ttl=1 time=0.488 ms
Sleeping for 1 second to wait for any additional packets to arrive.

--- 228.168.100.51 mping statistics ---
5 packets transmitted, 10 packets received, 0% packet loss
round-trip min/avg/max = 0.356/0.436/0.493 ms
```

---

For the address input to `mping`, use the actual multicast address that is during clustering. CAA creates a default multicast address if one is not specified during cluster creation. This default multicast address is formed by combining (by using OR) 228.0.0.0 with the lower 24 bits of the IP address of the host. As an example, in our case the host IP address is 192.168.100.51, so the default multicast address is 228.168.100.51.

## Troubleshooting

If `mping` fails to receive packets from host to host in the network environment, some issue in the network path exists regarding multicast packet flow.

To troubleshoot the issue, use the following general guidelines:

- ▶ Review the switch vendor's documentation for guidance regarding multicast usage in the switch setup.
- ▶ Disable IGMP snooping on the switches. Most switches allow for disabling IGMP snooping. If your network environment does, disable IGMP snooping and allow all multicast traffic to flow without any problems across switches.
- ▶ If your network requirements do not allow snooping to be disabled, debug the problem by disabling IGMP snooping and then adding network components, one at a time, for snooping.
- ▶ Debug, if necessary, by eliminating any cascaded switch configurations by having only one switch between the hosts.

## 12.2 Distribution preference for service IP aliases

When you use IP aliasing with multiple service IP addresses configured, PowerHA analyzes the total number of aliases, which are defined to PowerHA or not, and assigns each service address to the least loaded interface. PowerHA provides added control over their placement so that you can define a distribution preference for your service IP label aliases.

This network-wide attribute can be used to customize the load-balancing of PowerHA service IP labels, considering any persistent IP labels that are already configured. The distribution that is selected is maintained during cluster startup and subsequent cluster events. The distribution preference is maintained if acceptable network interfaces are available in the cluster. However, PowerHA always keeps service IP labels active, even if the preference cannot be satisfied.

The placement of the service IP labels can be specified with these distribution preferences:

- ▶ Anti-Colocation
  - The default. PowerHA distributes the service IP labels across all boot IP interfaces in the same PowerHA network on the node. The first service label that is placed on the interface is the source address for all outgoing communication on that interface.
- ▶ Colocation
  - PowerHA allocates all service IP addresses on the same boot IP interface.
- ▶ Colocation with persistent label
  - PowerHA allocates all service IP addresses on the boot IP interface that is hosting the persistent IP label, which can be useful in environments with virtual private network (VPN) and firewall configuration, where only one interface is granted external connectivity. The persistent label is the source address.

- ▶ Colocation with source  
Service labels are mapped by using the colocation preference. You can choose one service label as a source for outgoing communication. The service label that is chosen in the next field is the source address.
- ▶ Colocation with Persistent Label and Source  
Service labels are mapped to the same physical interface that has the persistent IP label for this network. You may choose one service label as the source for outgoing communication. The service label that is chosen in the next field is the source address.
- ▶ Anti-Colocation with source  
Service labels are mapped by using the Anti-Colocation preference. If not enough adapters are available, more than one service label can be placed on one adapter. You may select one label as the source address for outgoing communication.
- ▶ Anti-Colocation with persistent label  
PowerHA distributes all service IP labels across all boot IP interfaces in the same logical network that are *not* hosting the persistent IP label. If no other interfaces are available, the service IP labels share the adapter with the persistent IP label.
- ▶ Anti-Colocation with persistent label and source  
Service labels are mapped by using the Anti-Colocation with Persistent preference. One service address can be chosen as a source address for the case when more service addresses exist than the boot adapters.
- ▶ Disable Firstalias  
PowerHA 7.1 automatically configures the service IP as an alias by using the **firstalias** option regardless of the user's setting. However, in certain scenarios, such as Network Installation Management (NIM) operations, the default **firstalias** feature can cause errors. With this option, the user can disable **firstalias** and retain the historic default original mode.

### 12.2.1 Configuring a service IP distribution policy

The distribution preference can be set or changed dynamically. To configure this type of distribution policy, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources → Resources → Configure Service IP Labels/Addresses → Configure Service IP Labels/Addresses Distribution Preferences**, and then press Enter.  
PowerHA displays a network list in a window.
2. Select the network for which you want to specify the policy and press Enter.
3. From the Configure Service IP Labels/Address Distribution Preference panel, press F4 and choose a distribution preference.
4. Most options also allow you to choose a specific service IP for outgoing packets, which can be useful especially when firewalls are involved. Figure 12-1 on page 503 shows an example.

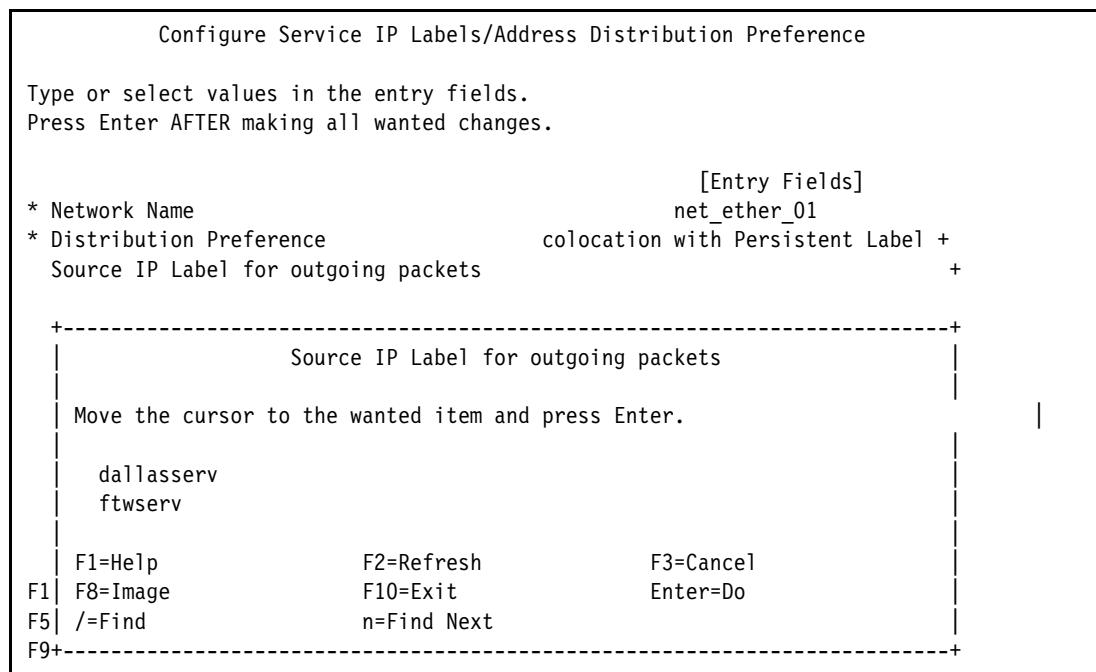


Figure 12-1 Specifying a service IP address for outgoing packets

5. Press Enter to accept your selection and update the PowerHA Object Data Manager (ODM) on the local node.
6. Verify and synchronize the cluster by running `clmgr sync cluster`. If the cluster is active, it initiates a dynamic reconfiguration.

**Note:** Configuring the service IP distribution policy results in this message:

`cldare: Detected changes to service IP label applsvc. Changing parameters of the service IP label through a DARE may result in releasing resource group <name>.`

## Viewing the distribution preferences for service IP label aliases

You can display the current distribution preferences for each network by using the `cltopinfo` command. If the setting is the default of Anti-Colocation, no additional details are shown. However, as Example 12-2 shows for `net_ether_01`, if the option is set to anything else, it is clearly displayed in the output of the `cltopinfo -w` command.

Example 12-2 Verifying a service IP distribution policy by running the `cltopinfo` command

---

```
[ashley:root] /utilities # cltopinfo -w

Network net_ether_01
 NODE maddi:
 ftwserv 10.10.10.52
 dallasserv 10.10.10.51
 maddi_xd 192.168.150.52
 NODE jessica:
 ftwserv 10.10.10.52
 dallasserv 10.10.10.51
 jessica_xd 192.168.150.51
 NODE ashley:
 ftwserv 10.10.10.52
 dallasserv 10.10.10.51
```

```

ashley_xd 192.168.150.53
Network net_ether_01 is using the following distribution preference for service labels:
Colocation with persistent - service labels will be mapped to the same interface as the
persistent label.

Network net_ether_010
 NODE maddi:
 maddi 192.168.100.52
 NODE jessica:
 jessica 192.168.100.51
 NODE ashley:
 ashley 192.168.100.53

```

---

## 12.2.2 Example scenarios with a service IP distribution policy

In our testing, we changed the service IP distribution policy (as shown in 12.2.1, “Configuring a service IP distribution policy” on page 502) to colocation with persistent, with cluster services down on all the nodes. On cluster startup, verify that the IP labels and persistent IP addresses are all placed on the same adapter as specified by the policy, as shown in Example 12-3.

*Example 12-3 Verifying that the service IP policy is used after a cluster startup*

```

jessica:# more /etc/hosts
10.10.31.31 jessicaa # base address 1
10.10.32.31 jessicab # base address 2
192.168.100.31 jessicapers # jessica persistent address
192.168.100.82 maddisvc # maddi service address
192.168.100.83 ashleysvc # ashley service address

jessica:# netstat -i
Name Mtu Network Address Ipkts Ierrs Opkts Oerrs Coll
en0 1500 link#2 0.2.55.4f.c4.ab 5044669 0 1828909 0 0
en0 1500 10.10.31 jessicaa 5044669 0 1828909 0 0
en0 1500 192.168.100 jessicapers 5044669 0 1828909 0 0
en0 1500 192.168.100 maddisvc 5044669 0 1828909 0 0
en0 1500 192.168.100 ashleysvc 5044669 0 1828909 0 0
en3 1500 link#3 0.20.35.e2.7f.8d 3191047 0 1410806 0 0
en3 1500 10.10.32 jessicab 3191047 0 1410806 0 0
lo0 16896 link#1 0.0.0.0 1952676 0 1957548 0 0
lo0 16896 127 localhost 1952676 0 1957548 0 0
lo0 16896 localhost 1952676 0 1957548 0 0

```

---

**Note:** In this output, the node **jessica** had the resource groups (RGs) for nodes **maddi** and **ashley** and their corresponding service IP addresses. The distribution policy was set to colocation with persistent.

Our testing of the dynamic change of this policy resulted in no move of any of the labels after a synchronization. The following message was logged during the synchronization of the cluster after making the service IP distribution policy change:

Verifying additional prerequisites for Dynamic Reconfiguration...

```
cldare: Detected changes to service IP label maddisvc. Changing parameters of service IP label through a DARE may result in releasing resource group APP1_RG.
```

```
cldare: Detected changes to service IP label ashleysvc. Changing parameters of service IP label through a DARE may result in releasing resource group APP2_RG.
```

**Note:** For this instance, the message that is logged is generic and is reported only because a change was detected. If that was the only change that was made, no resources are taken offline.

A change to the service IP distribution policy is enforced only when we manually invoke a swap event or stop and restart PowerHA on a node. This behavior is the intended one of the feature to avoid any potential disruption of connectivity to those IP addresses. The remaining cluster nodes will not enforce the policy unless the cluster services are also stopped and restarted on them.

## 12.3 Cluster tunables

CAA monitors the interfaces of each node. When using multicast, gossip packets are periodically sent from each node in the cluster for timing purposes. These gossip packets are automatically replied to by the other nodes in the cluster. The packet exchanges are used to calculate the round-trip time.

The round-trip time (rtt) value is shown in the output of the `lsccluster -i` and `lsccluster -m` commands. The mean deviation in network rtt is the average round-trip time, which is automatically managed by CAA.

### 12.3.1 Changing cluster-wide tunables

To change the cluster heartbeat settings, modify the attributes for the PowerHA cluster from the custom cluster configuration, as shown in Figure 12-2.

| Cluster heartbeat settings                                                                 |            |
|--------------------------------------------------------------------------------------------|------------|
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes. |            |
| [Entry Fields]                                                                             |            |
| * Network Failure Detection Time                                                           | [20] #     |
| * Node Failure Detection Timeout                                                           | [30] #     |
| * Node Failure Detection Grace Period                                                      | [10] #     |
| * Node Failure Detection Timeout during LPM                                                | [600] #    |
| * LPM Node Policy                                                                          | [manage] + |
| * Repository Mode                                                                          | Event +    |
| * Config Timeout                                                                           | [240] #    |
| * Disaster Recovery                                                                        | Enabled +  |
| * PVM Watchdog timer                                                                       | Disable    |

Figure 12-2 Cluster heartbeat settings for standard one-site clusters

Use the **smitty sysmirror** fast path and select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Manage the Cluster** → **Cluster heartbeat settings**.

The following definitions apply for the fields:

- ▶ **Network Failure Detection Time:** The time in seconds that the health management layer waits before declaring a network failure indication. The minimum and maximum values are in effect on the CAA layer and can be seen by running the **clctrl -tune -L network\_fdt** command. Although the MIN and MAX values for network\_fdt and the values that are shown in the output of the command are in milliseconds, enter the values in this screen in seconds.

**Note:** This setting is global across all networks.

- ▶ **Node failure detection timeout:** The time in seconds that the health management layer waits before preparing to declare a node failure indication. The minimum and maximum values are in effect on the CAA layer and can be seen by running the **clctrl -tune -L node\_timeout** command. Although the MIN and MAX values for network\_fdt and the values that are shown in the output of the command are in milliseconds, enter the values in this screen in seconds.
- ▶ **Node failure detection grace period:** The time in seconds that the node waits after the Node Failure Detection Timeout before declaring that a node failed. The minimum and maximum values are in effect on the CAA layer and can be seen by running the **clctrl -tune -L node\_down\_delay** command. Although the MIN and MAX values for network\_fdt and the values that are shown in the output of the command are in milliseconds, enter the values in this screen in seconds.
- ▶ **Node failure detection timeout during LPM:** If specified, this timeout value (in seconds) is used during Live Partition Mobility (LPM) instead of the Node Failure Detection Timeout value. You can use this option to increase the Node Failure Detection Timeout during the LPM duration to ensure that it is greater than the LPM freeze duration to avoid any risk of unwanted cluster events. Enter a value 10 - 600.

- ▶ LPM Node Policy: Specifies the action to take on the node during an LPM operation. If unmanage is selected, the cluster services stop with the Unmanage Resource Groups option during the duration of the LPM operation. Otherwise, PowerHA SystemMirror continues to monitor the RGs and application availability.
- ▶ Repository Mode: Controls the node behavior when cluster repository disk access is lost. Valid values are Assert or Event. Event is the default. When the value is set to Assert, the node fails when it loses access to the cluster primary repository without moving to back up repositories. When the value is set to Event, an AHAFS event is generated.
- ▶ Config Timeout: Specifies the CAA configuration timeout for a configuration change. A positive value indicates the maximum number of seconds CAA waits on the execution of client-side callouts, including scripts and the CAA configuration code. A value of zero disables the timeout. The default value is 240 seconds. The valid range is 0 - 2147483647.
- ▶ Disaster Recovery: To enable or disable the CAA physical volume identifier (PVID)-based identification when universally unique identifier (UUID)-based authentication fails. A value of 1 is the default (enabled). The 0 value is disabled.
- ▶ PVM Watchdog Timer: Controls the behavior of the CAA PVM Watchdog timer. Here are the valid values:
  - DISABLE: The tunable is disabled.
  - DUMP\_RESTART: CAA dumps and restarts the logical partition (LPAR) when the virtual machine (VM) fails to reset the timer.
  - HARD\_RESET: CAA hard resets the LPAR when VM fails to reset the timer.
  - HARD\_POWER\_OFF: CAA hard powers off the LPAR when VM fails to reset the timer. The user must start the LPAR by using Hardware Management Console (HMC) options.

When cluster sites are used, that is, specifically linked sites, the options differ, as shown in Figure 12-3.

| Cluster heartbeat settings                                                                                                                                                                                                                                                   |                                                                                                 |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes.                                                                                                                                                                                   |                                                                                                 |
| [Entry Fields]                                                                                                                                                                                                                                                               |                                                                                                 |
| * Node Failure Detection Timeout<br>* Node Failure Detection Grace Period<br>* Node Failure Detection Timeout during LPM<br>* LPM Node Policy<br>* Link Failure Detection Timeout<br>* Site Heartbeat Cycle<br>* Repository Mode<br>* Config Timeout<br>* PVM Watchdog timer | [30] #<br>[10] #<br>[600] #<br>[manage] +<br>[30] #<br>[1] #<br>Event #<br>[240] #<br>Disable + |

*Figure 12-3 Linked cluster heartbeat settings*

There are more parameters that are available:

- ▶ Link failure detection timeout: The time (in seconds) that the health management layer waits before declaring that the inter-site link failed. A link failure detection can cause the cluster to switch to another link and continue the communication. If all the links failed, a site failure occurs. The default is 30 seconds.
- ▶ Site heartbeat cycle: A number factor (1 - 10) that controls the heartbeat among the sites.

All the options that are available when sites are defined are shown in Figure 12-3 on page 507. Note the difference between Figure 12-2 on page 506 and Figure 12-3 on page 507.

Most changes require cluster synchronization, but this specific change is dynamic, so a cluster restart is *not* required. However, rarely is there a good reason to not perform at least a verification after a change is made.

### 12.3.2 Resetting cluster tunables to the cluster defaults

To restore all tunables to their default setting, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Manage the Cluster** → **Reset Cluster Tunables**.
2. Choose either Yes or No to synchronize the cluster. There is rarely a good reason to run this option and *not* synchronize the cluster. Press Enter.

### 12.3.3 Changing the network settings

To change the current network settings for a network, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Cluster Nodes and Networks** → **Manage Networks and Network Interfaces** → **Networks** → **Change/Show a Network**.
2. Choose the specific network from the list and press Enter.
3. Complete the following fields, as shown in Figure 12-4 on page 509. A description of each field follows.
  - New Network Name: Enter the name for this PowerHA SystemMirror network. Use alphanumeric characters and underscores, and no more than 31 characters.
  - Network Type: Press F4 and select the network type from the list:
    - XD\_data
    - XD\_ip
    - ether
  - Network Mask: Enter the netmask as "X.X.X.X" to configure an IP version 4 (IPv4) network. X is [0 - 255]. Enter the prefix length as [1 - 128] to configure an IPv6 network.
  - Network attribute: Press F4 and choose one of these options:
    - public
    - private

- Unstable Threshold: Network instability occurs when there are an excessive number of events received over a period. Unstable Threshold defines the number of events that must be received inside the unstable period for the network to be declared as unstable. Provide an integer value 1 - 99.
  - Unstable Period (seconds): Network instability occurs when there are an excessive number of events received over a period. The unstable period defines the period that is used to determine instability. If the threshold number of events is received inside the unstable period, the network is declared as unstable. Provide an integer value 1 - 120 seconds.
4. Synchronize the cluster.

| Change/Show a Network                                                                                                                              |                 |                |   |
|----------------------------------------------------------------------------------------------------------------------------------------------------|-----------------|----------------|---|
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes.                                                         |                 |                |   |
| [Entry Fields]                                                                                                                                     |                 |                |   |
| * Network Name<br>New Network Name                                                                                                                 | net_ether_010   |                |   |
| * Network Type                                                                                                                                     | [bdb_net]       | +<br>[XD_data] |   |
| * Netmask(IPv4)/Prefix Length(IPv6)                                                                                                                | [255.255.255.0] |                |   |
| * Network attribute                                                                                                                                | public          | +<br>[3]       | # |
| * Unstable Threshold                                                                                                                               | [3]             |                | # |
| * Unstable Period (seconds)                                                                                                                        | [90]            |                | # |
| F1=Help      F2=Refresh      F3=Cancel      F4=List<br>F5=Reset      F6=Command      F7>Edit      F8=Image<br>F9=Shell      F10=Exit      Enter=Do |                 |                |   |

Figure 12-4 Change/Show a Network

## 12.4 Site-specific service IP labels

The site-specific service IP label feature provides the ability to have unique service addresses at each site. This ability can help with using different subnets at each site. The feature can be used for the following IP network types:

- ▶ ether
- ▶ XD\_data
- ▶ XD\_ip

This feature can also be used with regular service IP labels and persistent IP labels. In general, use persistent IP labels, especially ones that are node-bound with XD\_data networks because no communication occurs through the service IP label that is configurable on multiple nodes.

To configure and use site-specific service IP labels, sites must be defined to the cluster. After you add a cluster and add nodes to the cluster, complete the following steps:

1. Add sites.
2. Add more networks as needed (ether XD\_data or XD\_ip).
3. Add interfaces to each network.

4. Add service IP labels:
  - Configurable on multiple nodes.
  - Specify the associated site.
5. Add RGs.
6. Add service IP labels to the RGs.
7. Synchronize the cluster.
8. Test site-specific IP failover.

**Important:** In PowerHA Enterprise Edition, configurations that use site-specific IP labels with XD network types *require* configuring a persistent IP label on each node.

In our test scenario, we have a 2-node cluster (`maddi` and `jessica`) that has a single Ether network with a single interface that is defined to it. We also have a volume group (VG) available on each node that is named `xsitevg`. Our starting topology is shown in Example 12-4.

---

*Example 12-4 Starting topology for a site-specific test scenario*

---

```
[jessica:root] / # cltopinfo
Cluster Name: xsite_cluster
Cluster Type: Standard
Heartbeat Type: Unicast
Repository Disk: hdisk1 (00f6f5d015a4310b)
```

There are 2 nodes and 2 networks defined

NODE `maddi`:

```
Network net_ether_010
 maddi 192.168.100.52
```

NODE `jessica`:

```
Network net_ether_010
 jessica 192.168.100.51
```

---

## Adding sites

To add the sites, complete the following steps:

1. Run the `smitty sysmirror` fast path, select **Cluster Nodes and Networks** → **Manage Site** → **Add a Site**, and press Enter.
2. We add the two sites `dallas` and `fortworth`. Node `jessica` is a part of the `dallas` site, and node `maddi` is a part of the `fortworth` site. The Add a Site menu is shown in Figure 12-5 on page 511.

Add a Site

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

|                                                                                                                                                    |                                                          |   |
|----------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------|---|
| * Site Name<br>* Site Nodes<br>Cluster Type                                                                                                        | <b>[Entry Fields]</b><br>[dallas]<br>jessica<br>Standard | + |
| F1=Help      F2=Refresh      F3=Cancel      F4=List<br>F5=Reset      F6=Command      F7>Edit      F8=Image<br>F9=Shell      F10=Exit      Enter=Do |                                                          |   |

Figure 12-5 Add a Site

After you add the sites, the cluster type automatically changes from Standard to Stretched. The next site that is added automatically shows the new cluster type. This automatic change is fine for cross-site Logical Volume Manager (LVM) configurations, but it will not work if you plan to use PowerHA Enterprise Edition and a *linked* cluster. In that case, you must delete and re-create the cluster.

Figure 12-6 shows the output from adding the site.

COMMAND STATUS

|                                                                                                                                                                                                                                                                                                   |             |            |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------|------------|
| Command: OK                                                                                                                                                                                                                                                                                       | stdout: yes | stderr: no |
| Before command completion, additional instructions may appear below.                                                                                                                                                                                                                              |             |            |
| claddssite: Node jessica has been added to the site dallas<br>Note: Cluster type has been changed to Stretched cluster.<br>If you want to work with sites in a Linked cluster,<br>you need to remove the cluster definition and start<br>again, specifying the Linked cluster as the cluster type |             |            |

Figure 12-6 Adding a site return results

## Adding a network

To add a network, complete the following steps:

1. Using the **smitty sysmirror** fast path, select **Cluster Nodes and Networks** → **Manage Networks and Network Interfaces** → **Networks** → **Add a Network**, and press Enter. Choose the network type, in our case we select **XD\_ip**, and press Enter.
2. You can keep the default network name, as we did, or specify one as wanted.

Figure 12-7 shows the Add an IP-Based Network to the HACMP Cluster panel.

| Add an IP-Based Network to the HACMP Cluster                                               |                 |
|--------------------------------------------------------------------------------------------|-----------------|
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes. |                 |
| [Entry Fields]                                                                             |                 |
| * Network Name                                                                             | [net_XD_ip_01]  |
| * Network Type                                                                             | XD_ip           |
| * Netmask(IPv4)/Prefix Length(IPv6)                                                        | [255.255.255.0] |
| * Network attribute                                                                        | public +        |

Figure 12-7 Adding a network

### Adding interfaces to a network

To add interfaces to the newly created network, complete the following steps:

1. Using the **smitty sysmirror** fast path and select **Cluster Nodes and Networks** → **Manage Networks and Network Interfaces** → **Network Interfaces** → **Add a Network Interface**. Choose the newly created network, which in our case is net\_XD\_ip\_01. Press Enter.
2. Complete the options that you want. In our case, we add jessica\_xd on node jessica, as shown in Figure 12-8. We also repeat the steps for maddi\_xd on node maddi.

| Add a Network Interface                                                                    |                |           |          |
|--------------------------------------------------------------------------------------------|----------------|-----------|----------|
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes. |                |           |          |
| [Entry Fields]                                                                             |                |           |          |
| * IP Label/Address                                                                         | [jessica_xd] + |           |          |
| * Network Type                                                                             | XD_ip          |           |          |
| * Network Name                                                                             | net_XD_ip_01   |           |          |
| * Node Name                                                                                | [jessica] +    |           |          |
| Network Interface                                                                          | []             |           |          |
| F1=Help                                                                                    | F2=Refresh     | F3=Cancel | F4=List  |
| F5=Reset                                                                                   | F6=Command     | F7>Edit   | F8=Image |
| F9=Shell                                                                                   | F10=Exit       | Enter=Do  |          |

Figure 12-8 Adding network interfaces

### Adding service IP labels

To define the site-specific service IP labels, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Cluster Applications and Resource** → **Resources** → **Configure Service IP Labels/Addresses** → **Add a Service IP Label/Address**.
2. Choose the network (net\_XD\_ip\_01 in our case), choose an IP label from the list, and press Enter. For the IP label to appear in the list, it must already be in the /etc/hosts file, and preferably on every node of the cluster.

Ensure that you specify the Associated Site that matches the node in which the service label belongs, as shown in Figure 12-9 on page 513.

Repeat this step as needed for each service IP label. We repeated and added the service IP label ftwserv to the fortworth site.

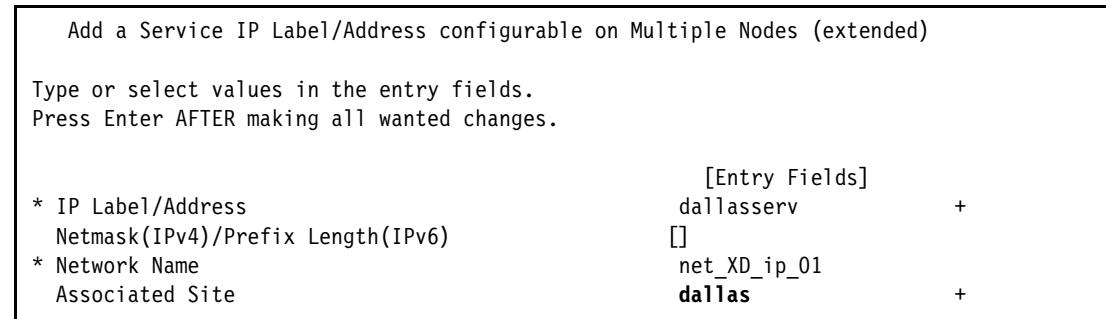


Figure 12-9 Adding a site-specific service IP label

The configured topology is shown in Example 12-5.

---

*Example 12-5 New topology after adding network and service IP addresses*

---

```
[jessica:root] / # cltopinfo
Cluster Name: xsite_cluster
Cluster Type: Stretched
Heartbeat Type: Unicast
Repository Disk: hdisk1 (00f6f5d015a4310b)
Cluster Nodes:
 Site 1 (dallas):
 jessica
 Site 2 (fortworth):
 maddi
```

There are 2 nodes and 2 networks defined

```
NODE maddi:
 Network net_XD_ip_01
 ftwser 10.10.10.52
 dallasserv 10.10.10.51
 maddi_xd 192.168.150.52
 Network net_ether_010
 maddi 192.168.100.52

NODE jessica:
 Network net_XD_ip_01
 ftwser 10.10.10.52
 dallasserv 10.10.10.51
 jessica_xd 192.168.150.51
 Network net_ether_010
 jessica 192.168.100.51
```

```
[jessica:root] / # clsssite
```

| Sitename  | Site Nodes | Dominance | Protection Type |
|-----------|------------|-----------|-----------------|
| dallas    | jessica    |           | NONE            |
| fortworth | maddi      |           | NONE            |

## Adding a resource group

To add an RG, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources → Resource Groups → Add a Resource Group**, and press Enter.
2. Complete the fields. Our configuration is shown in Figure 12-10.

**Note:** The additional options, which are shown in bold in Figure 12-10 on page 514, are available only if sites are defined.

|                                                                                                                                                                                                            |                                                                                                                                                      |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| Add a Resource Group (extended)                                                                                                                                                                            |                                                                                                                                                      |
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes.                                                                                                                 |                                                                                                                                                      |
| * Resource Group Name<br><b>Inter-Site Management Policy</b><br>* Participating Nodes from Primary Site<br>Participating Nodes from Secondary Site<br>Startup Policy<br>Failover Policy<br>Fallback Policy | [Entry Fields]<br>[xsiteRG]<br>[ignore] +<br>[jessica] +<br>[maddi] +<br>Online On Home Node Only+<br>Failover To Next Priority+<br>Never Fallback + |

Figure 12-10 Add a Resource Group

## Adding service IP labels into the resource group

To add the service IP labels into the RG, complete the following steps:

1. Use the **smitty sysmirror** fast path, select **Cluster Applications and Resources → Resource Groups → Change/Show Resources and Attributes for a Resource Group**, choose the resource group (xsiteRG in our case), and press Enter.
2. Specify the policies that you want. Choose Service IP Labels/Addresses and press F4 to see a list of the service IP labels that were previously created. Choose both by pressing F7, and then press Enter. After completing the fields, press Enter to add the resources into the RG. Our example is shown in Figure 12-11 on page 515.

| Change/Show All Resources and Attributes for a Resource Group                              |                           |
|--------------------------------------------------------------------------------------------|---------------------------|
| Type or select values in the entry fields.<br>Press Enter AFTER making all wanted changes. |                           |
| <b>[TOP]</b>                                                                               |                           |
| Resource Group Name                                                                        | [Entry Fields]<br>xsiteRG |
| Inter-site Management Policy                                                               | ignore                    |
| Participating Nodes from Primary Site                                                      | jessica                   |
| Participating Nodes from Secondary Site                                                    | maddi                     |
| Startup Policy                                                                             | Online On Home Node Only  |
| Fallover Policy                                                                            | Fallover To Next Priority |
| Fallback Policy                                                                            | Never Fallback            |
| Service IP Labels/Addresses                                                                | [dallaserv ftwserv] +     |
| Application Controller Name                                                                | [] +                      |
| Volume Groups                                                                              | [xsitevg] +               |
| <b>[MORE...34]</b>                                                                         |                           |

Figure 12-11 Adding a service IP into a resource group

## Synchronizing the cluster

To synchronize the cluster, complete the following steps:

1. Use the **smitty sysmirror** fast path and select **Cluster Applications and Resources → Verify and Synchronize Cluster Configuration**.
2. Synchronize and verify.
3. Test the cluster behavior.

## Testing a site-specific IP failover

To display the results of site-specific IP failover tests, we show the **netstat** output before cluster start, after start, and after failover. The initial configuration is shown in Example 12-6.

Example 12-6 Beginning IP configuration

---

```
[jessica:root] / # clcmd netstat -i

NODE maddi

Name Mtu Network Address Ipkts Ierrs Opkts Oerrs Coll
en0 1500 link#2 7a.40.c8.b3.15.2 42474 0 41723 0 0
en0 1500 192.168.100 maddi 42474 0 41723 0 0
en1 1500 link#3 7a.40.c8.b3.15.3 5917 0 4802 0 0
en1 1500 192.168.150 maddi_xd 5917 0 4802 0 0
lo0 16896 link#1 5965 0 5965 0 0
lo0 16896 127 loopback 5965 0 5965 0 0
lo0 16896 loopback 5965 0 5965 0 0

NODE jessica

Name Mtu Network Address Ipkts Ierrs Opkts Oerrs Coll
```

|            |             |                    |                        |             |          |             |          |          |
|------------|-------------|--------------------|------------------------|-------------|----------|-------------|----------|----------|
| en0        | 1500        | link#2             | ee.af.1.71.78.2        | 45506       | 0        | 44229       | 0        | 0        |
| en0        | 1500        | 192.168.100        | jessica                | 45506       | 0        | 44229       | 0        | 0        |
| <b>en1</b> | <b>1500</b> | <b>link#3</b>      | <b>ee.af.1.71.78.3</b> | <b>5685</b> | <b>0</b> | <b>5040</b> | <b>0</b> | <b>0</b> |
| <b>en1</b> | <b>1500</b> | <b>192.168.150</b> | <b>jessica_xd</b>      | <b>5685</b> | <b>0</b> | <b>5040</b> | <b>0</b> | <b>0</b> |
| lo0        | 16896       | link#1             |                        | 15647       | 0        | 15647       | 0        | 0        |
| lo0        | 16896       | 127                | loopback               | 15647       | 0        | 15647       | 0        | 0        |
| lo0        | 16896       | loopback           |                        | 15647       | 0        | 15647       | 0        | 0        |

The text that is in bold text indicates that en1 is configured only with the boot IP address on both nodes. On starting cluster services, because jessica is the primary, it acquires the service address that is specific to the dallas site of dallasserv. The secondary node, maddi, remains unchanged, as shown in Example 12-7.

*Example 12-7 IP configuration after the startup on the primary site*

---

```
[jessica:root] / # clcmd netstat -i
```

| -----        |             |                 |                   |              |          |             |          |          |
|--------------|-------------|-----------------|-------------------|--------------|----------|-------------|----------|----------|
| NODE maddi   |             |                 |                   |              |          |             |          |          |
| Name         | Mtu         | Network         | Address           | Ipkts        | Ierrrs   | Opkts       | Oerrrs   | Coll     |
| en0          | 1500        | link#2          | 7a.40.c8.b3.15.2  | 52447        | 0        | 51059       | 0        | 0        |
| en0          | 1500        | 192.168.100     | maddi             | 52447        | 0        | 51059       | 0        | 0        |
| en1          | 1500        | link#3          | 7a.40.c8.b3.15.3  | 9622         | 0        | 9213        | 0        | 0        |
| en1          | 1500        | 192.168.150     | maddi_xd          | 9622         | 0        | 9213        | 0        | 0        |
| lo0          | 16896       | link#1          |                   | 7683         | 0        | 7683        | 0        | 0        |
| lo0          | 16896       | 127             | loopback          | 7683         | 0        | 7683        | 0        | 0        |
| lo0          | 16896       | loopback        |                   | 7683         | 0        | 7683        | 0        | 0        |
| -----        |             |                 |                   |              |          |             |          |          |
| NODE jessica |             |                 |                   |              |          |             |          |          |
| Name         | Mtu         | Network         | Address           | Ipkts        | Ierrrs   | Opkts       | Oerrrs   | Coll     |
| en0          | 1500        | link#2          | ee.af.1.71.78.2   | 55556        | 0        | 54709       | 0        | 0        |
| en0          | 1500        | 192.168.100     | jessica           | 55556        | 0        | 54709       | 0        | 0        |
| en1          | 1500        | link#3          | ee.af.1.71.78.3   | 10161        | 0        | 8680        | 0        | 0        |
| <b>en1</b>   | <b>1500</b> | <b>10.10.10</b> | <b>dallasserv</b> | <b>10161</b> | <b>0</b> | <b>8680</b> | <b>0</b> | <b>0</b> |
| en1          | 1500        | 192.168.150     | jessica_xd        | 10161        | 0        | 8680        | 0        | 0        |
| lo0          | 16896       | link#1          |                   | 24361        | 0        | 24361       | 0        | 0        |
| lo0          | 16896       | 127             | loopback          | 24361        | 0        | 24361       | 0        | 0        |
| lo0          | 16896       | loopback        |                   | 24361        | 0        | 24361       | 0        | 0        |

We move the RG to the fortworth site by using the System Management Interface Tool (SMIT) fast path **smitty cl\_resgrp\_move\_node.select**. On success, the primary site service IP address, dallasserv, is removed, and the secondary site IP address, ftwserv, is brought online, as shown in Example 12-8.

*Example 12-8 IP address configuration after starting on a remote secondary site*

---

```
[jessica:root] / # clcmd netstat -i
```

| -----      |             |                 |                  |             |          |             |          |          |
|------------|-------------|-----------------|------------------|-------------|----------|-------------|----------|----------|
| NODE maddi |             |                 |                  |             |          |             |          |          |
| Name       | Mtu         | Network         | Address          | Ipkts       | Ierrrs   | Opkts       | Oerrrs   | Coll     |
| en0        | 1500        | link#2          | 7a.40.c8.b3.15.2 | 54634       | 0        | 53164       | 0        | 0        |
| en0        | 1500        | 192.168.100     | maddi            | 54634       | 0        | 53164       | 0        | 0        |
| en1        | 1500        | link#3          | 7a.40.c8.b3.15.3 | 9722        | 0        | 9264        | 0        | 0        |
| <b>en1</b> | <b>1500</b> | <b>10.10.10</b> | <b>ftwserv</b>   | <b>9722</b> | <b>0</b> | <b>9264</b> | <b>0</b> | <b>0</b> |
| en1        | 1500        | 192.168.150     | maddi_xd         | 9722        | 0        | 9264        | 0        | 0        |

| 1o0          | 16896 | link#1      |                 | 8332  | 0     | 8332  | 0     | 0    |
|--------------|-------|-------------|-----------------|-------|-------|-------|-------|------|
| 1o0          | 16896 | 127         | loopback        | 8332  | 0     | 8332  | 0     | 0    |
| 1o0          | 16896 | loopback    |                 | 8332  | 0     | 8332  | 0     | 0    |
| <hr/>        |       |             |                 |       |       |       |       |      |
| NODE jessica |       |             |                 |       |       |       |       |      |
| <hr/>        |       |             |                 |       |       |       |       |      |
| Name         | Mtu   | Network     | Address         | Ipkts | Ierrs | Opkts | Oerrs | Coll |
| en0          | 1500  | link#2      | ee.af.1.71.78.2 | 57916 | 0     | 57041 | 0     | 0    |
| en0          | 1500  | 192.168.100 | jessica         | 57916 | 0     | 57041 | 0     | 0    |
| en1          | 1500  | link#3      | ee.af.1.71.78.3 | 10262 | 0     | 8730  | 0     | 0    |
| en1          | 1500  | 192.168.150 | jessica_xd      | 10262 | 0     | 8730  | 0     | 0    |
| 1o0          | 16896 | link#1      |                 | 25069 | 0     | 25069 | 0     | 0    |
| 1o0          | 16896 | 127         | loopback        | 25069 | 0     | 25069 | 0     | 0    |
| 1o0          | 16896 | loopback    |                 | 25069 | 0     | 25069 | 0     | 0    |

---

## 12.5 Understanding the netmon.cf file

The netmon.cf file provides more network monitoring functions to help you determine whether an adapter is available or not. There are times where clients with virtual adapters are not able to determine whether the physical adapter in the Virtual I/O Server (VIOS) is active and able to accept traffic. The solution is a new entry format in the netmon.cf file to define network addresses that are pinged to determine whether the adapter is available. The new format of the entries in the file is described in this section.

There is a newer option available (poll\_uplink0), which is recommended to address this issue. For more information, see 12.6, “Using poll\_uplink” on page 520.

### 12.5.1 The netmon.cf format for virtual Ethernet environments

The netmon.cf file addresses problems that can occur with adapters in a virtual I/O environment.

#### The problem that netmon addresses

This netmon.cf format was added to support PowerHA in a virtual I/O environment. PowerHA customers that use VIOS within their clusters have experienced problems with specific scenarios where an entire central processing complex (CPC) is unplugged from the network, but the PowerHA node within it does not detect a local adapter-down event because traffic being passed between the VIOS clients looks like normal external traffic from the perspective of the LPAR’s operating system.

There is already a general guideline against having two PowerHA nodes in the same cluster that use the same VIOS because heartbeats can be passed between the nodes through the server even when no real network connectivity exists. The problem that is addressed by netmon.cf is not the same as that issue, although similarities exist.

In PowerHA, heartbeating is used as a reliable means of monitoring an adapter’s state over a long period. When heartbeating is not working, a decision must be made about whether the local adapter has failed. Does the neighbor have a problem or is it something between them? The local node must act only if the local adapter is the problem. If its own adapter is good, then we assume that it is still reachable by other clients regardless of the neighbor’s state (because the neighbor is responsible for acting on its local adapters failures).

The decision of which adapter failed, local or remote, is made based on whether any network traffic can be seen on the local adapter by using the inbound byte count of the interface. Where VIOS is involved, this test becomes unreliable because there is no way to distinguish whether inbound traffic came in from the VIOS connection to the outside world or from a neighboring virtual I/O client. To the LPAR, a VIOS's virtual adapters can be indistinguishable from a real adapter.

## Problem resolution

The netmon.cf format was added to help in virtual environments. This new format allows customers to declare that an adapter should be considered active only if it can ping a set of specified targets.

**Important:** For this fix to be effective, the customer *must* select targets that are outside the VIOS environment and not reachable by hopping from one VIOS to another one. Cluster verification will not determine whether they are valid or not.

## Configuring netmon.cf

The netmon.cf file must be placed in the /usr/es/sbin/cluster directory on all cluster nodes. Up to 32 targets can be provided for each interface. If *any* specific target is pingable, the adapter is considered “up.”

Targets are specified by using the existing netmon.cf configuration file with this new format, as shown in Example 12-9.

*Example 12-9 The netmon.cf format*

---

```
!REQD <owner> <target>

Parameters:

!REQD :An explicit string; it *must* be at the beginning of the line (no leading spaces).

<owner> : The interface this line is intended to be used by; that is, the code monitoring
the adapter that is specified here will determine its own up or down status by whether it
can ping any of the targets (below) specified in these lines. The owner can be specified as
a hostname, IP address, or interface name. In the case of hostname or IP address, it *must*
refer to the boot name or IP (no service aliases). In the case of a hostname, it must be
resolvable to an IP address or the line is ignored. The string "!ALL" specifies all
adapters.

<target> : The IP address or hostname you want the owner to try to ping. As with normal
netmon.cf entries, a hostname target must be resolvable to an IP address to be usable.
```

---

**Attention:** The *traditional* format of the netmon.cf file is not valid in PowerHA 7 and later, so it is ignored. Only the !REQD lines are used.

The order from one line to the other is unimportant. Commented lines are allowed on or between lines, and they are ignored. With IBM AIX 7.1 with Technology Level (TL) 4 or earlier, you can specify the same owner entry up to 32 different lines in the netmon.cf file (any more lines are ignored). In IBM AIX 7.1 TL 4 and later and AIX 7.2 and later, only the last five entries for an owner entry are considered. For an owner adapter that is listed on more than one line, the adapter is considered available if it can ping any of the provided targets.

## 12.5.2 netmon.cf examples

The following examples explain the syntax:

- ▶ In Example 12-10, interface en2 is considered “up” only if it can ping either 100.12.7.9 or 100.12.7.10.

---

*Example 12-10 netmon.cf example 1*

---

```
!REQD en2 100.12.7.9
!REQD en2 100.12.7.10
```

---

- ▶ In Example 12-11, the adapter that owns host1.ibm is considered “up” only if it can ping 100.12.7.9 or whatever host4.ibm resolves to. The adapter owning 100.12.7.20 is considered “up” only if it can ping 100.12.7.10 or whatever host5.ibm resolves to. It is possible that 100.12.7.20 is the IP address for host1.ibm.

---

*Example 12-11 netmon.cf example 2*

---

```
!REQD host1.ibm 100.12.7.9
!REQD host1.ibm host4.ibm
!REQD 100.12.7.20 100.12.7.10
!REQD 100.12.7.20 host5.ibm
```

---

- ▶ In Example 12-12, all adapters are available only if they can ping the 100.12.7.9, 110.12.7.9, or 111.100.1.10 IP addresses. The en1 owner entry has an extra target of 9.12.11.10.

---

*Example 12-12 netmon.cf example 3*

---

```
!REQD !ALL 100.12.7.9
!REQD !ALL 110.12.7.9
!REQD !ALL 111.100.1.10
!REQD en1 9.12.11.10
```

---

All adapters are considered up only if they can ping 100.12.7.9, 110.12.7.9, or 111.100.1.10. Interface en1 has one extra target: 9.12.11.10. In this example, having any traditional lines is pointless because all the adapters use the new method.

## 12.5.3 Implications

The following implications of the new format should be considered:

- ▶ Any interfaces that are not included as an *owner* of one of the !REQD lines in the netmon.cf continue to behave in the old manner, even if you are using this new function for other interfaces.
- ▶ This format does *not* change heartbeating behavior in any way. It changes only how the decision is made regarding whether a local adapter is up or down. This new logic is used in these situations:
  - Upon startup, before heartbeating rings are formed.
  - During heartbeat failure, when contact with a neighbor is initially lost.
  - During periods when heartbeating is not possible, such as when a node is the only one up in the cluster.

- ▶ Invoking the format changes the definition of a *good* adapter from “Am I able to receive *any* network traffic?” to “Can I successfully ping certain addresses?” regardless of how much traffic is seen:
  - An adapter is inherently more likely to be falsely considered down because the second definition is more restrictive.
  - If you find that you must leverage this new function, be as generous as possible with the number of targets that you provide for each interface.

## 12.6 Using poll\_uplink

As described in 12.5, “Understanding the netmon.cf file” on page 517, in PowerHA, network down detection is performed by CAA. CAA by default checks for IP traffic and for the link status of an interface. The link state and IP packet count might give the appearance that all is well, when in fact there is no communication to outside the physical server. One solution to this problem is to use the netmon.cf file, but a better option to address this issue is **poll\_uplink**.

As a best practice, use **poll\_uplink** with Virtual Ethernet and Shared Ethernet Adapter (SEA) backed configurations because it is better at determining whether an interface is up or down.

To use the **poll\_uplink** option, you must have the following versions and settings:

- ▶ VIOS 2.2.3.4 or later installed in all related VIOSs.
- ▶ The LPAR must be at AIX 7.1 TL3 SP3 or later.
- ▶ The option **poll\_uplink** must be set in the LPAR on the virtual entX interface.

The option **poll\_uplink** can be defined directly on the virtual interface for SEA failover or the Etherchannel device. To enable **poll\_uplink**, run the following command:

```
chdev -l entX -a poll_uplink=yes -P
```

**Important:** Restart the LPAR to activate **poll\_uplink**.

The following options are available:

- ▶ **poll\_uplink (yes, no)**
- ▶ **poll\_uplink\_int (100 milliseconds (ms) - 5000 ms)**

To display the settings, run the **lsattr -E1 entX** command. Example 12-13 shows the default settings for **poll\_uplink**.

*Example 12-13 Poll\_uplink default adapter value*

---

```
lsattr -E1 ent0 | grep "poll_up"
poll_uplink no Enable Uplink Polling True
poll_uplink_int 1000 Time interval for Uplink Polling True
```

---

Use the **entstat** command to check the **poll\_uplink** status and verify that it is enabled. Example 12-14 on page 521 shows an excerpt of the **entstat** command output in an LPAR where **poll\_uplink** is set to no.

---

*Example 12-14 Poll\_uplink disabled entstat output*

---

```
entstat -d ent0

ETHERNET STATISTICS (en0) :
Device Type: Virtual I/O Ethernet Adapter (1-lan)
...
General Statistics:

No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 20000
Driver Flags: Up Broadcast Running
Simplex 64BitSupport ChecksumOffload
DataRateSet VIOENT
...
LAN State: Operational
...
```

---

Example 12-15 shows the **entstat** command output on a system where **poll\_uplink** is enabled and where all physical links that are related to this virtual interface are up. The text in bold shows the additional displayed content:

- ▶ **VIRTUAL\_PORT**
- ▶ **PHYS\_LINK\_UP**
- ▶ **Bridge Status: Up**

---

*Example 12-15 Poll\_uplink enabled entstat output when physical link is up*

---

```
entstat -d ent0

ETHERNET STATISTICS (en0) :
Device Type: Virtual I/O Ethernet Adapter (1-lan)
...
General Statistics:

No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 20000
Driver Flags: Up Broadcast Running
Simplex 64BitSupport ChecksumOffload
DataRateSet VIOENT VIRTUAL_PORT <-----
PHYS_LINK_UP <-----
...
LAN State: Operational
Bridge Status: Up <-----
```

---

Example 12-16 shows the **entstat** command output on a system where **poll\_uplink** is enabled and where all physical links that are related to this virtual interface are down. The **PHYS\_LINK\_UP** no longer displays, and the Bridge Status changes from Up to Unknown.

---

*Example 12-16 Poll\_uplink enabled entstat output when the physical link is down*

---

```
entstat -d ent0

ETHERNET STATISTICS (en0) :
Device Type: Virtual I/O Ethernet Adapter (1-lan)
...
General Statistics:

```

```
No mbuf Errors: 0
Adapter Reset Count: 0
Adapter Data Rate: 20000
Driver Flags: Up Broadcast Running
 Simplex 64BitSupport ChecksumOffload
 DataRateSet VIOENT VIRTUAL_PORT
...
LAN State: Operational
Bridge Status: Unknown <-----
```

---

## 12.7 Understanding the clhosts file

The clhosts file contains IP address information that helps to enable communication among monitoring daemons on clients and within the PowerHA cluster nodes. The tools that use this file include **clinfoES** and **clstat**. The file is on all PowerHA cluster servers and clients in the /usr/es/sbin/cluster/etc/ directory.

When a monitor daemon starts, it reads the /usr/es/sbin/cluster/etc/clhosts file to determine which nodes are available for communication. Therefore, it is important for these files to be in place when trying to use the monitoring tools from a client outside the cluster. When the server portion of PowerHA is installed, the clhosts file is updated on the cluster nodes with the loopback address (127.0.0.1). The contents of the file within each cluster node typically contains only the following line immediately after installation:

```
127.0.0.1 # HACMP/ES for AIX
```

### 12.7.1 Creating the clhosts file

PowerHA automatically generates the clhosts file that is needed by clients when you perform a verification with the automatic corrective action feature enabled. The verification creates a /usr/es/sbin/cluster/etc/clhosts.client file on all cluster nodes. The file is similar to Example 12-17.

*Example 12-17 The clhosts.client file*

---

```
$clverify$
/usr/es/sbin/cluster/etc/clhosts.client Created by HACMP Verification / Synchron
ization Corrective Actions
Date Created: 11/19/2022 at 16:42:10
#
10.10.10.52 #ftwserv
10.10.10.51 #dallasserv

192.168.150.51 #jessica_xd
192.168.150.52 #maddi_xd

192.168.150.53 #ashley_xd
192.168.100.53 #ashley
```

---

All the addresses are imported, including the boot, service, and persistent IP labels. Before using any of the monitor utilities from a client node, the clhosts.client file must be copied over to all clients as /usr/es/sbin/cluster/etc/clhosts. Remove the client extension when you copy the file to the client nodes.

**Important:** The `c1hosts` file on a client must never contain `127.0.0.1`, loopback, or `localhost`.

### Using `c1stat` on a client requires a `c1hosts` file

When running the `c1stat` utility from a client, the `c1infoES` daemon obtains its cluster status information from the server-side SNMP and populates the PowerHA Management Information Base (MIB) on the client side. It cannot communicate with the daemon, and it reports that it cannot find any clusters if it has no available `c1hosts` file.

In this type of environment, implementing a `c1hosts` file on the client is critical. This file provides the `c1infoES` daemon with the addresses to attempt communication with the SNMP process running on the PowerHA cluster nodes.





# Cross-Site Logical Volume Manager stretched campus cluster

This chapter provides details about how to set up a cross-site Logical Volume Manager (LVM) mirroring cluster by using PowerHA SystemMirror 7.2.7 Standard Edition. This scenario is a common, storage-independent, and site-based cluster scenario.

This chapter contains the following topics:

- ▶ Cross-site LVM mirroring overview
- ▶ Test environment
- ▶ Configuring a cross-site LVM cluster
- ▶ Testing

## 13.1 Cross-site LVM mirroring overview

This section describes a disaster recovery (DR) solution that is based on AIX LVM mirroring and a stretched PowerHA cluster. It is built from the same components that are used for local cluster solutions with storage area network (SAN)-attached storage. Cross-site LVM mirroring replicates data across the SAN between the disk subsystems at separate sites, and PowerHA provides automated failover in a failure. This solution can provide an RPO of zero, and RTO of mere minutes. The biggest determining factor in recovery time is application recovery and restart time.

Remote disks can be combined into a volume group (VG) through the AIX LVM, and this VG can be imported to nodes at different sites. You can create logical volumes (LVs) and set up an LVM mirror with a copy at each site. Although LVM mirroring supports up to three copies, PowerHA supports only two sites. It is still possible to have three LVM copies with two LVM copies (even when you use two servers) at one site and one remote copy at another site, although it is an uncommon situation.

Although it is common to have the same storage type at each location, it is not a requirement. This similarity is an advantage for this type of configuration because it is storage type neutral. While the storage is supported for SAN attachment to AIX and provides adequate performance, it is most likely a valid candidate to use in this configuration.

The main difference between local clusters and cluster solutions with cross-site mirroring is as follows:

- ▶ In local clusters, all nodes and storage subsystems are in the same location.
- ▶ With cross-site mirrored clusters, nodes and storage subsystems are at different sites.
- ▶ Each site has at least one cluster node and one storage subsystem with all the necessary IP and SAN connectivity, similar to a local cluster.
- ▶ Use ignore for the resource group (RG) inter-site management policy.

The increased availability of metropolitan area networks (MANs) in recent years has made this solution more feasible and popular.

This solution offers automation of AIX LVM mirroring within SAN disk subsystems between different sites. It also provides automatic LVM mirroring synchronization and disk device activation when, after a disk or site failure, a node or disk becomes available again.

Each node in a cross-site LVM cluster accesses all storage subsystems. The data availability is ensured through the LVM mirroring between the volumes on different storage subsystems on different sites.

In a complete site failure, PowerHA performs a takeover of the resources to the secondary site according to the cluster policy configuration. It activates all defined VGs from the surviving mirrored copy. If one storage subsystem fails, I/O might experience a temporary delay, but it continues to access data from the active mirroring copy on the surviving disk subsystem.

PowerHA drives automatic LVM mirroring synchronization. After the failed site joins the cluster, it automatically fixes removed and missing physical volumes (PVs) (PV states removed and missing) and synchronizes data. However, automatic synchronization is not possible for all cases. But, Cluster Single Point of Control (C-SPOC) can be used to synchronize the data from the surviving mirrors to stale mirrors after a disk or site failure.

### 13.1.1 Requirements

The following requirements must be met to ensure data integrity and the correct PowerHA reaction in a site or disk subsystem failure:

- ▶ A server and storage unit at each of the two sites.
- ▶ SAN and local area network (LAN) connectivity across or between sites. Redundant infrastructure both within and across sites is a best practice.
- ▶ PowerHA Standard Edition is supported (allows stretched clusters and supports site creation).
- ▶ A 2-site stretched cluster must be configured.
- ▶ The **force varyon** attribute for the RG must be set to true.
- ▶ The LVs' allocation policy must be set to superstrict to ensure that LV copies are allocated on different volumes, and the primary and secondary copy of each logical partition (LPAR) is allocated on disks at different sites).
- ▶ The LV mirrored copies must be allocated on separate volumes that are on different disk subsystems at the different sites.
- ▶ Use mirror pools to help define the disks at each site.

When adding more storage space, for example, increasing the size of the mirrored file system, it is necessary to ensure that the new LPARs are allocated on different volumes and different disk subsystems according to the requirements above. For this task, increase the LV first with the appropriate volume selections, and then increase the file system, preferably by using C-SPOC.

### 13.1.2 Planning considerations

Here we describe some considerations regarding SAN setup, Fibre Channel connections, and the LAN environment. The considerations and limitations are based on the technologies and protocols that are used for cross-site mirroring cluster implementation.

The SAN network can be expanded beyond the original site by using advanced technology. Here is an example of what kind of technology can be used for expansion. This list is not exhaustive.

- ▶ FCIP router.
- ▶ Wave division multiplexing (WDM) devices:
  - Coarse Wavelength Division Multiplexing (CWDM), which is the less expensive component of the WDM technology.
  - Dense Wave length Division Multiplexing (DWDM).

The infrastructure should be resilient to failures by providing redundant components. Also, the SAN interconnection must provide sufficient bandwidth to allow for adequate performance for the synchronous-based mirroring that LVM provides. Distance between sites is important for latency that also can result in performance issues. The LVM mirroring itself creates extra I/O, which has performance implications. We describe some of these implications and provide best practices about how to deal with them by using available settings in the rest of this section.

### **Repository disks**

Similar to a local cluster, a stretched cross-site LVM mirrored cluster consists of only a single repository disk. Decide which site should contain the repository disk. If the disk is at the primary site and the primary site goes down, a failover can and should still succeed. However, it is a best practice to define a backup repository disk to the cluster at the opposite site from the primary repository disk. In a primary site failure, the repository disk is taken over with the backup repository disk by using the automatic repository replacement feature within PowerHA.

### **Mirror pools**

Although technically not a requirement, it is a best practice to use the AIX LVM capability of mirror pools. Using mirror pools correctly helps to both create and maintain copies across separate storage subsystems by ensuring a separate and complete copy of all data at each site.

Mirror pools make it possible to divide the PVs of a VG into separate pools. A mirror pool is made up of one or more PVs. Each PV can belong only to one mirror pool at a time. When creating an LV, each copy of the LV that is created can be assigned to a mirror pool. LV copies that are assigned to a mirror pool allocate only partitions from the PVs in that mirror pool. With this approach, you can restrict the disks that an LV copy can use. Without mirror pools, the only way to restrict which PV is used for allocation when creating or extending an LV is to use a map file.

### **Tips for faster disk failure detection**

AIX disk storage subsystem failure detection can be accelerated by changing the attributes of hdisks and interfaces. Most of these changes are applicable, although as described in Table 13-1, some changes are applicable only to certain disk storage subsystems. For more information, consult with the storage vendor.

*Table 13-1 Device type, device name, device attributes, and settings*

| Device type                 | Device name | Device attribute       | Setting     |
|-----------------------------|-------------|------------------------|-------------|
| Virtual SCSI                | vscsi#      | <b>vscsi_err_recov</b> | fast_fail   |
| Fibre Channel (FC) SCSI I/O | fscsi#      | <b>dyntrk</b>          | yes         |
|                             |             | <b>fc_err_recover</b>  | fast_fail   |
| Disk                        | hdisk#      | <b>algorithm</b>       | round_robin |
|                             |             | <b>hcheck_interval</b> | 30          |
|                             |             | <b>timeout_policy</b>  | fail_path   |
| DS8000 specific             |             | <b>FC3_REC</b>         | true        |

**Tip:** For more information, see [IBM AIX Multi Path \(MPIO\) configuration best practices](#).

AIX disk storage subsystem failure detection is prolonged in a redundant virtual SCSI environment, which is one of many reasons why N\_Port ID Virtualization (NPIV) is a better choice than virtual SCSI to minimize the time that is required for AIX to detect and recover from a disk storage subsystem failure in a virtualized environment.

The number of paths to each hdisk or logical unit number (LUN) affects the time that is required to detect and recover from a disk storage subsystem failure. The more paths, the longer recovery takes. As a best practice, use no more than two paths per server port to access a LUN. Two server ports per LUN are sufficient unless I/O bandwidth requirements are high.

When a disk storage subsystem fails, AIX must detect the failure of each LUN that is presented by the subsystem. The more LUNs that there are, the more potential application write I/O stalls there are while AIX detects a LUN failure. The number of LUNs are dictated by the LUN size and the disk storage capacity that is required. Because many I/O requests can be driven simultaneously to a single hdisk or LUN (it is limited by the hdisk's queue\_depth attribute), use fewer larger LUNs compared to smaller LUNs.

But too few LUNs can sometimes lead to a performance bottleneck. The AIX hdisk device driver is single-threaded, which limits the number of IOPS that AIX can drive to a single hdisk. Therefore, as a best practice, do not drive more than 5,000 IOPS to a single hdisk.

Usually, IOPS to a single hdisk or LUN is constrained by disk storage subsystem performance before the hdisk IOPS limit is reached, but the limit can be exceeded on LUNs on an IBM FlashSystem server or on solid-state disk drives.

If the anticipated workload on a VG is such that the hdisk IOPS limit might be exceeded, create the VG on smaller hdisks or LUNs.

If an anticipated workload is such that it tends to drive I/O to only one file or file system at a time, especially when I/O requests are sequential, you can help AIX process failures of multiple LUNs simultaneously by striping LVs (including the underlying file systems) across hdisks or LUNs by using the -S flag on the `mk1v` command. It does *not* help to configure LVs with physical partition (PP) striping by specifying -e x on the `mk1v` command. If PP striping is used with such a workload, AIX might still process a disk storage subsystem failure one LUN at a time.

## Pros and cons

Here is a list of reasons why you might want to implement cross-site LVM mirroring:

- ▶ It is inexpensive. LVM mirroring is a no-charge feature of AIX.
- ▶ It is storage-neutral, which means that it does not require special storage to use.
- ▶ It is simple to implement. Most AIX admins are knowledgeable about LVM mirroring.
- ▶ Unlike most storage replication offerings, both copies have full host read/write access.

Here is a list of reasons why you might *not* want to implement cross-site LVM mirroring:

- ▶ It is specific to AIX. There is no heterogeneous operating system support.
- ▶ It provides synchronous replication only.
- ▶ It cannot copy raw disks.
- ▶ It presents potential system performance implications because it doubles the write I/Os.
- ▶ There is no way to define primary (source) or secondary (target) copies, although you can specify a preferred read option for storage like flash.
- ▶ Due to AIX I/O hang times in a copy loss, it still might not prevent application failure.
- ▶ Quorum is disabled, and forced varyon of the VG is enabled. This configuration can lead to data inconsistencies, so you must carefully plan the recovery.
- ▶ Like most data replication, it is also good at copying bad data. It does not eliminate the need for backup and backout procedures.

**Important:** Even with redundant components, most environments, such as cross-site LVM mirroring, can handle only one failure. That failure must be corrected before another failure occurs. A series of rolling failures might still result in an outage or data corruption.

## 13.2 Test environment

Figure 13-1 shows an overview of our test environment. It is a 2-site, 3-node cluster with storage at each site providing one LVM copy.

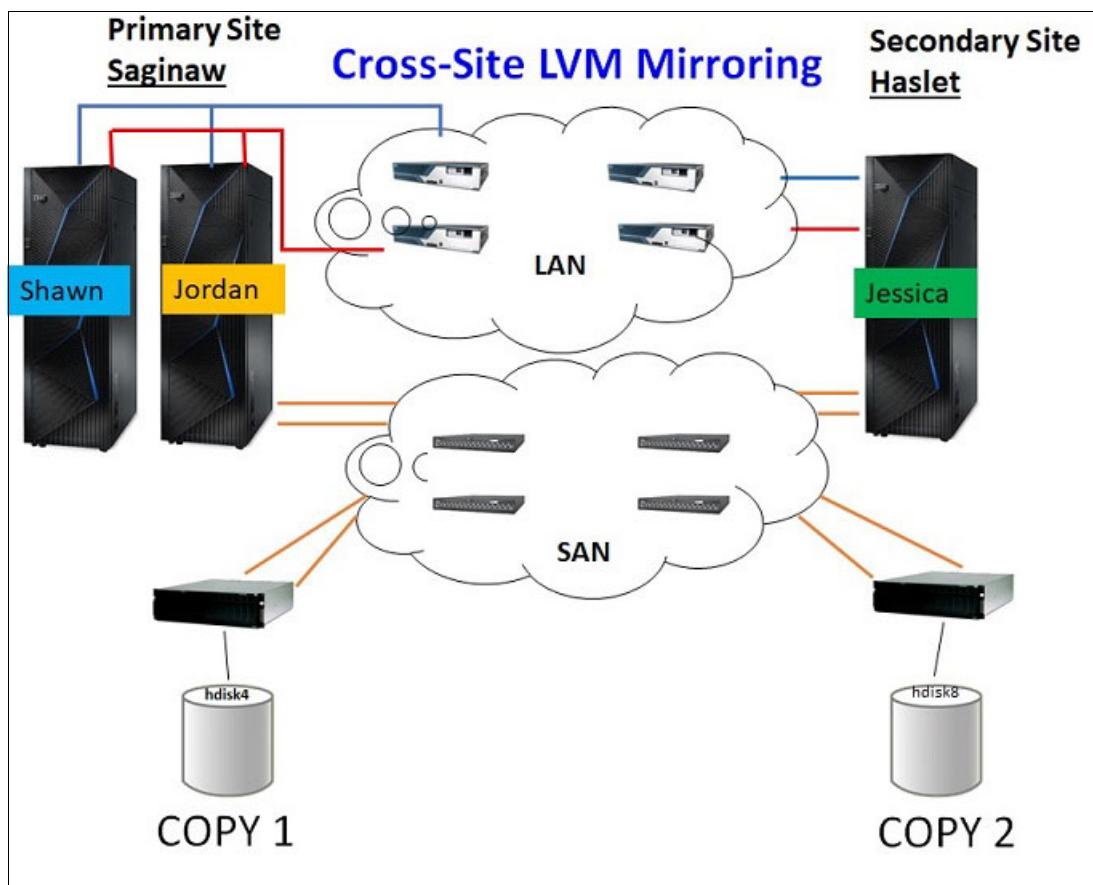


Figure 13-1 Test environment

## 13.3 Configuring a cross-site LVM cluster

This section describes how to configure a cross-site LVM cluster.

### 13.3.1 Topology creation

Create a 2-site (Saginaw and Haslet), 3-node (Shawn, Jordan, and Jessica) cluster by using the SystemMirror User Interface (SMUI) as follows:

1. Log in to the SMUI server.
2. Select **Health Summary** → **Create a New Cluster** in the left navigation pane, as shown in Figure 13-2.

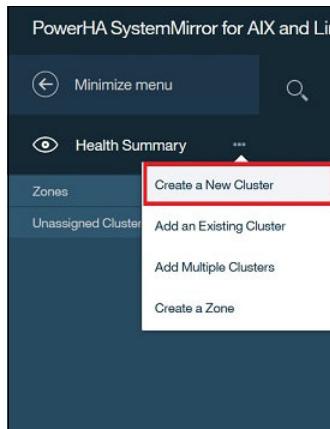


Figure 13-2 SMUI: Create a New Cluster

- Enter the hostname and login credentials of one of the cluster nodes. In our scenario, it is node shawn, as shown in Figure 13-3. Click **Continue**.

PowerHA SystemMirror for AIX and Linux

Hello root

### Create a Cluster

1. Node Authentication → 2. Cluster Settings →

3. Assign Nodes → 4. Summary

Specify a node for the new cluster, along with valid login credentials for that node. The node that you specify is used to collect information about the new cluster environment.  
\*Required field

Hostname or IP Address\*  
shawn

User ID (root access required)\*  
root

Select authentication type\*

Password    Private key file  
 Private key file with passphrase

Password\*  
\*\*\*\*\*

Do you want to enable GUI server backup communication? ?

Cancel Continue

Figure 13-3 SMUI: First node login

- Enter the cluster name and select **Stretched** for the cluster type. In our scenario, the cluster name is xlvm\_cluster, as shown in Figure 13-4. Click **Continue**.

PowerHA SystemMirror for AIX and Linux

Hello root

3. Assign Nodes → 4. Assign Repository Disks → 5. Summary

You must provide a cluster name and choose the cluster type for a node.  
\*Required field

Cluster name\*  
xlvm\_cluster

Select the Type of Cluster\* ?

Standard  
 Stretched  
 Linked

Back Continue

Figure 13-4 SMUI: Cluster name and type

5. Enter the site names and nodes per site, as shown in Figure 13-5. Click **Continue**.

The screenshot shows the 'Define the new cluster settings' screen in the PowerHA SystemMirror for AIX and Linux SMUI. It displays two sections for creating sites and adding nodes.

**Create Site Name 1\***

- Enter hostname\*:
- Assign custom name:
- Persistent IP Address:  Select ▾

|   |        |                          |          |                                        |
|---|--------|--------------------------|----------|----------------------------------------|
| 1 | Shawn  | <input type="checkbox"/> | Select ▾ | <span style="font-size: 2em;">-</span> |
| 2 | Jordan | <input type="checkbox"/> | Select ▾ | <span style="font-size: 2em;">-</span> |

**Create Site Name 2\***

- Enter hostname\*:
- Assign custom name:
- Persistent IP Address:  Select ▾

|   |         |                          |          |
|---|---------|--------------------------|----------|
| 1 | jessica | <input type="checkbox"/> | Select ▾ |
|---|---------|--------------------------|----------|

**Buttons:**

- Back
- Continue

Figure 13-5 SMUI: Add sites and nodes per site

- Choose a repository disk by selecting the checkbox next to the hdisk name, as shown in Figure 13-6. A list of available shared disks is displayed. You can hover the cursor over each disk and see the physical volume identifier (PVID) and size to verify that you are picking the correct one. When you are done, click **Continue**.

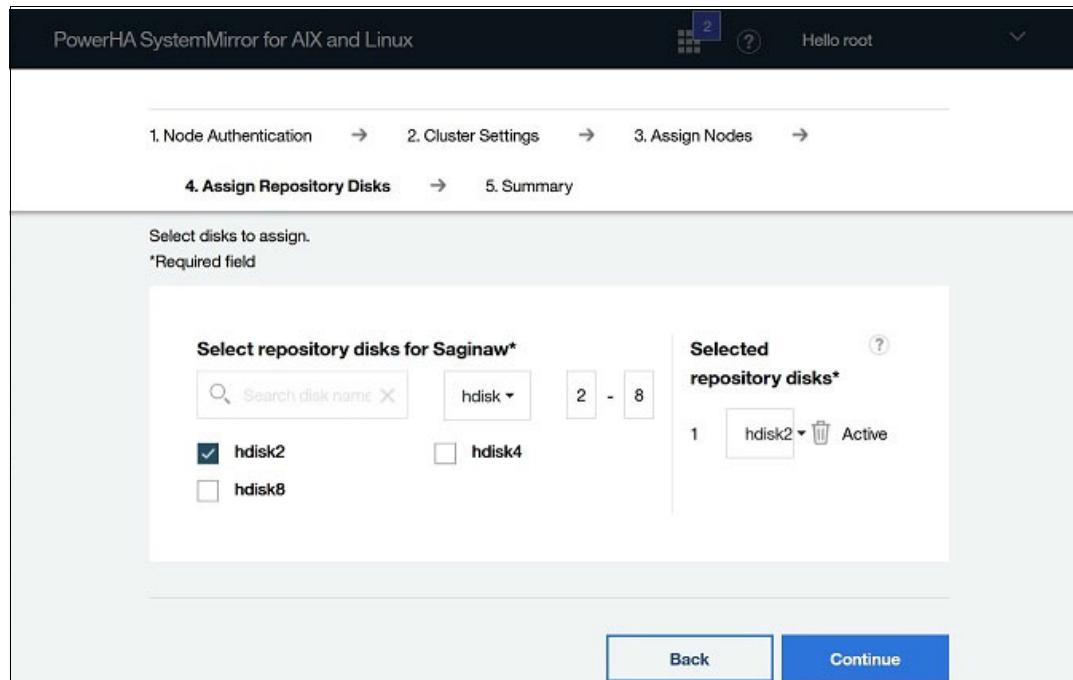


Figure 13-6 SMUI: Assign a repository disk

- Now, a cluster summary is presented, as shown in Figure 13-8 on page 535. Review the details. Confirm them by clicking **Submit** to complete.

After successful creation, the cluster is displayed in the SMUI as unassigned and offline, as shown in Figure 13-7. Only the base Cluster Aware AIX (CAA) cluster has been created. The resources, service IP, VG, and RG still must be created.

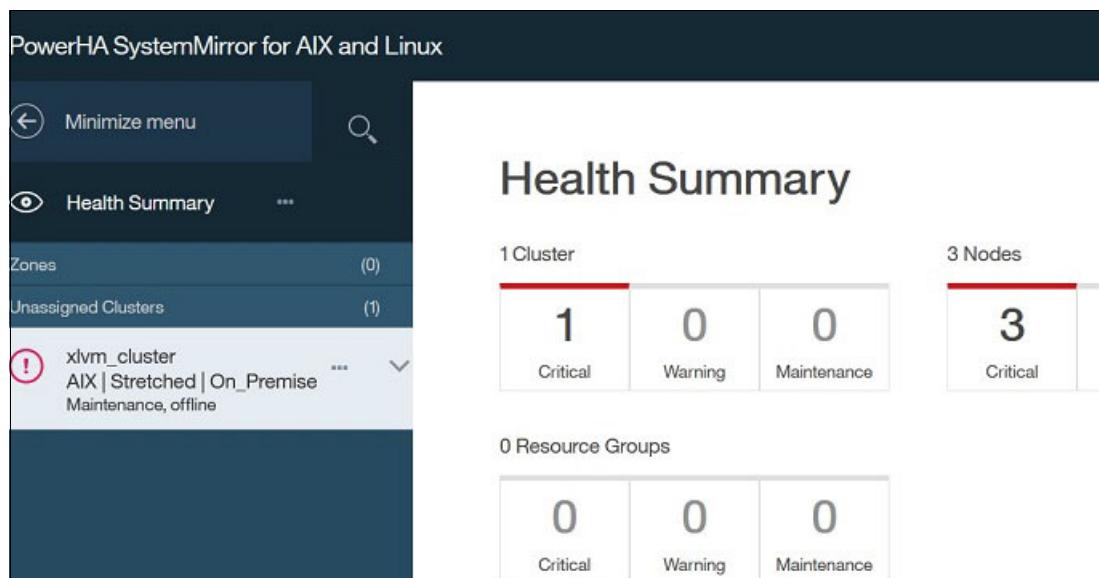


Figure 13-7 SMUI: Base cluster completed

Figure 13-8 shows a summary of your base cluster creation.

The screenshot shows the SMUI Cluster summary screen. At the top, it displays "PowerHA SystemMirror for AIX and Linux" and "Hello root". Below this, the title "Summary of your specifications for the new cluster" is followed by "Cluster Settings".

| Cluster name* | Cluster Type | Communication Type |
|---------------|--------------|--------------------|
| xlvm_cluster  | Stretched    | Unicast            |

Below the cluster settings is the "Node Authentication" section:

| Hostname or IP Address | User ID | Authentication Type |
|------------------------|---------|---------------------|
| shawn                  | root    | Password            |

The "Assigned Nodes" section lists two groups:

- Saginaw**: Nodes shawn, jordan
- Haslet**: Nodes jessica

The "Assigned Repository Disk" section lists hdisk2.

At the bottom right are "Back" and "Submit" buttons.

Figure 13-8 SMUI: Cluster summary

### 13.3.2 Resource group creation

Typically, with the System Management Interface Tool (SMIT) or `c1mgr`, it is common to create resources first and then the RG. The exact order is not important if the result is the same. In the SMUI, you create an RG first, and then create or add resources for it. Only the common basic resources for Service IP, VGs, and application controllers can be created from the SMUI. For anything else, you must use SMIT or `c1mgr`.

To create an RG by using the SMUI, complete the following steps:

1. Click the three dots next to the cluster and select **Create a Resource Group**, as shown in Figure 13-9.

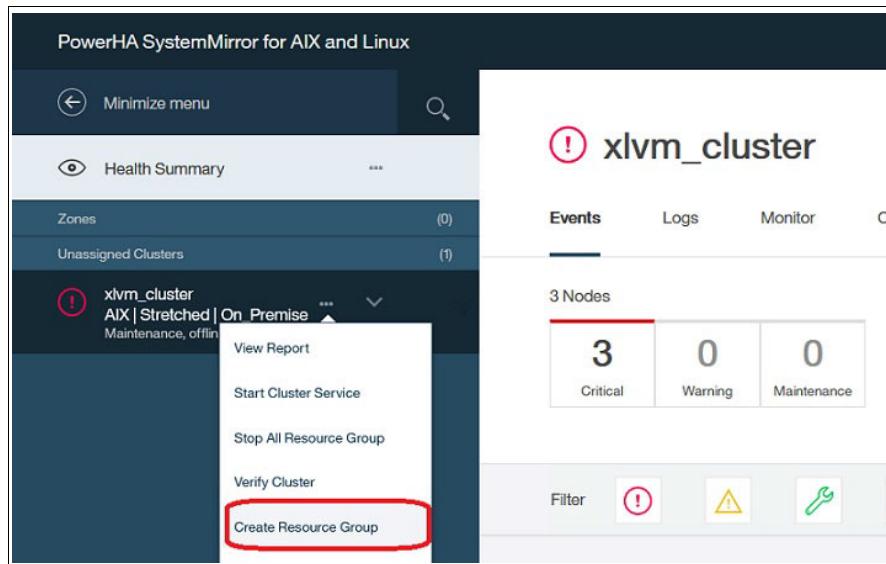


Figure 13-9 SMUI: Creating a resource group

2. In the next window, complete all the options and click **Continue**. For our scenario, we used the following information, which is shown in Figure 13-10 on page 537:
  - a. Resource Group Name: xsiterg
  - b. Make Primary Site: Saginaw
  - c. Add nodes in order of: shawn and jordan
  - d. For site Haslet add node: jessica
  - e. Inter-Site management: ignore

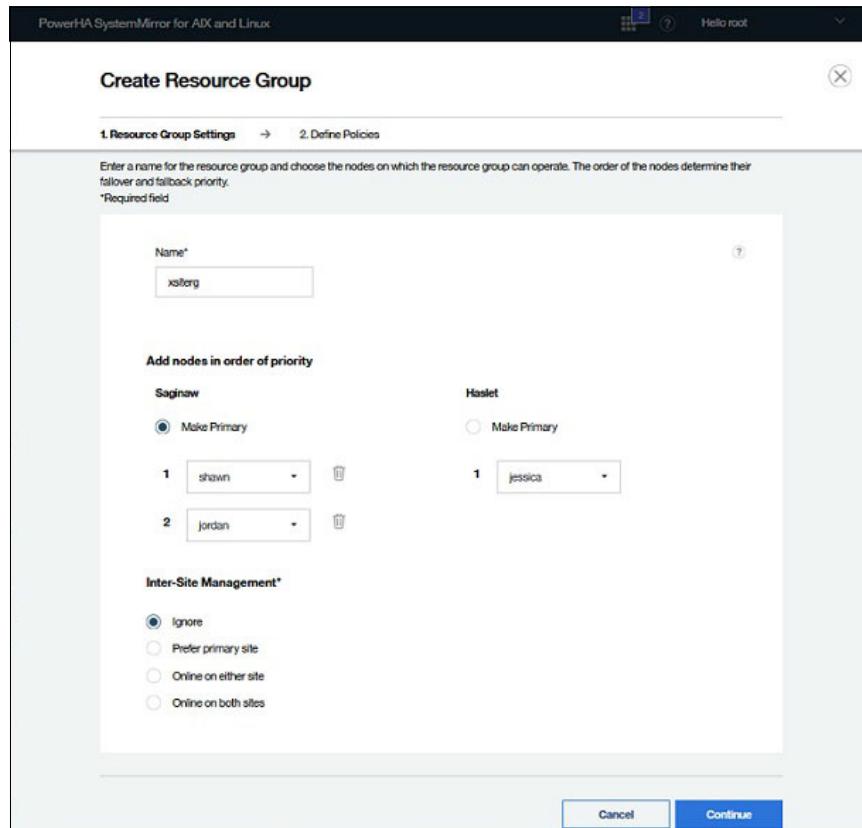


Figure 13-10 SMUI: Add participating nodes to a resource group

**Note:** For inter-site management, we choose ignore because it is what should be used with cross-site LVM configurations.

- Choose the Startup, Failover and Fallback policies, as shown in Figure 13-11, and then click **Create**. For our scenario, they are as follows:
  - Startup: Online on Home Node
  - Failover: Move the resource group to the next available node in order of priority.
  - Fallback: Avoid another outage and never return to a previous node.

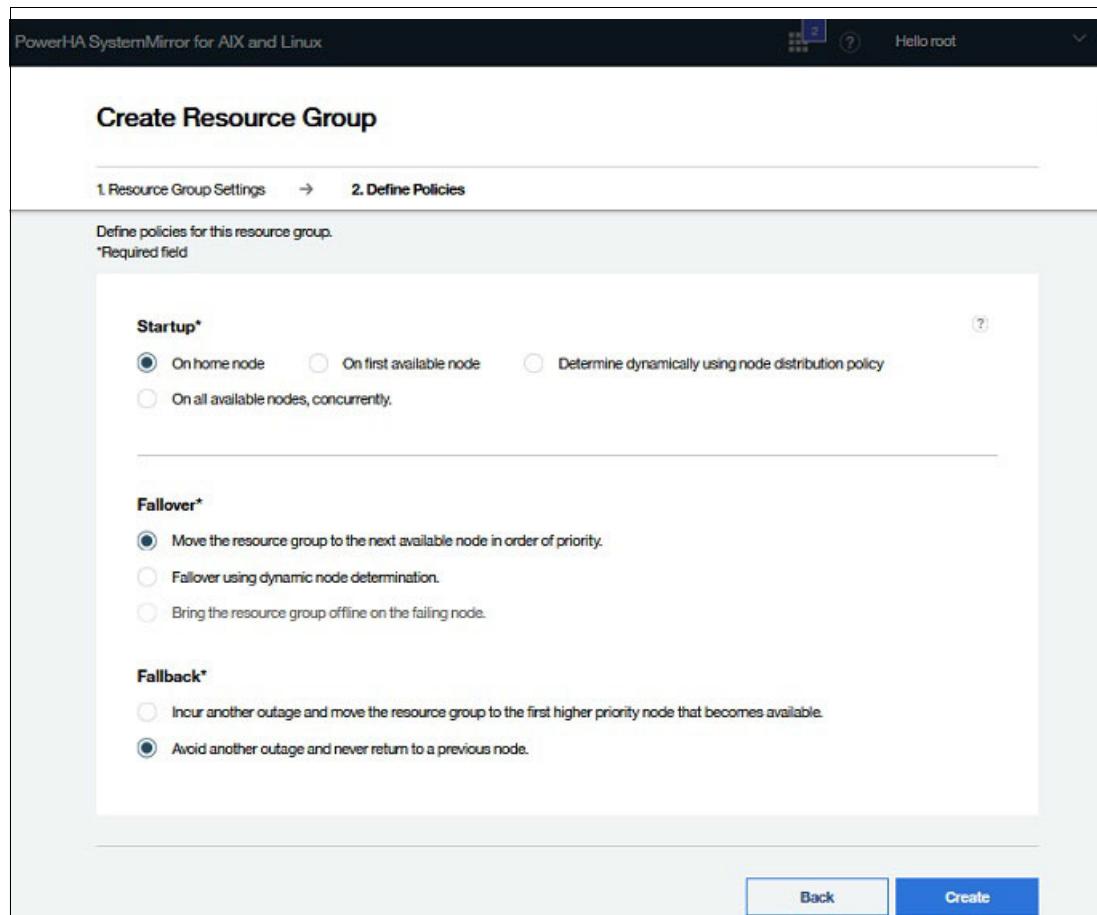


Figure 13-11 SMUI: Resource group behavior policies

- After clicking **Create**, the RG is created. The dialog box that is shown in Figure 13-12 opens.

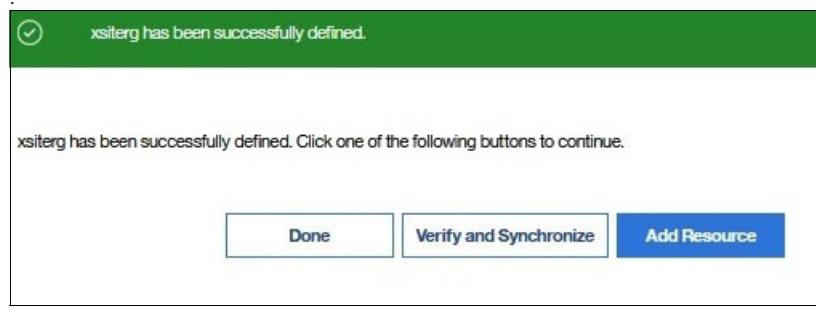


Figure 13-12 SMUI: Add Resource

In our case, we click **Add Resource** and continue with the procedure that is described in 13.3.3, “Defining resources” on page 539.

### 13.3.3 Defining resources

Complete the following steps:

- Figure 13-13 shows four basic resource options. We choose Service IP by clicking **Add Resource**.

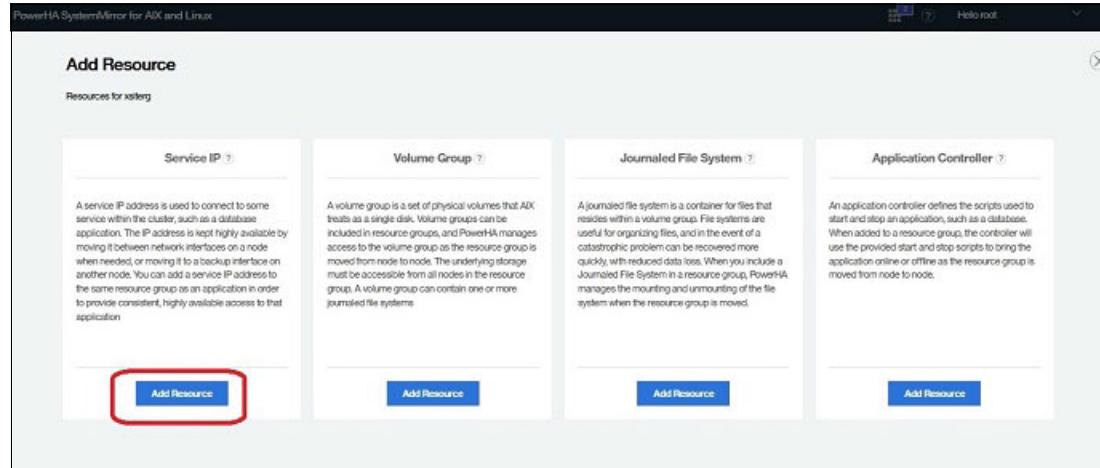


Figure 13-13 SMUI: Add Service IP

- Figure 13-14 shows a window where you can define an IP address. We select **Existing IP Address** because we already added it into the /etc/hosts file and selected **bdbsvc** from the drop-down menu. Otherwise, you can choose to select **Define IP Address** and manually enter it. In either case, make your choice and click **Continue**.

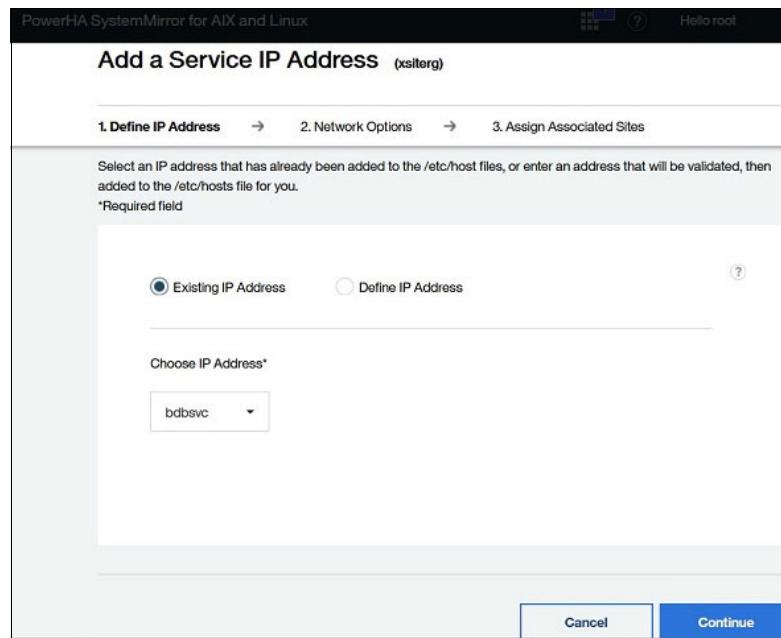


Figure 13-14 SMUI: Define an existing IP address

3. Figure 13-15 shows a window where you can choose a Network Name and Subnet mask. Make your choices, and then click **Continue**. The subnet mask is optional because it is inherited from the network.

The screenshot shows the 'Add a Service IP Address' window for PowerHA SystemMirror. The title bar says 'PowerHA SystemMirror for AIX and Linux'. The main heading is 'Add a Service IP Address (xsiterg)'. Below it, a progress bar shows '1. Define IP Address → 2. Network Options → 3. Assign Associated Sites'. A message box says 'Specify a valid network name \*Required field'. It contains two input fields: 'Network Name\*' with 'net\_ether\_01' selected, and 'Subnet mask' with 'Enter subnet mask'. At the bottom are 'Back' and 'Continue' buttons.

Figure 13-15 SMUI: Network and subnet mask for a service IP

4. Figure 13-16 shows the window where you specify whether the service IP should be associated with a specific site. In our scenario, we allow the same service IP to traverse both sites, so we select **No Site**, as shown in Figure 13-16. Click **Create** to complete.

The screenshot shows the 'Add a Service IP Address' window for PowerHA SystemMirror. The title bar says 'PowerHA SystemMirror for AIX and Linux'. The main heading is 'Add a Service IP Address (xsiterg)'. Below it, a progress bar shows '1. Define IP Address → 2. Network Options → 3. Assign Associated Sites'. A message box says 'Specify a valid network name \*Required field'. It contains a 'Select Site' section with three radio buttons: 'Saginaw', 'Haslet', and 'No Site'. The 'No Site' option is selected. At the bottom are 'Back' and 'Create' buttons.

Figure 13-16 SMUI: Choosing a site association for a service IP

### 13.3.4 Defining an application controller

Complete the following steps:

1. When you get the confirmation message from step 4 on page 540, you can continue to define more resources by clicking **Add Resource**, as shown in Figure 13-17.

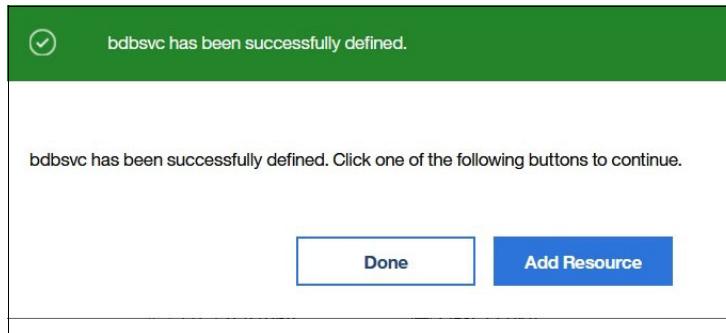


Figure 13-17 SMUI: Service IP created

2. The next window shows the resources that you can create, as shown in Figure 13-13 on page 539. Select Application Controller, and complete the following fields:

- Application controller name: dummyapp
- Specify start script: /usr/bin/banner start
- Specify stop script: /usr/bin/banner stop

There is the checkbox option to automatically copy the scripts to all nodes. We do not use it in our scenario, but it is a useful option.

- Startup mode: Background
- Monitor CPU usage: No

Because we entered No, the remaining two options do not apply, so we leave them blank.

After completing the fields, click **Continue**, as shown in Figure 13-18. You created the resource, which is automatically added into the RG.

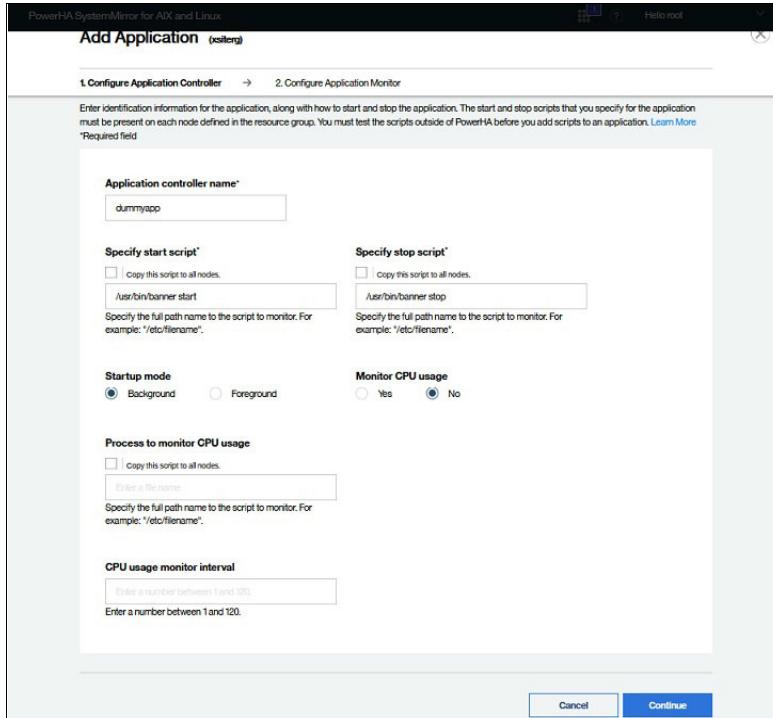


Figure 13-18 SMUI: Define application controller

The dialog box that is shown in Figure 13-19 opens.

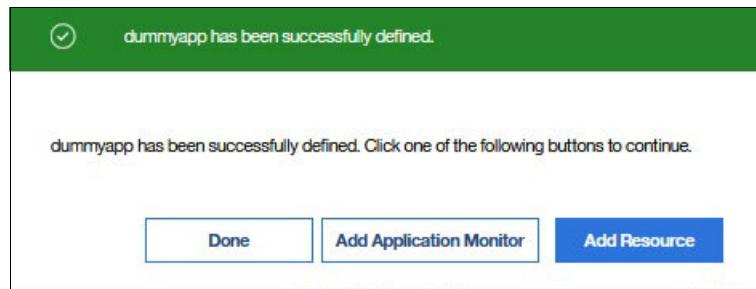


Figure 13-19 SMUI: Add resource

### 13.3.5 Defining and creating volume groups

Complete the following steps:

1. Select **Add Resource**, as shown in Figure 13-19. The window that is shown in Figure 13-13 on page 539 opens, which you can use to select what resources to add.
2. Choose the VG option. The window that is shown in Figure 13-20 on page 543 opens.

PowerHA SystemMirror for AIX and Linux

### Add Volume Group

1. Volume Group Settings → 2. Select physical volume → 3. Mirror pools

Enter a name for the volume group that you want to create and define the volume group settings for xsitevg  
\*Required field

Select volume group type\*

Scalable    Original    Big

Volume group name\*

xsitevg

Partition Size

256 MB

Maximum physical partitions \*

32 x 1024 = 32768

Maximum logical volumes \*

256

Cancel Continue

Figure 13-20 SMUI: Create volume group type and size

3. Complete the fields and click **Continue**. In our scenario, we completed them as follows:
  - Select volume group type: Scalable
  - Volume group name: xsitevg
  - Partition Size: 256MB
  - Maximum PPs: 32
  - Maximum logical volumes: 256
4. A window opens where you can choose the specific disks that you want to be members of the VG. In our scenario, we have only two available disks, hdisk4 and hdisk8, as shown in Figure 13-21. We chose them both.

PowerHA SystemMirror for AIX and Linux

### Add Volume Group

1. Volume Group Settings → 2. Select physical volume → 3. Mirror pools

Select disks that you want to assign to the volume group.  
\*Required field

Select physical volume\*

Search disk name X   hdisk 4 - 8

hdisk4    hdisk8

Back Continue

Figure 13-21 SMUI: Choose disks for the volume group

- Click **Continue** to open the **Mirror pools** tab, as shown in Figure 13-22.

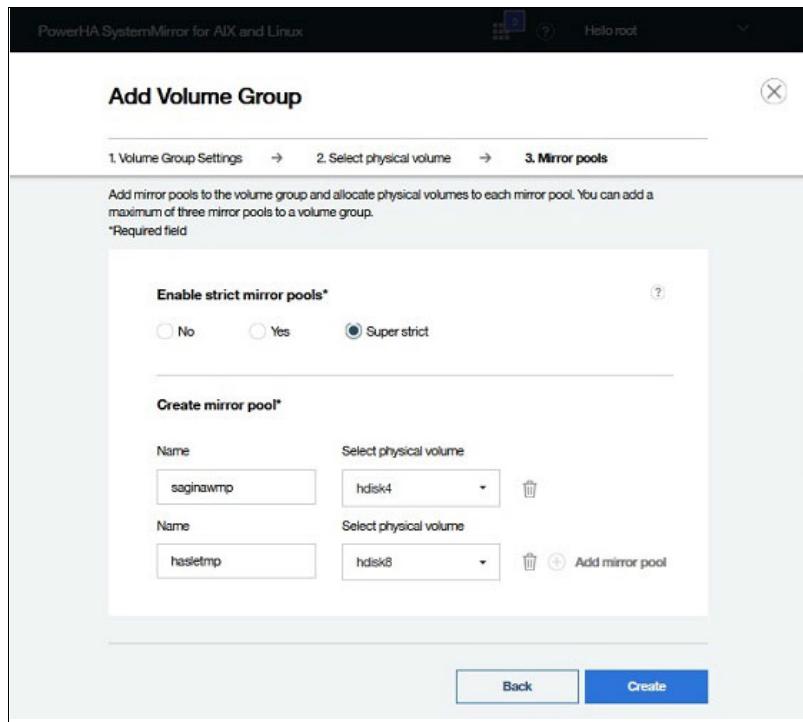


Figure 13-22 SMUI: Create and assign mirror pools

- We chose and created the options as follows:

- Enable strict mirror pools: supersstrict
- Create mirror pools
  - saginawmp hdisk4
  - hasletmp hdisk8

### 13.3.6 Creating mirrored logical volumes and file systems

Historically, and often today, admins manually create LVs first and then create the file system. The main reason is to specify the exact LV name and choose inline logs for the file system. If those two things are needed, use C-SPOC to accomplish these tasks, as described in 7.4, “Shared storage management” on page 275.

In our scenario, we use the SMUI to create the file system. We create a generically named LV and a jfslog1v. The SMUI also auto-detects the fact that the VG is already defined with mirror pools and will mirror the LV properly.

Complete the following steps:

- Click **Add Resource** from the successful VG creation status window. Select **Journaled File System** from the Add Resource window, which is similar to the one that is shown in Figure 13-13 on page 539. The Add Journaled File System window opens, as shown in Figure 13-23 on page 545.

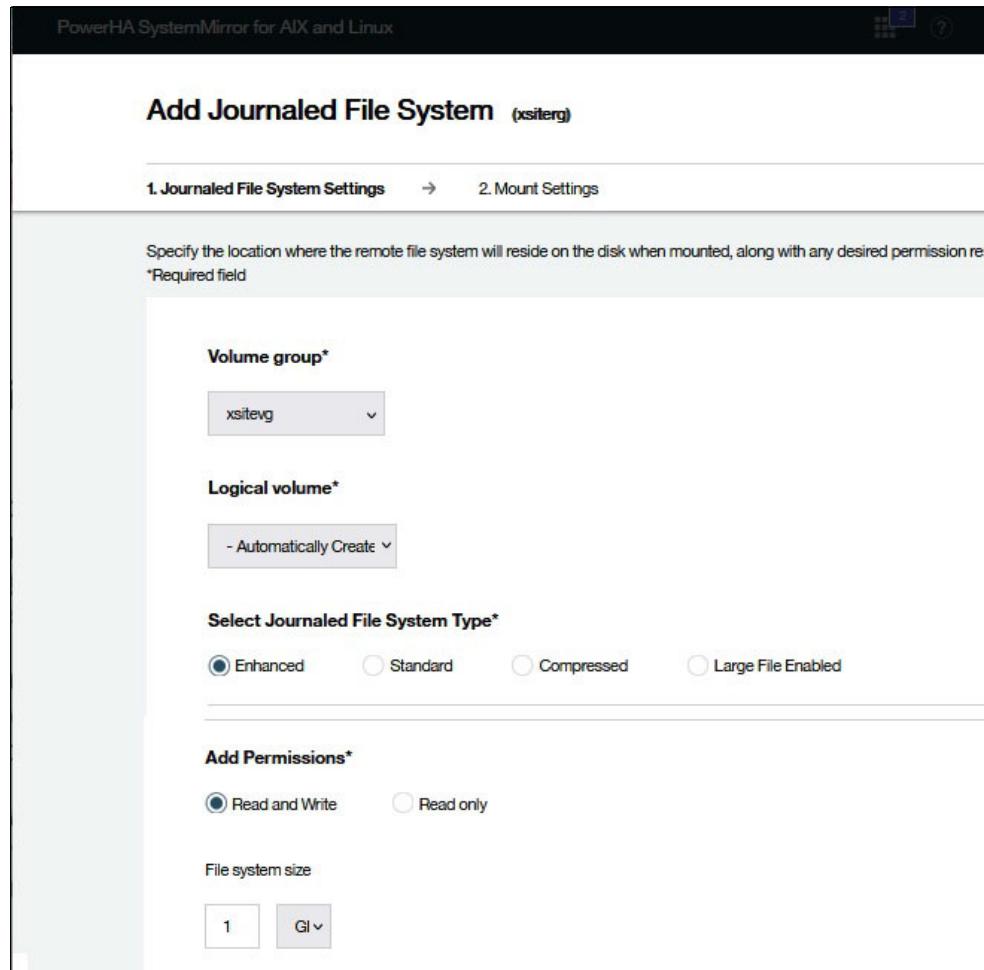


Figure 13-23 SMUI: Create Journaled File System 2

2. We chose the defaults except for changing permissions to Read and Write, and the size of 1 GB. If any existing LVs without a file system were detected, they would have been available for selection. In our case, we have none, so we allow SMUI to auto-create the LV for us.

Click **Continue**. You see the secondary window for Mount Options, as shown in Figure 13-24. In our example, we enter the mount point name `/smuijfstest` and click **Create**.

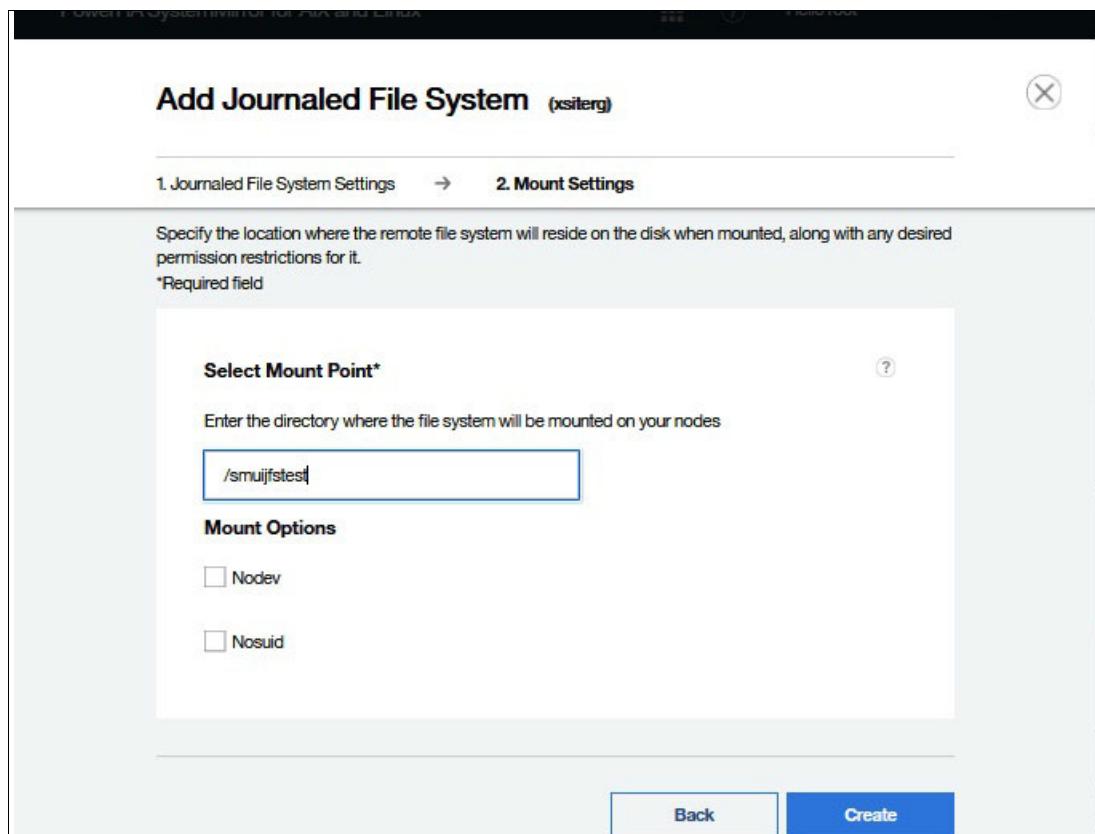


Figure 13-24 SMUI: JFS2 mount point and mount options

When the process completes, you can either click **Done** or **Add Resource**, as shown in Figure 13-25. In our case, we click **Done**. However, if more resources must be added, repeat the process by clicking **Add Resource**.

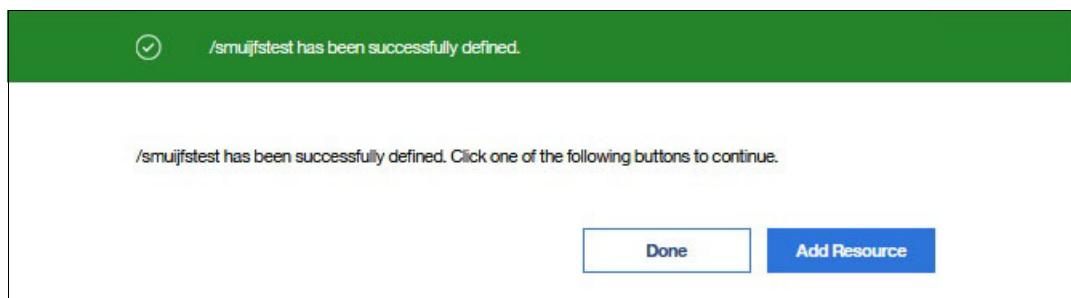


Figure 13-25 SMUI: JFS2 successful creation

### 13.3.7 Cluster configuration validation

The resources are automatically added to the RG after their creation, but that process should be validated. You should also verify that the LVM mirroring was created with superstrict mirror pools. Because these validations are not visible within the same SMUI window, we chose to use the CLI.

## Resource groups and resources

Example 13-1 shows the RG information from the **clshowres** command. We edited out the fields that do not apply. The option of Use forced varyon for volume groups, if necessary should be set to true.

*Example 13-1 Using the clshowres command to display a resource group configuration*

|                                                   |                              |
|---------------------------------------------------|------------------------------|
| Resource Group Name                               | xsiterg                      |
| Participating Node Names                          | shawn jordan jessica         |
| Startup Policy                                    | Online on Home Node Only     |
| Fallover Policy                                   | Fallover To Next Priority No |
| de In The List                                    |                              |
| Fallback Policy                                   | Never Fallback               |
| Site Relationship                                 | ignore                       |
| Node Priority                                     |                              |
| Service IP Label                                  | bdbsvc                       |
| Filesystems /smuijfstest                          |                              |
| Filesystems Consistency Check                     | fsck                         |
| Filesystems Recovery Method                       | sequential                   |
| Use forced varyon for volume groups, if necessary | false                        |
| Application Servers                               | dummyapp                     |
| Filesystems mounted before IP configured          | false                        |

Like most options, you can set the setting by using either SMIT or the CLI. In our case, we use the CLI and run the following command:

```
clmgr modify resource_group 'xsiterg' FORCED_VARYON='true'
```

## LVM mirror configuration

Now, you want to check that your LV is mirrored correctly, both by disk and mirror pools. In our case, it is a simple process because we have only two disks and two mirror pools, with one disk per mirror pool.

### Logical volume attributes

Check the partition mapping of the LV, as shown in Example 13-2. You can see that one entire copy is on each disk, which is what we want in our case.

*Example 13-2 Logical volume mirror mappings*

```
lslv -m lv01
lv01:/smuijfstest
LPAR PP1 PV1 PP2 PV2 PP3 PV3
0001 0077 hdisk4 0077 hdisk8
0002 0078 hdisk4 0078 hdisk8
0003 0079 hdisk4 0079 hdisk8
0004 0080 hdisk4 0080 hdisk8
0005 0081 hdisk4 0081 hdisk8
0006 0082 hdisk4 0082 hdisk8
0007 0083 hdisk4 0083 hdisk8
0008 0084 hdisk4 0084 hdisk8
```

Specifically check the LV attributes that are highlighted in bold in Example 13-3. The attributes correspond to the upper bound, strictness, and preferred read. In our case, we must restrict the upper bound and enable superstrictness, which is a best practice. Although superstrict for the mirror pools is already enabled, it should be the highest determining factor.

*Example 13-3 Logical volume attributes before*

---

```
ls1v 1v01
LOGICAL VOLUME: 1v01 VOLUME GROUP: xsitevg
LV IDENTIFIER: 00c472c00004b0000000184f02ea9cc.2 PERMISSION: read/writ
VG STATE: active/complete LV STATE: closed/syncd
TYPE: jfs2 WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
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: 1024
MOUNT POINT: /smuijfstest LABEL: /smuijfstest
DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes
Serialize IO ?: NO
INFINITE RETRY: no PREFERRED READ: 0
DEVICESUBTYPE: DS_LVZ
COPY 1 MIRROR POOL: saginawmp
COPY 2 MIRROR POOL: hasletmp
COPY 3 MIRROR POOL: None
ENCRYPTION: no
```

---

To change the LV attributes upper bound and strictness, use the C-SPOC CLI:

```
/usr/es/sbin/cluster/cspoc/cli_chlv -s s -u 1 1v01
```

This command should be run for every LV in the VG. In our case, we also have 1og1v01. We verify that the attributes are set as needed, as shown in Example 13-4.

*Example 13-4 Logical volume attributes after*

---

```
ls1v 1v01
LOGICAL VOLUME: 1v01 VOLUME GROUP: xsitevg
LV IDENTIFIER: 00c472c00004b0000000184f02ea9cc.2 PERMISSION: read/write
VG STATE: active/complete LV STATE: closed/syncd
TYPE: jfs2 WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
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: 1
MOUNT POINT: /smuijfstest LABEL: /smuijfstest
DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?: NO
```

---

---

|                     |           |                   |
|---------------------|-----------|-------------------|
| INFINITE RETRY:     | no        | PREFERRED READ: 0 |
| DEVICE SUBTYPE:     | DS_LVZ    |                   |
| COPY 1 MIRROR POOL: | saginawmp |                   |
| COPY 2 MIRROR POOL: | hasletmp  |                   |
| COPY 3 MIRROR POOL: | None      |                   |
| ENCRYPTION:         | no        |                   |

---

PREFERRED READ has not changed. Technically, it is not required to use this option, but for the best performance, it is a best practice to always read from the disk locally on each site. This action can be intelligently controlled by PowerHA by using a combination of attributes at both the VG and the mirror pool levels, by using LVM –Preferred Read and Storage Location.

### **Volume group attributes**

In all clusters, the shared VG attribute of **AUTO ON** should be disabled (No). For LVM mirrored configurations, disable **QUORUM**. Check or assign the LVM Preferred Read option by starting C-SPOC and completing the following steps:

1. Run **smitty cl\_vgsc**, and then select Change>Show characteristics of a Volume Group.
2. Select the xsitevg VG from the list.
3. Check and set the settings for the attributes as follows, and the ones that are highlighted in bold in Example 13-5:
  - Activate volume group AUTOMATICALLY at system restart?: no
  - A QUORUM of disks required to keep the volume group online?: no
  - Mirror Pool Strictness: Superstrict
  - LVM Preferred Read: siteaffinity

The only attribute that might require further explanation is LVM Preferred Read. It has the following options (press F4 to get the list):

|              |                                                                                                                                                            |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| roundrobin   | The default policy for LVM Preferred Read copy. LVM decides which mirror copy should be used to read the data.                                             |
| favorcopy    | Choose if you would like to read from Flash Storage regardless of where the RG is online.                                                                  |
| siteaffinity | Choose if you would like to read from storage at the local site where the RG is online. The siteaffinity option is available only for site-based clusters. |

#### *Example 13-5 Volume group attributes*

---

##### Change>Show characteristics of a Volume Group

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

|                                                                |                      |   |
|----------------------------------------------------------------|----------------------|---|
| * VOLUME GROUP name                                            | xsitevg              |   |
| Resource Group Name                                            | xsiterg              |   |
| Node Names                                                     | shawn,jordan,jessica |   |
| * Activate volume group AUTOMATICALLY at system restart?       | no                   | + |
| * A QUORUM of disks required to keep the volume group online ? | no                   | + |
| Change to big VG format?                                       | no                   | + |
| Change to scalable VG format?                                  | no                   | + |

---

|                                 |                     |   |
|---------------------------------|---------------------|---|
| LTG Size in kbytes              | 512                 | + |
| Set hotspare characteristics    | n                   | + |
| Max PPs per VG in units of 1024 | 32                  | + |
| Max Logical Volumes             | 256                 | + |
| <b>Mirror Pool Strictness</b>   | <b>Superstrict</b>  | + |
| <b>LVM Preferred Read</b>       | <b>siteaffinity</b> | + |
| Enable LVM Encryption           | no                  | + |

---

As is often the case after any change, the cluster must be synchronized. Synchronization is often done after every change or group of changes. The benefit of doing so after every change is that it makes it simpler to troubleshoot any problems that are encountered.

For example, after setting siteaffinity and synchronizing the cluster, you see the warning that is shown in Example 13-6. It is a reminder that you must assign the mirror pools to the sites.

---

*Example 13-6 Verification warning about mirror pools*

WARNING: LVM Preferred Read for volume group xsitevg is set to siteaffinity, but the associated storage location, Saginaw, is not configured for any mirror pool copy.

Hence, the LVM Preferred Read setting is overridden as roundrobin so that AIX will decide which copy needs to be used while reading the data.

To check this cluster setting from the CLI, run the command that is shown in Example 13-7.

---

*Example 13-7 Checking the LVM\_PREFERRRED\_READ attribute*

---

```
clmgr query cluster|grep -i lvm
LVM_PREFERRRED_READ="siteaffinity"
```

---

### **Mirror pool attributes**

With PowerHA, you can assign each mirror pool to a specific site. If you combine this function with LVM Preferred Read, PowerHA automatically can specify the proper disk to read from each site. It also helps to ensure that VG mirroring is configured and maintained properly, which greatly minimizing the chance that the mirroring goes askew.

In our case, hdisk4 is local to the saginawmp mirror pool, and hdisk8 is local to hasletmp, as shown in Example 13-8.

---

*Example 13-8 Display mirror pools*

---

```
lsmp -A xsitevg
VOLUME GROUP: xsitevg Mirror Pool Super Strict: yes

MIRROR POOL: saginawmp Mirroring Mode: SYNC
MIRROR POOL: hasletmp Mirroring Mode: SYNC

lspv -P
Physical Volume Volume Group Mirror Pool
hdisk2 rootvg
hdisk3 caavg_private
hdisk4 xsitevg saginawmp
hdisk8 xsitevg hasletmp
```

---

To assign each mirror pool to their respective site, use C-SPOC to complete the following steps:

1. Run `smitty cl_mirrorpool_mgt`, and then select Change>Show Characteristics of a Mirror Pool.
2. Choose the mirror pool from the list, which in our case is saginawmp.
3. Select Storage Location, press F4, and select the site that you want. In our case, it is Saginaw.
4. Press Enter twice to run the process.
5. Repeat as needed for each mirror pool. In our case, we repeated the process with hasletmp and assigned it to site Haslet.

When all creations and changes complete, the cluster still needs to be synchronized to push those changes across all cluster nodes.

## Synchronizing a cluster by using the SMUI

A cluster can be synchronized by using SMIT or the `clmgr` CLI. In this example, we synchronize the cluster by using the SMUI.

In the left navigation pane, expand the cluster to show all the sites and nodes. You see that node shawn has a yellow warning indicator, and that there are unsynchronized changes. The cluster changes were made on that specific node, but you must synchronize from and on that node. To do so, click the three dots next to node shawn and select **Synchronize Cluster**, as shown in Figure 13-26.

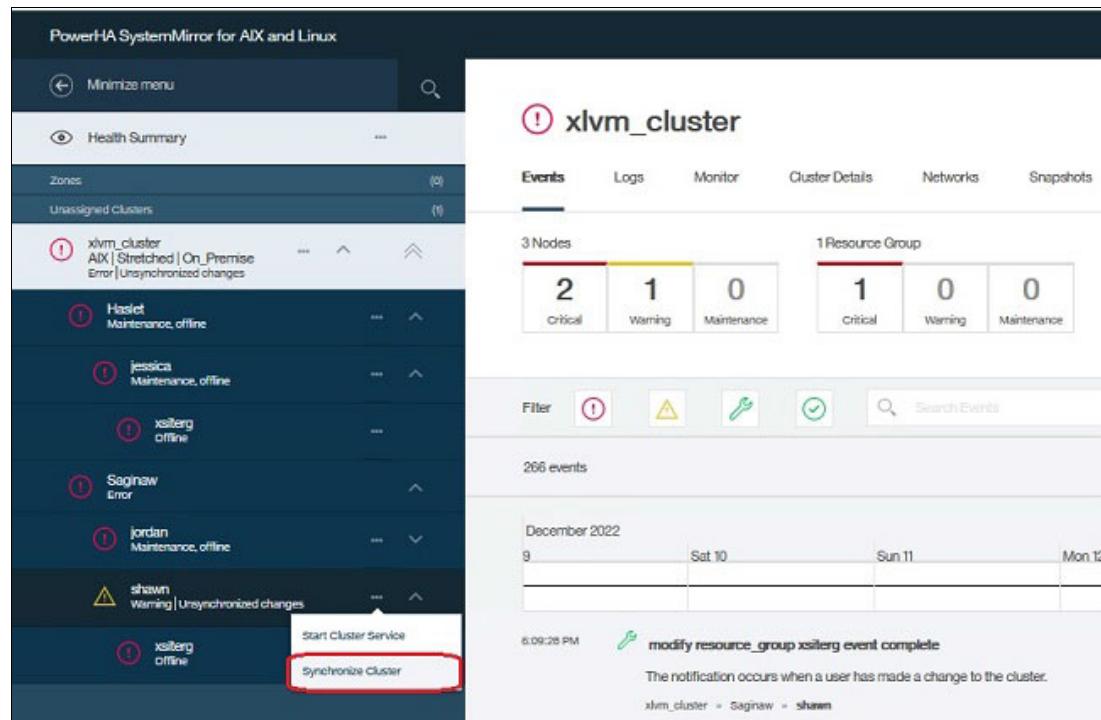


Figure 13-26 SMUI: Choose the node to synchronize the cluster

When the synchronization completes, you receive a confirmation message that the synchronization completed successfully, and the yellow warning indicator has been removed. All cluster items show red, which is normal because the entire cluster is offline, as shown in Figure 13-27.

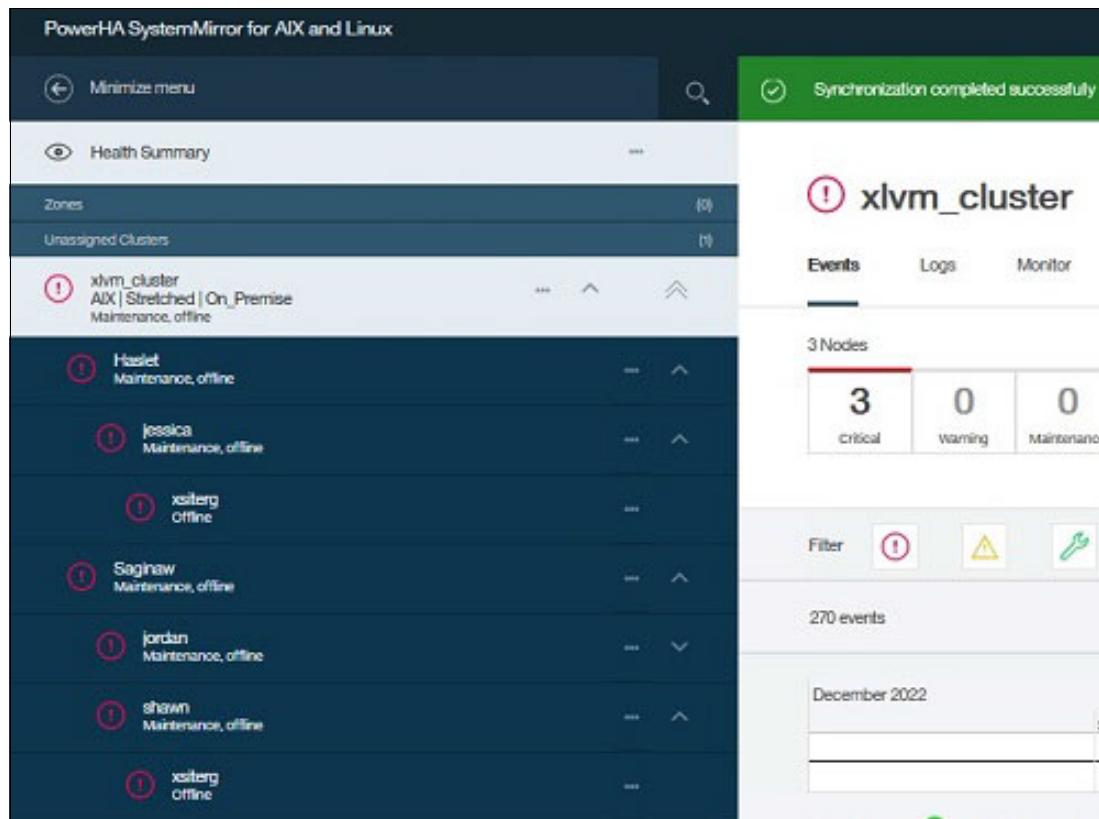


Figure 13-27 SMUI: Sync status completed

## 13.4 Testing

This section describes some of the common major failures, such as:

- ▶ Local node failure within the primary site
- ▶ Rolling node failures that are promoted to a site failure
- ▶ Primary site local storage failure
- ▶ Primary site remote storage failure
- ▶ Primary site all storage failure

The first two test scenarios are not any different than in a typical, local shared cluster. However, it is important to show the most common likely failures that might be encountered. Also, unless otherwise stated, every test begins with all the nodes active in the cluster, as shown in Figure 13-28 on page 553. We also validate before and after failover that PREFERRED READ is set to the correct value on each site.

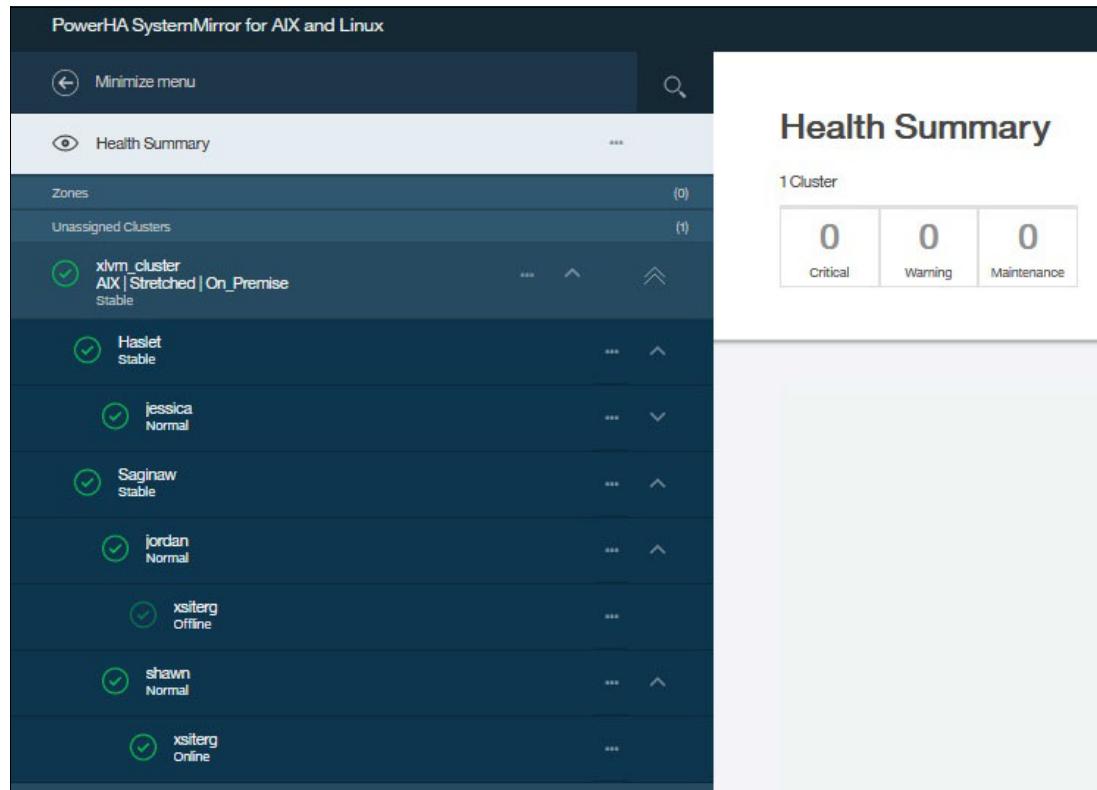


Figure 13-28 SMUI: Cluster active

### 13.4.1 Local node failure within the primary site

In this scenario, the RG is online on its home node shawn, as shown in Figure 13-28. In our case, we verify that PREFERRED READ is set to 1, which corresponds to the saginawmp mirror pool, as shown in Example 13-9.

*Example 13-9 PREFERRED READ is the primary mirror pool*

---

```
lslv lv01
LOGICAL VOLUME: lv01 VOLUME GROUP: xsitevg
LV IDENTIFIER: 00c472c000004b0000000184f02ea9cc.2 PERMISSION: read/write
VG STATE: active/complete LV STATE: closed/syncd
TYPE: jfs2 WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
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: 1
MOUNT POINT: /smuijfstest LABEL: /smuijfstest
DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?: NO
INFINITE RETRY: no PREFERRED READ: 1
DEVICESUBTYPE: DS_LVZ
COPY 1 MIRROR POOL: saginawmp
```

```
COPY 2 MIRROR POOL: hasletmp
COPY 3 MIRROR POOL: None
ENCRYPTION: no
```

In our case, we fail the node by running `reboot -q`. PowerHA detects the node failure and acquires the RG on the next available node within the same site, jordan, as shown in Figure 13-29. You can see that the RG is online, the primary node state is unknown, and the primary site has an error because of the previously created failure.

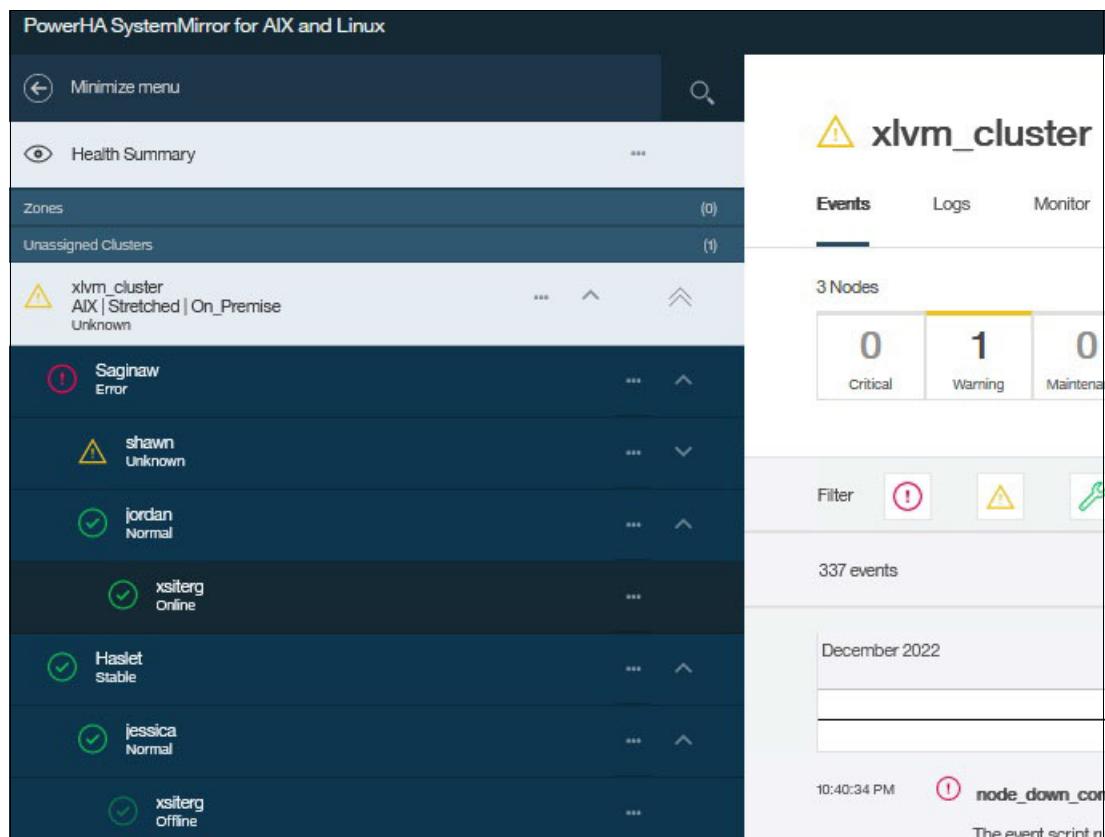


Figure 13-29 Resource group acquired by secondary node jordan

We check the LV settings again, and they are the same as shown in Example 13-9 on page 553.

### 13.4.2 Rolling node failures that are promoted to a site failure

In this scenario, we continue from 13.4.1, “Local node failure within the primary site” on page 553. The primary node shawn is not active in the cluster and the RG is online on node jordan, as shown in Figure 13-29. We now fail the secondary node jordan by running `reboot -q`. PowerHA detects that the node and the site are down, and that the RG is acquired at the secondary site on node jessica, as shown in Figure 13-30 on page 555.

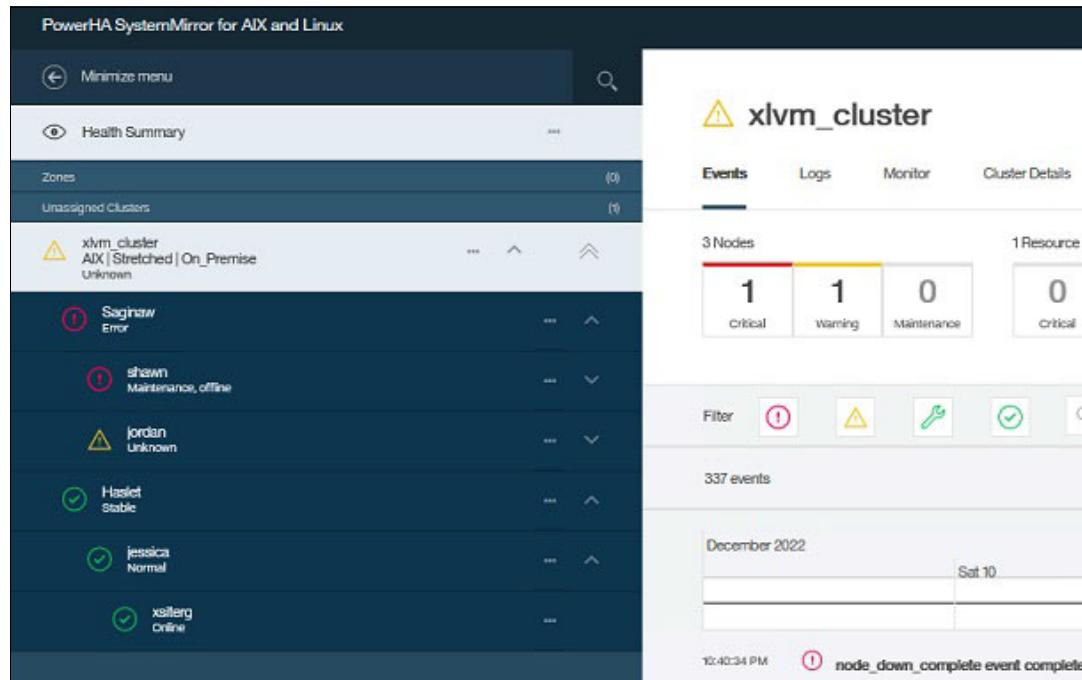


Figure 13-30 Resource group that is acquired by the secondary site node jessica

When we check the LV, we see that PREFERRED READ changed to 2, as shown in Example 13-10, which is what we expected because the LV is now the haslettmp mirror pool.

---

*Example 13-10 PREFERRED READ is the secondary site*

---

```
ls1v lv01
LOGICAL VOLUME: lv01 VOLUME GROUP: xsitevg
LV IDENTIFIER: 00c472c000004b0000000184f02ea9cc.2 PERMISSION: read/write
VG STATE: active/complete LV STATE: closed/syncd
TYPE: jfs2 WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
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: 1
MOUNT POINT: /smuijfstest LABEL: /smuijfstest
DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/ACTIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?: NO
INFINITE RETRY: no PREFERRED READ: 2
DEVICESUBTYPE: DS_LVZ
COPY 1 MIRROR POOL: saginawmp
COPY 2 MIRROR POOL: haslettmp
COPY 3 MIRROR POOL: None
ENCRYPTION: no
```

---

### 13.4.3 Primary site local storage failure

For our scenario, we start with a fully active cluster, as shown in Figure 13-28 on page 553.

In our scenario, we simulate the loss of access to the primary storage at the primary site. This simulation can be done in many ways, but we chose to unmap the volume hdisk4 at the primary site.

The disk loss is detected by AIX and reported in the error report, as shown in Example 13-11. There is no failover because the remote copy is still available, so there is continuous operation.

*Example 13-11 Error reporting failed disk at the primary site*

---

| #errpt | IDENTIFIER | TIMESTAMP  | T C | RESOURCE_NAME | DESCRIPTION                              |
|--------|------------|------------|-----|---------------|------------------------------------------|
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | EAA3D429   | 1212232322 | U S | LVDD          | PHYSICAL PARTITION MARKED STALE          |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | 52715FA5   | 1212232322 | U H | LVDD          | FAILED TO WRITE VOLUME GROUP STATUS AREA |
|        | E86653C3   | 1212232322 | P H | LVDD          | I/O ERROR DETECTED BY LVM                |
|        | F7DDA124   | 1212232322 | U H | LVDD          | PHYSICAL VOLUME DECLARED MISSING         |
|        | 52715FA5   | 1212232322 | U H | LVDD          | FAILED TO WRITE VOLUME GROUP STATUS AREA |
|        | E86653C3   | 1212232322 | P H | LVDD          | I/O ERROR DETECTED BY LVM                |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |
|        | EAA3D429   | 1212232322 | U S | LVDD          | PHYSICAL PARTITION MARKED STALE          |
|        | E86653C3   | 1212232322 | P H | LVDD          | I/O ERROR DETECTED BY LVM                |
|        | B6267342   | 1212232322 | P H | hdisk4        | DISK OPERATION ERROR                     |

---

The LV reports that it is stale and that the PV is missing, as shown in Example 13-12.

*Example 13-12 Mirrored logical volume is stale and the primary PV is missing*

---

| # lsvg -l xsitevg | xsitevg: | LV NAME | TYPE      | LPs      | PPs                | PVs          | LV STATE   | MOUNT POINT  |
|-------------------|----------|---------|-----------|----------|--------------------|--------------|------------|--------------|
|                   |          | loglv01 | jfs2log   | 1        | 2                  | 2            | open/stale | N/A          |
|                   |          | 1v01    | jfs2      | 128      | 256                | 2            | open/stale | /smuijfstest |
| # lsvg -p xsitevg | xsitevg: |         |           |          |                    |              |            |              |
| PV_NAME           | PV STATE |         | TOTAL PPs | FREE PPs | FREE               | DISTRIBUTION |            |              |
| hdisk4            | missing  |         | 374       | 245      | 75..00..20..75..75 |              |            |              |
| hdisk8            | active   |         | 374       | 245      | 75..00..20..75..75 |              |            |              |

---

To recover, remap the volume back to the primary node shown and run **varyonvg -c xsitevg**. The disk becomes active and auto syncs the stale mirrored copies.

### 13.4.4 Primary site remote storage failure

This scenario is almost identical to the scenario that is described in 13.4.4, “Primary site remote storage failure” on page 557. The only difference is that the disk that is at the secondary site, hdisk8, is the one that is unmapped from the storage host definition. The net effect is the same: The disk loss is detected by AIX and reported in the error report, as shown in Example 13-13.

*Example 13-13 Error reporting a failed disk at the primary site*

---

| #errpt   | IDENTIFIER | TIMESTAMP | T C      | RESOURCE_NAME | DESCRIPTION                              |
|----------|------------|-----------|----------|---------------|------------------------------------------|
| EAA3D429 | 1213003022 | U S       | LVDD     |               | PHYSICAL PARTITION MARKED STALE          |
| F7DDA124 | 1213003022 | U H       | LVDD     |               | PHYSICAL VOLUME DECLARED MISSING         |
| 52715FA5 | 1213003022 | U H       | LVDD     |               | FAILED TO WRITE VOLUME GROUP STATUS AREA |
| E86653C3 | 1213003022 | P H       | LVDD     |               | I/O ERROR DETECTED BY LVM                |
| B6267342 | 1213003022 | P H       | hdisk8   |               | DISK OPERATION ERROR                     |
| EAA3D429 | 1213003022 | U S       | LVDD     |               | PHYSICAL PARTITION MARKED STALE          |
| AA8AB241 | 1213003022 | T O       | clevmgrd |               | OPERATOR NOTIFICATION                    |
| AA8AB241 | 1213003022 | T O       | clevmgrd |               | OPERATOR NOTIFICATION                    |
| E86653C3 | 1213003022 | P H       | LVDD     |               | I/O ERROR DETECTED BY LVM                |
| B6267342 | 1213003022 | P H       | hdisk8   |               | DISK OPERATION ERROR                     |
| DE3B8540 | 1213003022 | P H       | hdisk8   |               | PATH HAS FAILED                          |
| DE3B8540 | 1213003022 | P H       | hdisk8   |               | PATH HAS FAILED                          |

---

AIX also marks the disk as missing and logs the LV as stale, as shown in Example 13-14.

*Example 13-14 Mirrored logical volume is stale and the secondary PV is missing*

---

| # lsvg -l xsitevg                                                               |
|---------------------------------------------------------------------------------|
| xsitevg:                                                                        |
| LV NAME           TYPE       LPs      PPs      PVs    LV STATE      MOUNT POINT |
| log1v01          jfs2log    1        2        2     open/stale    N/A           |
| lv01             jfs2       128      256      2     open/stale    /smuijfstest  |
| # lsvg -p xsitevg                                                               |
| xsitevg:                                                                        |
| PV_NAME          PV STATE           TOTAL PPs    FREE PPs    FREE DISTRIBUTION  |
| hdisk4           active           374        245       75..00..20..75..75       |
| hdisk8           missing         374        245       75..00..20..75..75        |

---

To recover, remap the volume back to the primary node shawn and run **varyonvg -c xsitevg**. The disk becomes active and auto syncs the stale mirrored copies.

### 13.4.5 Primary site all storage failure

In this scenario, we unmap hdisk4 and hdisk8 from nodes shawn and jordan at the primary site Saginaw. This action simulates an entire SAN failure at the primary site, and initiates a failover due to the loss of a quorum of a VG, as shown in Example 13-15.

*Example 13-15 Primary node loses access to all shared disks*

|          |                       |                                          |
|----------|-----------------------|------------------------------------------|
| CAD234BE | 1213004722 U H LVDD   | QUORUM LOST, VOLUME GROUP CLOSING        |
| CAD234BE | 1213004722 U H LVDD   | QUORUM LOST, VOLUME GROUP CLOSING        |
| F7DDA124 | 1213004722 U H LVDD   | PHYSICAL VOLUME DECLARED MISSING         |
| 52715FA5 | 1213004722 U H LVDD   | FAILED TO WRITE VOLUME GROUP STATUS AREA |
| E86653C3 | 1213004722 P H LVDD   | I/O ERROR DETECTED BY LVM                |
| B6267342 | 1213004722 P H hdisk8 | DISK OPERATION ERROR                     |

The secondary node jordan attempts to acquire the RG. The node starts the service IP, but because it also lost access to storage, it fails to activate the VG. This failure forces the RG to error out and then fall over to the secondary site Haslet onto node jessica. The failover succeeds, as shown in Figure 13-31. We verify the correct PREFERRED READ setting, and it is set as shown in Example 13-10 on page 555.

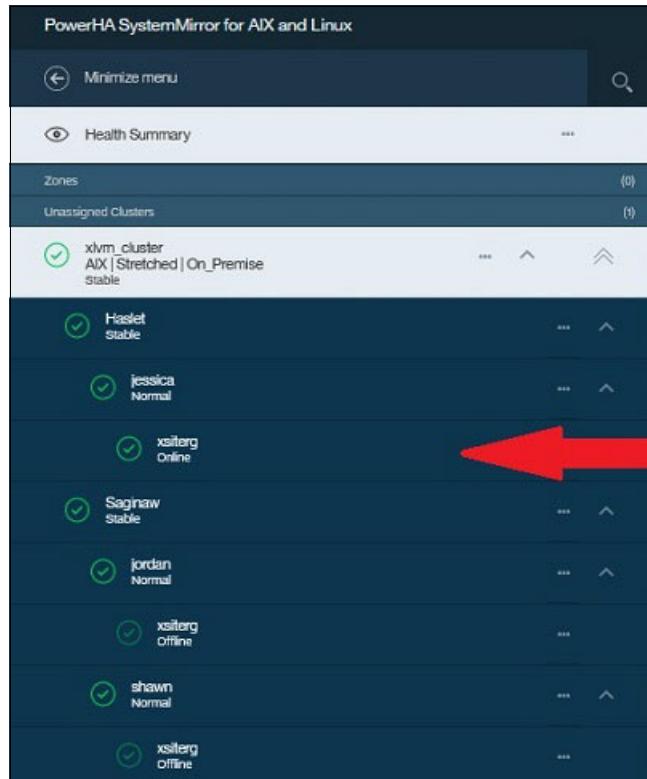


Figure 13-31 SMUI: RG on a secondary site

However, the overall cluster status looks normal because all nodes are still active in the cluster. There is clearly a SAN problem that must be resolved, but the cluster status might not show that. You most likely must sift through the error report and the cluster logs to determine the problem.

To recover, stop cluster services on both nodes at the primary site. Remap both volumes back to both nodes through the storage subsystem. When the process completes, verify that you can query the disks, as shown in Example 13-16. We ran this test multiple times, and on one occasion we could *not* query the disks. So, we ran `cfgmgr`, and then the query was successful.

### *Example 13-16 Disk access validation*

```
lquerypv -h /dev/hdisk4 80 10
00000080 00C472C0 F0139AA8 00000000 00000000 |...r.....|

lqueryvg -Ptp hdisk4
Physical: 00c472c0f0139aa8 2 0
 00c472c0chaee06e 1 0
```

Restart cluster services on both nodes. When the nodes join and stabilize, move the RG back to the primary node and site by running the `c1mr` command, as shown in Example 13-17.

*Example 13-17 Moving the resource group back to the primary site*

```
clmgr move rg xsiterg node=shawn
Attempting to move resource group xsiterg to node shawn
```

Waiting for the cluster to stabilize.....

Resource group movement is successful.  
Resource group xsiterg is online on node shawn.

Cluster Name: x1vm cluster

Resource Group Name: xsiterg

| Node           | Primary State | Secondary State  |
|----------------|---------------|------------------|
| shawn@Saginaw  | ONLINE        | OFFLINE          |
| jordan@Saginaw | OFFLINE       | OFFLINE          |
| jessica@Haslet | OFFLINE       | ONLINE SECONDARY |

#### **13.4.6 Primary site failure**

In this scenario, we simulate an entire site failure by halting both primary site nodes and unmapping the primary disk hdisk4 from the secondary site's node jessica. The result is similar to the scenario that is described in 13.4.5, "Primary site all storage failure" on page 558, that is, a site fallover occurs to jessica, but this time there is a missing disk.

If you look at the `hacmp.out` log file, you see that the disk is detected as missing, and as a result, the mirrored copies *cannot* be synchronized, as shown in Example 13-18.

*Example 13-18 The hacmp.out file*

```
xsiterg:clvaryonvg(88.404):xsitevg[1396] varyonvg -n -c -A -O xsitevg
+xsiterg:clvaryonvg(88.406):xsitevg[1396] 2>& 1
+xsiterg:clvaryonvg(151.375):xsitevg[1396] varyonvg_output=$'PV Status: \thdisk
4\t00c472c0f0139aa8\tPVMISSING\n \thdisk8\t00c472c0cbaee06e\tPVACTIVE
\nvaryonvg: Volume group xsitevg is varied on.'
+xsiterg:clvaryonvg(151.375):xsitevg[1397] varyonvg_rc=0
+xsiterg:clvaryonvg(151.375):xsitevg[1397] typeset -li varyonvg_rc
```

```

+xsitevg:clvaryonvg(151.375):xsitevg[1399] ((0 != 0))
+xsitevg:clvaryonvg(151.375):xsitevg[1576] : At this point, xsitevg should be
varied on
.....
lv01 did not complete successfully' lv01

Dec 14 2022 20:19:35 !!!!!!! ERROR !!!!!!!

Dec 14 2022 20:19:35 syncvg for lv01 did not complete successfully
<LAT>|2022-12-14T20:19:35|ERROR|syncvg for lv01 did not complete successfully|</

```

---

The missing disk, stale LVs, and stale partitions are shown in Example 13-19.

*Example 13-19 Volume group state after a site failure*

---

```

lsvg -p xsitevg
xsitevg:
PV_NAME PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
hdisk4 missing 374 245 75..00..20..75..75
hdisk8 active 374 245 75..00..20..75..75

lsvg -l xsitevg
xsitevg:
LV NAME TYPE LPs PPs PVs LV STATE MOUNT POINT
loglv01 jfs2log 1 2 2 open/stale N/A
lv01 jfs2 128 256 2 open/stale /smuijfstest

#lsvg xsitevg|grep -i stale
TOTAL PVs: 2 VG DESCRIPTORS: 3
STALE PVs: 0 STALE PPs: 3

```

---

In this scenario, recovering the primary node and site requires discovering the reason for the original site failure and deciding whether or when to move the primaries back. It is possible to move the node back with one copy, assuming that the primary storage is still unavailable. However, if the primary storage is available, synchronize the copies before moving the node back to the primary site.

When the primary storage is available again, validate that the missing disk is now accessible on the secondary site node jessica, as shown in Example 13-16 on page 559. If not, run **cfgmgr** and try again. When querying the disk is successful, reactivate the VG to change hdisk4 from missing to active by running **varyonvg -c xsitevg**. This command also auto-synchronizes the stale partitions. When the nodes are back in sync, as shown in Example 13-20, the cluster nodes at the primary site can be restarted. Lastly, the RG can be moved back to the primary, as shown in Example 13-17 on page 559.

*Example 13-20 Volume group state after a site failure*

---

```

lsvg -p xsitevg
xsitevg:
PV_NAME PV STATE TOTAL PPs FREE PPs FREE DISTRIBUTION
hdisk4 active 374 245 75..00..20..75..75
hdisk8 active 374 245 75..00..20..75..75

lsvg -l xsitevg
xsitevg:
LV NAME TYPE LPs PPs PVs LV STATE MOUNT POINT

```

---

|                             |         |     |     |   |                   |              |
|-----------------------------|---------|-----|-----|---|-------------------|--------------|
| loglv01                     | jfs2log | 1   | 2   | 2 | open/syncd        | N/A          |
| lv01                        | jfs2    | 128 | 256 | 2 | open/syncd        | /smuijfstest |
| #lsvg xsitevg grep -i stale |         |     |     |   |                   |              |
| TOTAL PVs:                  | 2       |     |     |   | VG DESCRIPTORS: 3 |              |
| STALE PVs:                  | 0       |     |     |   | STALE PPs: 0      |              |

---





# IBM PowerHA and IBM Power Virtual Server

This chapter contains the following topics:

- ▶ What is IBM Power Virtual Server
- ▶ IBM Power and IBM Power Virtual Server HADR options
- ▶ Cloud HADR for IBM Power Virtual Server
- ▶ Disaster recovery replication methods for cloud
- ▶ Geographic Logical Volume Manager concepts
- ▶ Block-storage based replication
- ▶ File-storage based replication
- ▶ Hybrid and multi-cloud deployment models
- ▶ Hybrid cloud
- ▶ Multiple public clouds
- ▶ Cold disaster recovery

## 14.1 Introduction

The section provides an overview of an IBM Power Virtual Server and the high availability and disaster recovery (HADR) options that are available for cloud and hybrid cloud combinations.

### 14.1.1 What is IBM Power Virtual Server

IBM Power Virtual Server is an IBM infrastructure as a service (IaaS) offering that enables IBM Power customers to extend their on-premises environments to the cloud. IBM Power Virtual Server servers are in IBM data centers that are distinct from the IBM Cloud servers with separate networks and direct-attached storage. The environment is in its own pod and the internal networks are fenced, but offer connectivity options to meet customer requirements. This infrastructure design enables IBM Power Virtual Server to maintain key enterprise software certification and support because the IBM Power Virtual Server architecture is identical to certified on-premises infrastructure. The virtual servers, also known as logical partition (LPAR)s, run on IBM Power hardware with the PowerVM hypervisor.

With the IBM Power Virtual Server, you can quickly create and deploy one or more virtual servers (that are running either the AIX, IBM i, or Linux operating systems). After you provision the IBM Power Virtual Server, you get access to infrastructure and physical computing resources without the need to manage or operate them. However, you must manage the operating system and the software applications and data.

### 14.1.2 IBM Power and IBM Power Virtual Server HADR options

IBM Power servers are one of the most reliable platforms in the industry, often approaching levels similar to what you experience with IBM Z. IBM Power servers are designed to match the requirements of the most critical data-intensive workloads. For the 13 years, IBM Power servers achieved the highest server reliability rankings in the ITIC 2021 Global Server Hardware and Server OS Reliability survey.<sup>1</sup>

To provide even higher levels of availability, IBM Power Systems and IBM Power Virtual Server provide a range of HADR solutions. There are three broad categories of HADR solutions to consider:

- ▶ Active-inactive solutions: virtual machine recovery options
- ▶ Active-passive solutions: clustering options
- ▶ Active-active solutions: advanced clustering

#### Active-inactive solutions: virtual machine recovery options

There are a range of virtual machine (VM) recovery solutions. Broadly speaking, VM recovery solutions are designed to move an entire VM from one physical server to another one. In Live Partition Mobility (LPM), a running LPAR or VM is dynamically moved to another server without interrupting service to the users so that a set of LPARs can be moved for a firmware or hardware maintenance event, which eliminates the need for a planned outage.

If the VM fails, it can be restarted on another server in the cluster by using Simplified Remote Restart (SRR) capabilities. The resulting outage time can be shortened because the LPAR can be quickly restarted on another server, which eliminates problem determination and repair time on the failed server.

---

<sup>1</sup> <https://www.ibm.com/downloads/cas/A856LOWK>

For a fully automated remote restart function, IBM provides IBM VM Recovery Manager (VMRM). IBM Power Virtual Server uses this function so that your Power Virtual Server virtual instance may be restarted on another IBM Power Virtual Server server.

With the addition of replicated storage, IBM VMRM can provide disaster recovery (DR) operations so that your workloads can quickly restart at a remote data center if your primary data center experiences a failure. For more information, see *Implementing IBM VM Recovery Manager for IBM Power Systems*, SG24-8426.

Although active-passive solutions provide a way of moving your workloads to avoid planned outages, they do not eliminate unplanned outages. They allow you only a quicker method of recovering your applications and shortening the length of the outage. VM recovery options are operating system neutral, so they support all operating systems that run on IBM Power servers or IBM Power Virtual Server.

### **Active-passive solutions: clustering options**

In the active-passive solution, a cluster of servers is set up to provide redundant hardware that is available to take over the workloads that are running on a server during an outage. Because the server taking over the workload is already running, the amount of time that is required to recover your workloads is further shortened compared to the VM restart options. In addition, the clustering software can recover without an outage from many failures by using redundant components such as network adapters and disks. For AIX and IBM i, the PowerHA SystemMirror family of solutions is optimized for mission-critical applications where the total annual downtime for outages is zero or near-zero because the family covers all outage types for both software and hardware. There is at least one active OS on each of the nodes in the cluster, which enables the capability of doing software updates on a system other than the production node. PowerHA SystemMirror covers both data center and multi-site configurations. If you are looking to drive total outage time for events to near-zero, this solution is the one that you should deploy.

### **Active-active solutions: advanced clustering**

Active-active solutions provide a higher level of availability than either of the previous options, but come with a higher cost in hardware and software. In an active-active solution, the application is running on two or more servers that are concurrently accessing the same data. This concurrent access from multiple application instances requires either OS-based mirroring or database-based mirroring. IBM provides two active-active solutions for IBM Power:

- ▶ IBM Db2 pureScale® is available for AIX and Linux.
- ▶ IBM Db2 Mirror for i is available for IBM i.

Other clustering solutions are available from vendors such as Oracle and VERITAS.

Table 14-1 highlights the differences between IBM Power HADR solutions.

*Table 14-1 High availability topology classification*

| Technology     | Active-active clustering                                                                                                                                                                                                        | Active-passive clustering                                                                                                                                                                                                                           | Active-inactive clustering                                                                                                                                                                                                                                                                  |
|----------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Definition     | Application clustering: Applications in the cluster have simultaneous access to the production data, so there is no app restart when an app node outage occurs. Certain types enable read-only access from the secondary nodes. | OS clustering; One OS that is in the cluster has access to the production data, and there are multiple active OS instances on all nodes in the cluster. Applications are restarted on a secondary node after an outage of a production node occurs. | VM clustering: One VM in a cluster pair has access to the data, one logical OS, and two physical copies. The OS and applications must be restarted on a secondary node after a primary node outage event occurs. LPM enables the VM to be moved nondisruptively for a planned outage event. |
| Outage types   | Software, hardware, high availability (HA), planned, and unplanned. The RTO is 0 with limited distance.                                                                                                                         | Software, hardware, HADR, planned, and unplanned. The RTO is greater than 0 with multiple sites.                                                                                                                                                    | Hardware, HADR, planned, and unplanned. The RTO is greater than 0 with multiple sites.                                                                                                                                                                                                      |
| OS integration | Inside the OS.                                                                                                                                                                                                                  | Inside the OS.                                                                                                                                                                                                                                      | OS-neutral.                                                                                                                                                                                                                                                                                 |
| RPO            | Sync mode only.                                                                                                                                                                                                                 | Sync/Async.                                                                                                                                                                                                                                         | Sync/Async.                                                                                                                                                                                                                                                                                 |
| RTO            | Zero.                                                                                                                                                                                                                           | Fast (minutes).                                                                                                                                                                                                                                     | Fast enough (VM restart).                                                                                                                                                                                                                                                                   |
| Licensing      | N <sup>a</sup> +N.                                                                                                                                                                                                              | N+1 licensing.                                                                                                                                                                                                                                      | N+0 licensing.                                                                                                                                                                                                                                                                              |
| IBM solution   | Db2 pureScale, Db2 Mirror, and IBM Storage Scale.                                                                                                                                                                               | PowerHA SystemMirror, Red Hat HA, and Linux HA.                                                                                                                                                                                                     | VMRM HA, LPM, and VMRM DR.                                                                                                                                                                                                                                                                  |

a. The number of licensed processor cores on each system in the cluster.

### 14.1.3 Cloud HADR for IBM Power Virtual Server

IBM Power Cloud (IBM Power Virtual Server) consists of AIX, IBM i, and Linux workloads running on LPARs (VMs) on IBM Power9 hardware in IBM Cloud (some data centers offer IBM Power8). These IBM Power servers are managed by the PowerVM hypervisor and virtualized with dual Virtual I/O Server servers (VIOSs), NAT external network access, private internal networking, and N\_Port ID Virtualization (NPIV) attached storage. IBM Power Virtual Server is an IaaS offering that includes the underlying infrastructure, the OS, and some licensed products. However, there is no access to the Hardware Management Console (HMC), VIOSs, or storage subsystems. There are some HA features that you can use to place LPARs and OS mirroring of storage within a data center, but any DR solutions rely on OS-managed replication (Geographic Logical Volume Manager (GLVM)) or application-managed replication.

## 14.2 Disaster recovery replication methods for cloud

The following DR replication methods are available for IBM Power Virtual Server running AIX on IBM Cloud:

- ▶ Storage-based data mirroring with IBM Power Virtual Server Global Replication Service (GRS)
- ▶ OS-based data mirroring:
  - PowerHA SystemMirror for AIX Enterprise Edition with GLVM
  - IBM Storage Scale (previously named IBM Spectrum Scale) Active File Management (AFM) and AFM DR
- ▶ Database replication: (not covered further in this publication)
  - Oracle DataGuard
  - Oracle GoldenGate
  - Db2 HADR
  - SAP HANA System Replication

### 14.2.1 Storage-based data mirroring

To ensure business continuity in uncertain conditions, a secure, HADR solution is necessary. Global replication is a valuable feature for HADR because it keeps your data off-site and away from the premises. If the primary instance is destroyed by a catastrophic incident, such as a fire, storm, flood, or other natural disaster, your secondary data instance is secure off-premises, and you can retrieve your data.

#### IBM Power Virtual Server Global Replication Solution

IBM Power Virtual Server now supports a GRS that provides data replication for your workloads. GRS is based on IBM Storwize Global Mirror Change Volume asynchronous replication technology. GRS on IBM Power Virtual Server exposes to the cloud-based servers the application programming interface (API) and command-line interface (CLI) to create and manage replication-enabled volumes.

Here are some of the benefits of global replication on IBM Power Virtual Server:

- ▶ Maintain a consistent and recoverable copy of the data at the remote site, which was created with minimal impact to applications at your local site.
- ▶ Efficiently synchronize the local and remote sites with support for failover and fallback modes, which help to reduce the time that is required to switch back to the local site after a planned or unplanned outage.
- ▶ Replicate more data in less time to remote locations.
- ▶ Maintain redundant data centers in distant geographies for rapid recovery from disasters.
- ▶ Eliminate costly dedicated networks for replication and avoid bandwidth upgrades.

**Restriction:** At the time of writing, GRS is enabled in two data centers (DAL12 and WDC06). More sites will be added over time. For more information, see [A Global Replication Service Solution Using IBM Power Virtual Server](#).

GRS aims to automate the complete DR solution by providing the API and CLI interfaces to create the recipe for the DR solution. GRS does not have a user interface (UI). IBM provides an automation toolkit for GRS.

The IBM Toolkit for AIX from Technology Services enables clients to automate DR functions and capabilities by integrating IBM Power Virtual Server with the capabilities of GRS. With the Toolkit, clients can manage their DR environment by using their existing AIX skills. The benefits include the following items:

- ▶ Simplify and automate operations of your multi-site DR solution in IBM Power Virtual Server.
- ▶ Provide a secondary IBM Cloud site as a host for your business application as a DR solution.

## 14.2.2 OS-based data mirroring

This section describes data mirroring solutions.

### **PowerHA SystemMirror for AIX Enterprise Edition with GLVM**

PowerHA SystemMirror for AIX uses the host-based mirroring feature that is called GLVM. GLVM is IP address-based replication instead of storage-based replication. At the time of writing, public cloud deployments do not support storage-based replication, so you must convert to GLVM.

GLVM uses caching in memory and backup on disk, so system capacity sizing is critical. Likewise, source and target system throughput must be closely matched. If you want to ensure that sufficient capacity is available in the public cloud, you must license as many processor cores as needed to conduct production operation at the required performance.

Figure 14-1 on page 569 provides an overview of PowerHA SystemMirror Enterprise Edition with GLVM on the IBM Power Virtual Server architecture. It consists of the following items:

- ▶ Traditional AIX Logical Volume Manager (LVM) native mirroring is replicated over IP addresses to the secondary system to maintain two (or three) identical copies in sync mode, and near-identical copies in async mode.
- ▶ The architecture is disk subsystem neutral, and it is implemented through Remote Physical Volumes (RPVs), which virtualize the remote disk to appear local. This architecture differs from logical unit numbers (LUNs), which are used through storage area network (SAN) storage.
- ▶ Easily managed by the administrator.
- ▶ You use term licensing for public cloud deployments because it is registered to the customer and not the serial number.
- ▶ Reserve sufficient capacity for running your production on a DR virtual server by licensing the number of cores that is required. N+1 licensing does not apply because expanded Capacity on Demand (CoD) cannot be ensured.

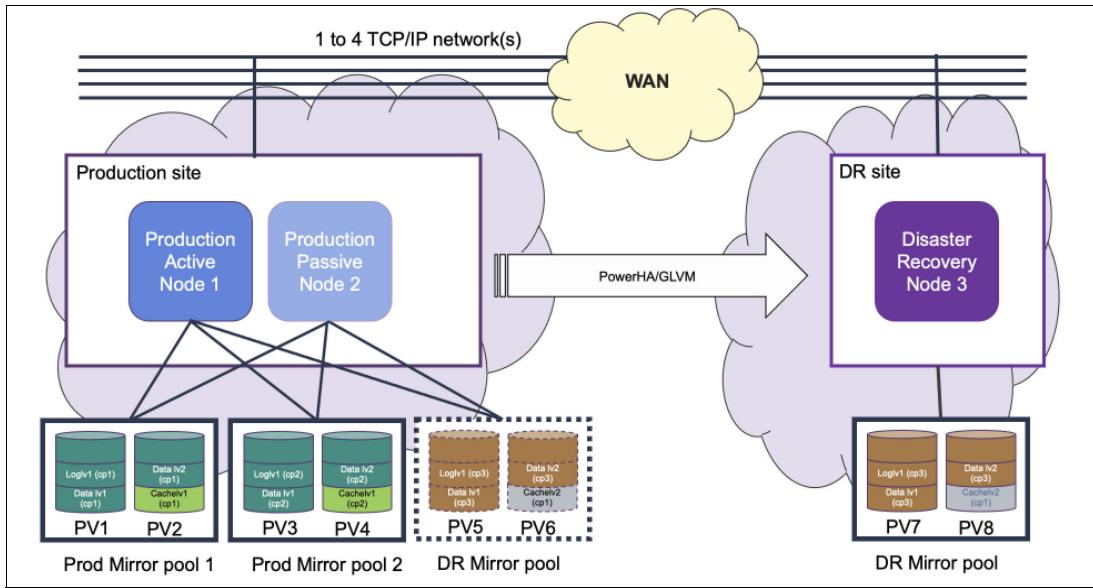


Figure 14-1 PowerHA SystemMirror for AIX on the IBM Power Virtual Server architecture

When using PowerHA SystemMirror for AIX with GLVM on IBM Power Virtual Server, consider the following items:

- ▶ There is no limit to the scalability from a production point of view if the systems are configured with sufficient disk, memory, and quality bandwidth. Source and target systems must be matched for equal throughput.
- ▶ For existing customers, you match the bandwidth and target configuration throughput to the on-premises deployments.

For more information, see *IBM Power Systems High Availability and Disaster Recovery Updates: Planning for a Multicloud Environment*, REDP-5663 and [AIX Disaster Recovery with IBM Power Virtual Server: An IBM Systems Lab Services Tutorial](#).

### 14.2.3 Geographic Logical Volume Manager concepts

GLVM provides software-based mirroring between two AIX systems over an IP network to protect against loss of data from the active site. GLVM works with any disk type that is supported by the AIX LVM. There is no requirement for the same type of disk subsystem at the source and destination, much like the AIX LVM can mirror between two different disk subsystems locally. GLVM also has no dependency on the type of data that is mirrored, and it supports both file systems and raw logical volumes (LVs).

The distance between the sites is limited only by the acceptable latency (for synchronous configurations) or by the size of the cache (for asynchronous configurations). For asynchronous replication, the size of the cache represents the maximum acceptable amount of data that can be lost in a disaster.

Figure 14-2 shows the GLVM architecture.

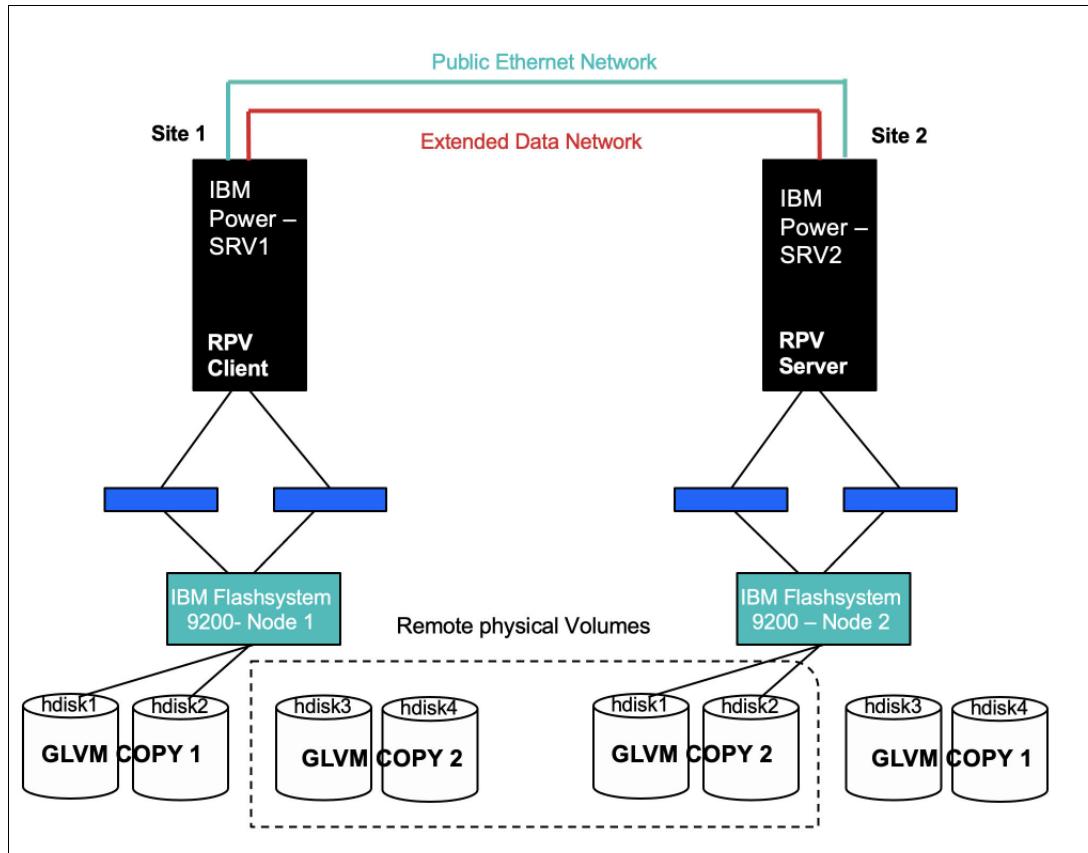


Figure 14-2 GLVM architecture

## Remote Physical Volume

In this architecture, it is the pseudo-local representation of the Remote Physical Volume (RPV) that allows the LVM to consider the physical volume (PV) at the remote site as another local, albeit slow, PV. The actual I/O operations are performed at the remote site.

### **The RPV client**

The RPV client is a pseudo-device driver that runs on the active server or site, that is, where the volume group (VG) is activated. There is one RPV client for each PV on the remote server or site, and it is named `hdisk#`. The LVM sees it as a disk and performs I/Os against this device. The RPV client definition includes the remote server address and timeout values.

### **The RPV server**

The RPV server is an instance of the kernel extension of the RPV device driver that runs on the node on the remote server or site, that is, on the node that has the PV. The RPV server receives and handles the I/O requests from the RPV client.

There is one RPV server for each replicated PV, and it is named `rpvserver#`.

### **The GLVM cache**

This cache is a special type of LV of type `aio_cache` that is designed for use in asynchronous mode GLVM. For asynchronous mode, rather than waiting for the write to be performed on the RPV, the write is recorded on the local cache, and then an acknowledgment is returned to the application. Later, the I/Os that are recorded in the cache are played in order against the remote disks and then deleted from the cache after successful acknowledgment.

### **Geographic Mirrored Volume Group**

A Geographic Mirrored Volume Group (GMVG) is an AIX VG that contains both local PVs and RPV clients.

You can mirror your data across two sites by configuring VGs that contain both local physical disks and RPVs. With an RPV device driver, the LVM does not distinguish between local and RPVs, that is, it maintains mirror copies of the data across attached disks. The LVM is unaware that some disks are at a remote site.

For PowerHA SystemMirror installations, GMVGs can be added to resource groups (RGs) and managed and monitored by PowerHA SystemMirror. Defining the GLVM VG is described in Chapter 15, “Geographic Logical Volume Manager wizard” on page 575.

For more information about GLVM, see the [PowerHA SystemMirror documentation](#) and *Asynchronous Geographic Logical Volume Mirroring Best Practices for Cloud Deployment*, REDP-5665.

#### **14.2.4 Block-storage based replication**

PowerHA SystemMirror software supports disk technology as an application-shared external disk in a HA cluster. For more information about the disk technologies that are supported by a particular version of the PowerHA SystemMirror and AIX OS, see the [PowerHA SystemMirror Hardware Support Matrix](#).

In a PowerHA SystemMirror cluster, a shared disk is an external disk that is attached to multiple cluster nodes, and it is used for application-shared storage.

In a nonconcurrent configuration, only one node owns the disk at a time. If the owner node fails, the next highest priority cluster node in the RG node list takes ownership of the shared disk and restarts the application to restore critical services to the client, which provides client applications access to the data that is stored on disk.

The takeover usually occurs within 30 – 300 seconds. This range depends on the number and types of disks that are used, the number of VGs and file systems (shared or cross-attached to a Network File System (NFS)), and the number of mission-critical applications in the cluster configuration.

When planning a shared external disk for a cluster, the goal is to eliminate single points of failure (SPOFs) in the disk storage subsystem.

PowerHA SystemMirror Enterprise Edition supports different block replication modes:

- ▶ Active-active: This type of relationship is created only for HyperSwap volumes. When HyperSwap is configured on the system, the HyperSwap volumes are on separate sites and an active-active relationship is automatically configured between them. Updates to the volumes in the relationship are updated simultaneously on both sites to provide DR solutions for the system.
- ▶ Metro Mirror is a type of remote copy that creates a synchronous copy of data from a primary volume to a secondary volume. A secondary volume can be either on the same system or on another system.

- ▶ The Global Mirror function provides an asynchronous copy process. When a host writes to the primary volume, confirmation of I/O completion is received before the write operation completes for the copy on the secondary volume.
- ▶ Global Mirror with Change Volumes provides the same basic function of asynchronous copy operations between source and target volumes for DR. A copy is taken of the primary volume in the relationship by using the change volume that is specified when the Global Mirror relationship with change volumes is created.

For more information, see [PowerHA SystemMirror 7.2 for AIX planning](#).

#### 14.2.5 File-storage based replication

IBM Storage Scale (previously known as IBM Spectrum Scale) is a cluster file system that allows several nodes to access a single file system or a group of file systems concurrently. The nodes can be SAN-attached, network-attached, a combination of SAN-attached and network-attached, or part of a shared-nothing cluster. This solution provides high-performance access to this shared data collection and can be used to support scale-out solutions or provide a HA platform.

**Important:** File-based replication and clustering are independent of PowerHA. They may be used in a complementary fashion to achieve a goal, but it is considered a customized solution with no formal integrated support.

In addition to general data access, IBM Storage Scale includes many other features, such as data replication, policy-based storage management, and multi-site operations. IBM Storage Scale can be run on virtualized instances, LPARs, or other hypervisors to enable common data access in scenarios. Multiple IBM Storage Scale clusters can share data over a local area network (LAN) or wide area network (WAN).

An IBM Storage Scale cluster with DR capabilities consists of two or three geographically different sites that work in a coordinated manner. Two of the sites consist of IBM Storage Scale nodes and storage resources that hold complete copies of the file system. If the third site is active, it consists of a single node and a single disk that are used as the IBM Storage Scale arbitration tie-breaker. The file system service fails over to the remaining subset of the cluster and uses the copy of the file system that survived the disaster to continue to provide data access in a hardware failure that causes the entire site to become inoperable, assuming that the arbitration site still exists.

IBM Storage Scale also supports asynchronous replication by using AFM, which is primarily designed for a head office or remote office configuration. It is available in IBM Storage Scale Standard Edition.

AFM provides a scalable, high-performance file system caching layer that is integrated with the IBM Storage Scale cluster file system. With AFM, you can create associations from a local IBM Storage Scale cluster to a remote cluster or storage, and define the location and flow of file data to automate the management of the data. You can implement a single namespace view across sites around the world.

AFM-based asynchronous DR (AFM DR) is a file-set-level replication DR capability. This capability is a one-for-one active-passive model that is represented by two sites: primary and secondary.

The primary site is a read/write file set where the applications are running and have read/write access to the data. The secondary site is read-only. All the data from the primary site is asynchronously synchronized with the secondary site. The primary and secondary sites can be independently created in a storage and network configuration. After the sites are created, you can establish a relationship between the two file sets. The primary site is available for the applications even when communication or secondary fails. When the connection with the secondary site is restored, the primary site detects the restored connection and asynchronously updates the secondary site.

For more information, see the [IBM Storage Scale 5.1.2 documentation](#).

## 14.3 Hybrid and multi-cloud deployment models

This section describes cloud deployment models.

### 14.3.1 Hybrid cloud

*Hybrid cloud* combines and unifies public cloud, private cloud, and the on-premises infrastructure to create a single, flexible, and cost-optimal IT infrastructure.

The advantage of this model is cost savings because enterprises can provision minimal resources and scale up as required during a HADR situation. In this scenario, the enterprises run their production workloads on-premises and use the resources in a public cloud for HADR, as shown in Figure 14-3.

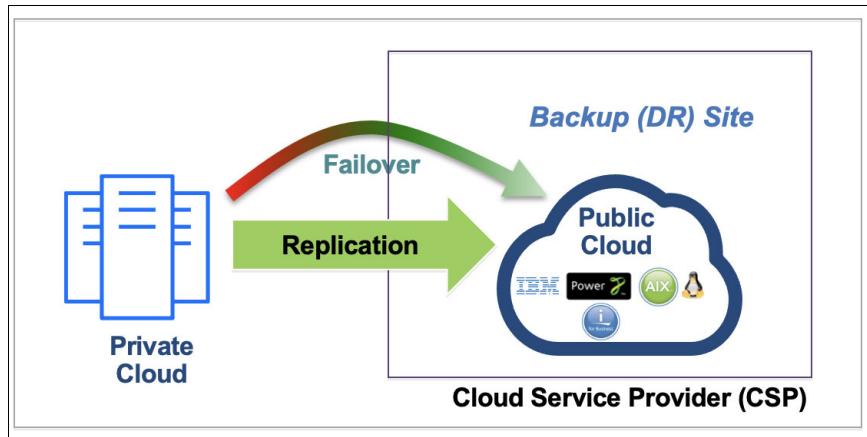


Figure 14-3 Hybrid cloud scenario

### 14.3.2 Multiple public clouds

*Multicloud* is when you use cloud services from two or more vendors. Multicloud provides organizations with more flexibility to optimize performance, control costs, and leverage the best cloud technologies that are available.

The main advantage of this model is enhanced resiliency. Outages can happen at any time for a cloud provider, which makes it risky for enterprises to rely on a single cloud vendor.

In this scenario, the enterprises run their production workloads on one public cloud and HADR on another public cloud, as shown in Figure 14-4.

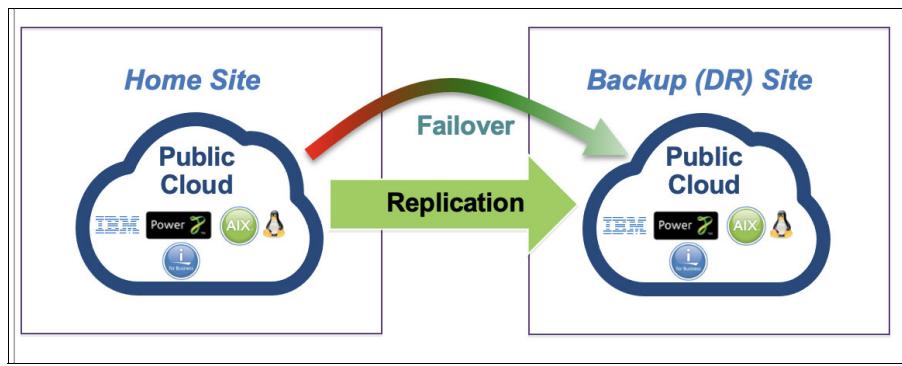


Figure 14-4 Multiple public cloud

### 14.3.3 Cold disaster recovery

This scenario is like the hybrid cloud scenario, but the VMs are deployed only during the HADR situation to minimize the HADR cost. This scenario can be a low-cost HADR solution.



# Geographic Logical Volume Manager wizard

This chapter contains the following topics regarding the PowerHA Geographic Logical Volume Manager (GLVM) configuration assistant (the GLVM wizard).

This chapter contains the following topics:

- ▶ Introduction
- ▶ Creating a cluster by using the SMUI
- ▶ Adding to an existing cluster
- ▶ GLVM cluster configuration
- ▶ Using the command-line interface

## 15.1 Introduction

The section provides an overview of using the PowerHA 7.2.7 SystemMirror User Interface (SMUI) GLVM configuration wizard. GLVM can be configured by either of the following methods:

- ▶ Cluster Creation

During cluster creation, the cluster creation wizard is used for GLVM configuration. As a prerequisite for the cluster creation wizard, you must have a volume group (VG) with its physical volumes (PVs) shared on all local site nodes. PVs of the same size must exist in the remote site and are automatically detected.

- ▶ Add to existing Cluster

The GLVM configuration wizard is used for linked clusters that are already managed by the PowerHA SMUI. You can create and manage multiple GLVM configurations in the same cluster by using the GLVM configuration wizard. PVs of the same size must exist in the remote site and are automatically detected.

The GLVM wizard is started by running the `c1_glvm_configuration` command, which is in the `/usr/es/sbin/cluster/glvm/utils` directory. The command can be run manually from the CLI, through the System Management Interface Tool (SMIT), or through the SMUI. The primary focus on this chapter is using the SMUI. The wizard itself requires an existing base cluster, and the SMUI can create a base cluster and configure GLVM simultaneously.

### 15.1.1 Prerequisites

Before attempting to use the SMUI to configure GLVM, ensure that the following prerequisites are met.

- ▶ Ensure that `cluster.es.assist.common` is installed.
- ▶ The VG must be available at all local site nodes and must not use any PVs that are shared on any nodes in the remote site.
- ▶ IP addresses and labels are in the `/etc/hosts` directory of all nodes.
- ▶ There must be enough of the same-sized disks and space that are available at a remote site to accommodate mirroring. Account for the space that is needed for the AIO cache logical volume (LV) too.
- ▶ There must be an available disk of at least 512 GB at each site for the cluster repository disk.
- ▶ Ensure that XD\_data networks are defined in the cluster (when adding to an existing cluster).
- ▶ Python 2.0.x or later must be installed on all the cluster nodes.

**Note:** To delete a GLVM configuration from the PowerHA SMUI, bring the cluster offline first.

## 15.1.2 Limitations

The PowerHA SMUI GLVM configuration wizard has the following limitations:

- ▶ Supports only asynchronous mirroring for GLVM.
- ▶ After you configure GLVM, you cannot modify the asynchronous cache size.
- ▶ The asynchronous cache utilization graph is updated only when the corresponding resource group (RG) is active and when the application controller is active.
- ▶ The asynchronous cache utilization graph is not visible when GLVM is configured outside of the GLVM wizard.
- ▶ Service IP addresses must be created and added to RG after the cluster and RG creation as needed.
- ▶ The Split/Merge policy defaults to the majority. You might want to change it.

## 15.1.3 Logs

The following logs are useful for observing and troubleshooting when you use the GLVM wizard through the SMUI:

- ▶ `/var/hacmp/log/c1_glvm_configuration.log`  
The primary log of the GLVM wizard on the primary node where the wizard runs.
- ▶ `/usr/es/sbin/cluster/ui/agent/logs/smui-agent.log`  
The SMUI log on the cluster node where the tasks run.
- ▶ `/usr/es/sbin/cluster/ui/server/logs/smui-server.log`  
The SMUI log on the SMUI server itself.

## 15.2 Creating a cluster by using the SMUI

To create an asynchronous GLVM cluster from the PowerHA SMUI, complete the following steps:

1. Log in to the SMUI server.

2. Select **Health Summary** → **Create a New Cluster**, as shown in Figure 15-1.

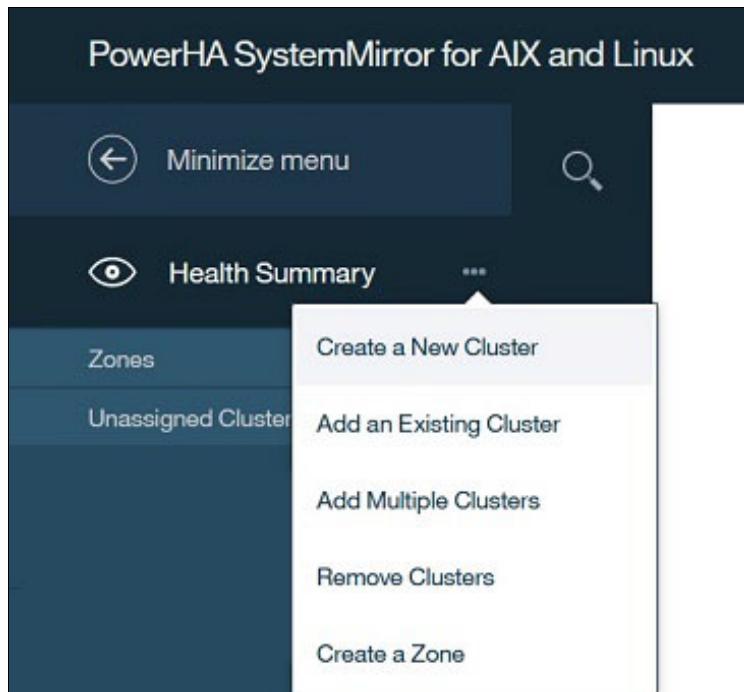


Figure 15-1 Create a New Cluster

3. Specify the primary node and its login credentials and click **Continue**, as shown in Figure 15-2.

A screenshot of the "Create a Cluster" wizard, step 1: Node Authentication. The title bar says "PowerHA SystemMirror for AIX and Linux" and "Hello root". The main title is "Create a Cluster". Below it is a navigation bar with steps: 1. Node Authentication → 2. Cluster Settings → 3. Assign Nodes → 4. Summary. The first step is active. The form asks for a node for the new cluster, specifying a valid login credential. It includes a note: "Specify a node for the new cluster, along with valid login credentials for that node. The node that you specify is used to collect information about the new cluster environment." and a required field indicator: "\*Required field". The form fields include:

- "Hostname or IP Address\*" input field containing "jordan".
- "User ID (root access required)\*" input field containing "root".
- "Select authentication type\*" radio buttons:
  - Password
  - Private key file
  - Private key file with passphrase
- "Password\*" input field containing "\*\*\*\*\*".
- A checkbox: "Do you want to enable GUI server backup communication? ?" which is unchecked.

At the bottom are "Cancel" and "Continue" buttons.

Figure 15-2 Specifying the primary node login credentials

4. Enter the cluster name and select **Do you want to configure GLVM?** For the type of cluster, select **Linked**. Click **Continue**, as shown in Figure 15-3.

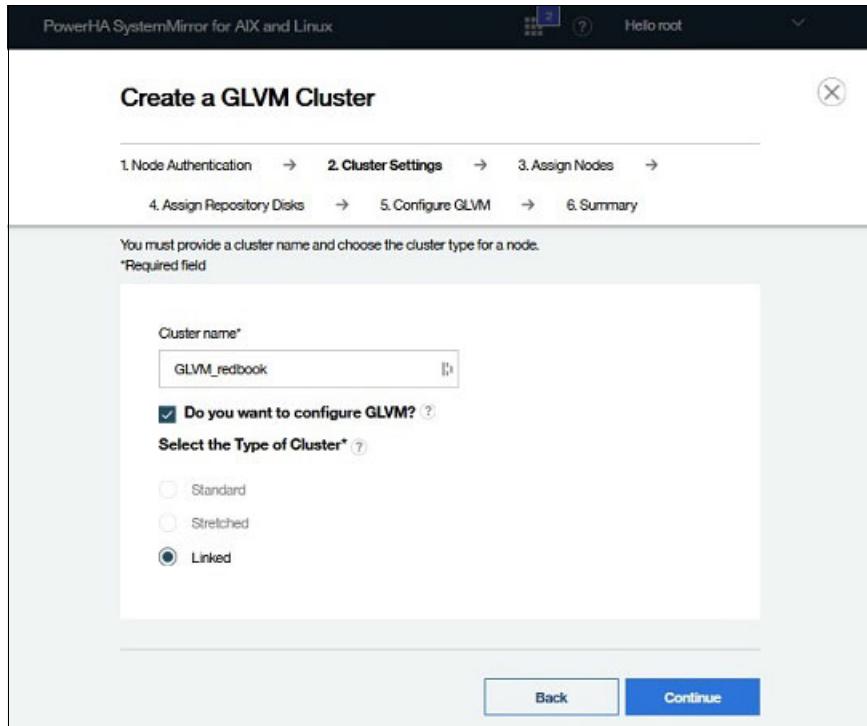


Figure 15-3 GLVM Cluster Name

A GLVM overview of requirements is shown in Figure 15-4.

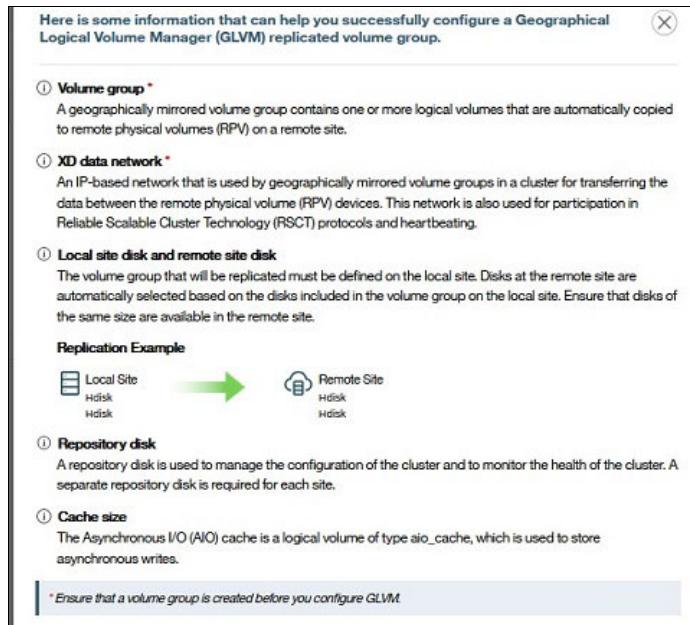


Figure 15-4 GLVM requirements

5. Specify the XD\_data network name, site names, nodes per site, and persistent alias. Click **Continue**, as shown in Figure 15-5.

The screenshot shows the 'Create a GLVM Cluster' wizard at step 6: Summary. The configuration details are as follows:

- XD Data Network Name\***: GLVM\_net
- Local Site**:
  - Local Site Name\***: Texas
  - Add a Node\***:
    - Enter hostname\*: jordan
    - Assign custom name: (empty)
    - Persistent IP Address?: jordan\_pers
- Remote Site**:
  - Remote Site Name\***: Maine
  - Add a Node\***:
    - Enter hostname\*: jessica
    - Assign custom name: (empty)
    - Persistent IP Address?: jessica\_pers

At the bottom right are 'Back' and 'Continue' buttons.

Figure 15-5 GLVM network and site definitions

6. Choose a repository disk for each site and click **Continue**, as shown in Figure 15-6.

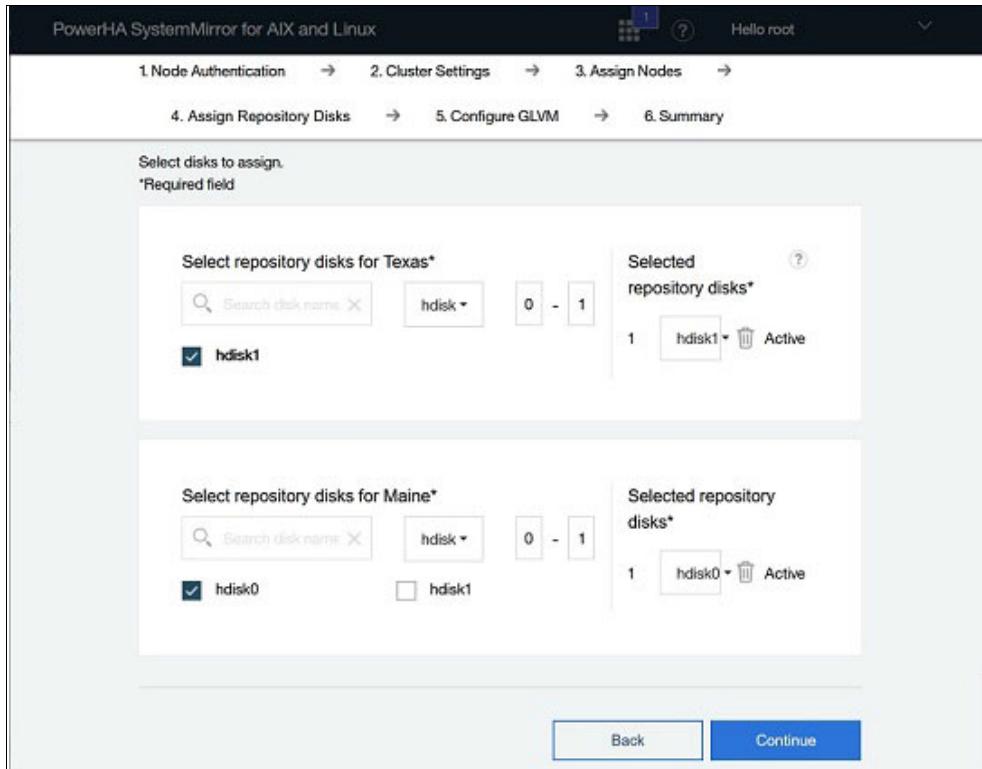


Figure 15-6 Site repository disks

7. Select a VG. Specify the asynchronous I/O cache size. Select **Disable** for the Compression radio button. Complete the I/O Group Latency and the Number of Parallel Logical volumes fields. Click **Continue**, as shown in Figure 15-7 on page 583.

Here are the descriptions for some of the options:

|                               |                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Async I/O Cache LV</b>     | An LV of type aio_cache that is used to store asynchronous writes. Instead of waiting for the write to complete on the Remote Physical Volume (RPV), the write is recorded on the local cache, and then acknowledgment is returned to the application. The I/Os that are recorded in the cache are played in order against the remote disks and then deleted from the cache after a successful write acknowledgment. |
| <b>I/O Group Latency</b>      | This parameter indicates the maximum expected delay (in milliseconds) before you receive the I/O acknowledgment in a mirror pool. Specify the VG that is associated with the mirror pool by using the -v flag. The default value is 10 milliseconds. If you specify lower values, I/O performance might improve at the cost of higher CPU consumption. This attribute is a specific VG attribute.                    |
| <b>Number of Parallel LPs</b> | The number of logical partitions (LPARs) to be synchronized in parallel. The valid range is 1 - 32. The number of parallel LPARs must be tailored to the machine, disks in the VG, system resources, and the VG mode.                                                                                                                                                                                                |

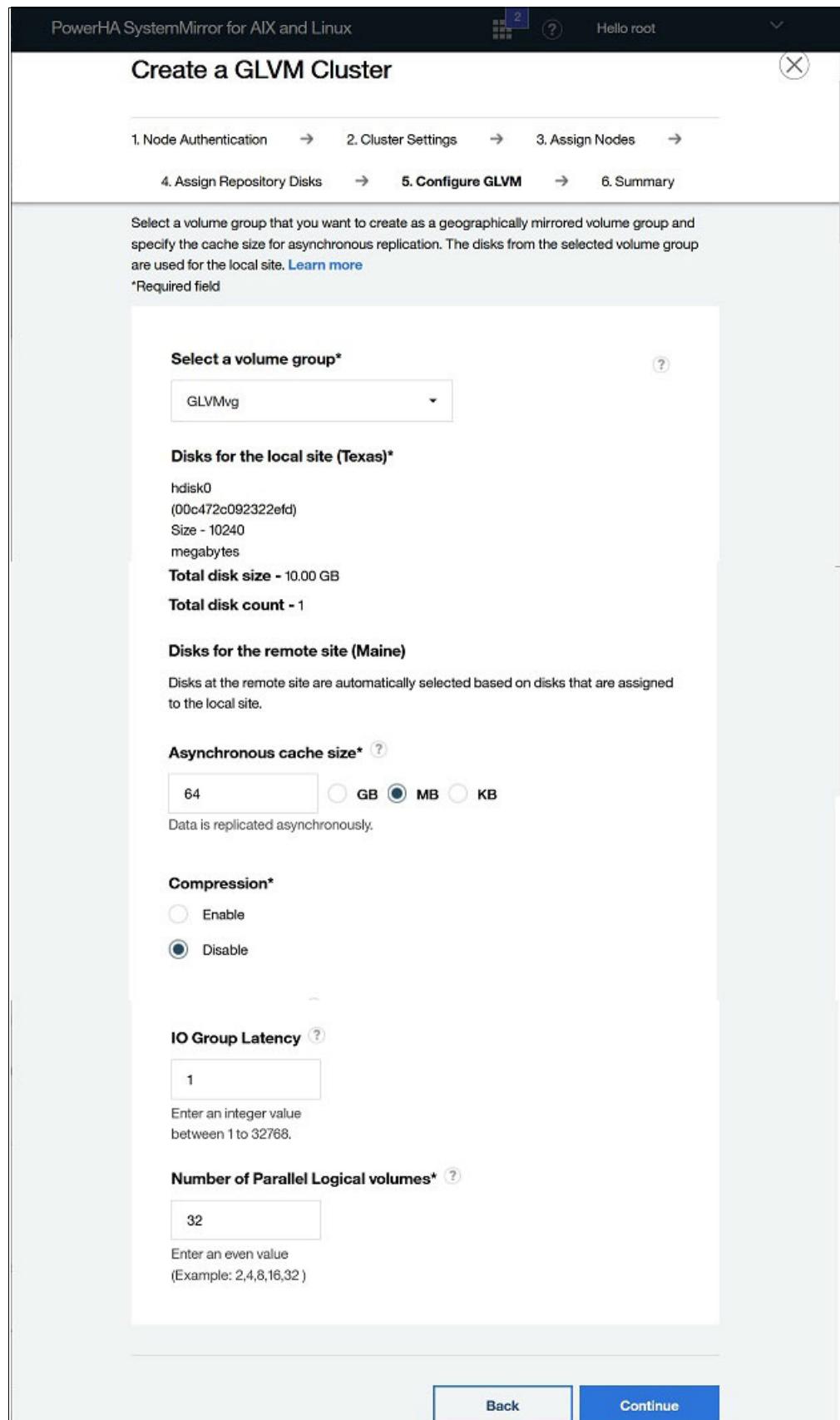


Figure 15-7 AIO cache LV size, compression, IO Group Latency, and Parallel Sync

8. The cluster summary is displayed. If everything looks correct, click **Submit**, as shown in Figure 15-8.

Summary of your specifications for the new GLVM cluster configuration.

| Cluster Settings |              |                    |
|------------------|--------------|--------------------|
| Cluster name*    | Cluster Type | Communication Type |
| GLVM_redbook     | Linked       | Unicast            |

| Node Authentication    |         |                     |
|------------------------|---------|---------------------|
| Hostname or IP Address | User ID | Authentication Type |
| jordan                 | root    | Password            |

| Assigned Nodes       |                       |  |
|----------------------|-----------------------|--|
| XD Data Network Name |                       |  |
| GLVM_net             |                       |  |
| Texas                |                       |  |
| Nodes                | Persistent IP Address |  |
| jordan               | jordan_pers           |  |
| Maine                |                       |  |
| Nodes                | Persistent IP Address |  |
| jessica              | jessica_pers          |  |

| Assigned Repository Disk |        |  |
|--------------------------|--------|--|
| Site 1                   | Site 2 |  |
| hdisk1                   | hdisk0 |  |

| GLVM Configuration      |                  |                                    |
|-------------------------|------------------|------------------------------------|
| Volume Group            |                  |                                    |
| GLVMvg                  |                  |                                    |
| Local Disk              | Remote Disk      |                                    |
| hdisk0                  | hdisk1           |                                    |
| Asynchronous cache size |                  |                                    |
| 64 GB                   |                  |                                    |
| Compression             | IO Group Latency | Number of Parallel Logical volumes |
| Disabled                | 1                | 32                                 |


Figure 15-8 Cluster summary

The cluster was successfully created, as shown in Figure 15-9.

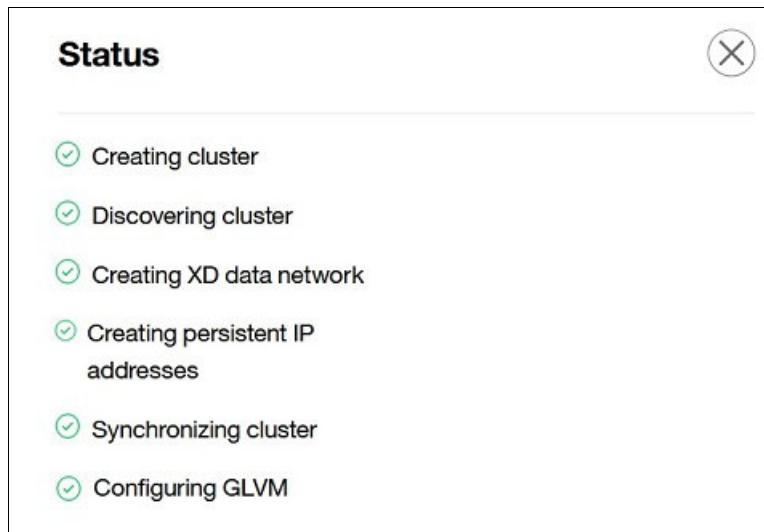


Figure 15-9 Successful creation

**Note:** At the time of writing, the SMUI summary window always displays the asynchronous I/O cache size in GB regardless what was chosen. However, the size is what you specified in GB, MB, or KB.

For more information, see 15.4, “GLVM cluster configuration” on page 595.

## 15.3 Adding to an existing cluster

In this scenario, we start with an existing 2-node linked cluster with the following configuration:

- ▶ Two sites
- ▶ One node at each site
- ▶ Two networks
- ▶ A repository disk at each site

The exact details are shown in Example 15-1.

Example 15-1 Base beginning cluster configuration

---

```
Cluster Name: GLVM_redbook
Cluster Type: Linked
Heartbeat Type: Unicast
Repository Disks:
 Site 1 (Texas@jordan): hdisk1
 Site 2 (Maine@jessica): hdisk0
Cluster Nodes:
 Site 1 (Texas):
 jordan
 Site 2 (Maine):
 jessica
```

```

There are 2 nodes and 2 networks defined
NODE jessica:
 Network GLVM_net
 jessica 10.2.30.190
 Network net_ether_010
 jessica_xd 192.168.100.90
NODE jordan:
 Network GLVM_net
 jordan 10.2.30.191
 Network net_ether_010
 jordan_xd 192.168.100.91

```

---

### 15.3.1 Using SMIT

When using SMIT, the GLVM wizard offers both replication methods, asynchronous and synchronous. Our example demonstrates the asynchronous method. There are a couple of differences between using SMIT compared to the SMUI.

The SMIT version creates the Geographic Mirrored Volume Group (GMVG) as new, but SMUI requires an existing VG (preferably with all the necessary LVs and file systems already created). The SMIT version creates a Journaled File System 2 (JFS2) and Journaled File System log (JFSlog), in addition to the async I/O LVs, as shown in Example 15-2.

*Example 15-2 Successful creation output*

```

WARNING: While creating ASYNC cache on node jordan, cache size 64 M is less than
40 percent of total disk space 4.0.WARNING: While creating ASYNC cache on node
jordan, cache size 32 M is less than 40 percent of total disk space 4.0.
Extracting the name for nodes from both local and remote sites.
Volume group GLVMvg is not configured to the cluster nodes jordan.
Continue creating the volume group on all the nodes.
RPVSitename is configured on node jordan, which is not the same as the
SystemMirror sitename.
PowerHA SystemMirror modifies RPVSitename to map it to sitename.
Configuring RPVsitenames.
Creating RPVServers on all nodes of the local site.
Creating RPVServers on node jordan.
Creating RPVServers on all nodes of the remote site.
Creating RPVServers on node jessica.
Successfully created RPVServers on all nodes of local and remote sites for volume
group:
GLVMvg.
Creating RPVClients on all nodes of the local site.
Creating RPVClients on node jordan.
Creating RPVClients on all nodes of the remote site.
Creating RPVClients on node jessica.
Successfully created RPVClients on all nodes of local and remote sites for volume
group:
GLVMvg.
Changing RPVServers and RPVClients to defined and available state accordingly to
facilitate the creation of VG.
INFO: Changing RPVServer rpvserver0 to defined state on node jordan.
Changing RPVClient hdisk2 to defined state on node jessica
Generating Unique Names for Mirror pools and Resource Group.
Unique names generated.

```

```
Created the GLVMvg volume group on node: jordan successfully
Creating the first mirror pool GLVMvgMP
Extending the VG to RPVClient disks and creating second mirror pool GLVMvgMP1
Creating the first ASYNC cache LV GLVMvgALV
Creating second ASYNC cache LV GLVMvgALV1
Setting attributes for GLVMvg on node jordan
Setting high water mark for mirror pool GLVMvgMP1
Setting high water mark for mirror pool GLVMvgMP
Varying on volume group: GLVMvg on node: jordan
Setting attributes for GLVMvg on node jordan
Setting attributes for GLVMvgALV1 on node jordan
Setting attributes for GLVMvgALV on node jordan
Varying off volume group: GLVMvg on node: jordan
Created async volume group GLVMvg on jordan successfully.
Changing RPVClient hdisk2 to defined state on node jordan
Changing RPVClient hdisk2 on node jordan to available state
Export the volume group GLVMvg on node jordan.
Importing the VG GLVMvg on node jordan
Varying on volume group: GLVMvg on node: jordan
Setting attributes for GLVMvg on node jordan
Varying off volume group: GLVMvg on node: jordan
Changing RPVClient hdisk2 to defined state on node jordan
Changing RPVServer rpvserver0 to defined state on node jessica
Changing RPVServer rpvserver0 on node jordan to available state
Importing the VG GLVMvg on node jessica
Changing RPVClient hdisk2 on node jessica to available state
Importing the VG GLVMvg on node jessica
Varying on volume group: GLVMvg on node: jessica
Setting attributes for GLVMvg on node jessica
Varying off volume group: GLVMvg on node: jessica
Changing RPVClient hdisk2 to defined state on node jessica
The definition of VG is available on all the nodes of the cluster.
Changing RPVServer rpvserver0 to defined state on node jordan
Generating resource group (RG) name.
Creating a resource group GLVMvg_RG.
Adding VG GLVMvg to RG GLVMvg_RG.
INFO: Successfully created the application monitor: GLVMvg_RG_GLVM_mon.
INFO: Successfully created the application server: GLVMvg_RG_GLVM_serv.
INFO: Successfully added application server GLVMvg_RG_GLVM_serv to Resource group
GLVMvg_RG.
INFO: Successfully created file collection for the GLVM resource group GLVMvg_RG.
INFO: Successfully updated rpvtunable compression=0 on node jordan.
INFO: Successfully updated rpvtunable compression=0 on node jessica.
INFO: Successfully updated GLVM tunable NO_PARALLEL_LPS for resource group
GLVMvg_RG.
Verifying and synchronising the cluster configuration ...
```

---

**Note:** The SMIT panel that is shown in Figure 15-10 does *not* prompt you about the usage of compression, I/O group latency, or the number of LPARs to sync in parallel like the SMUI does. These items are set to the defaults 0 (disabled), 1, and 32.

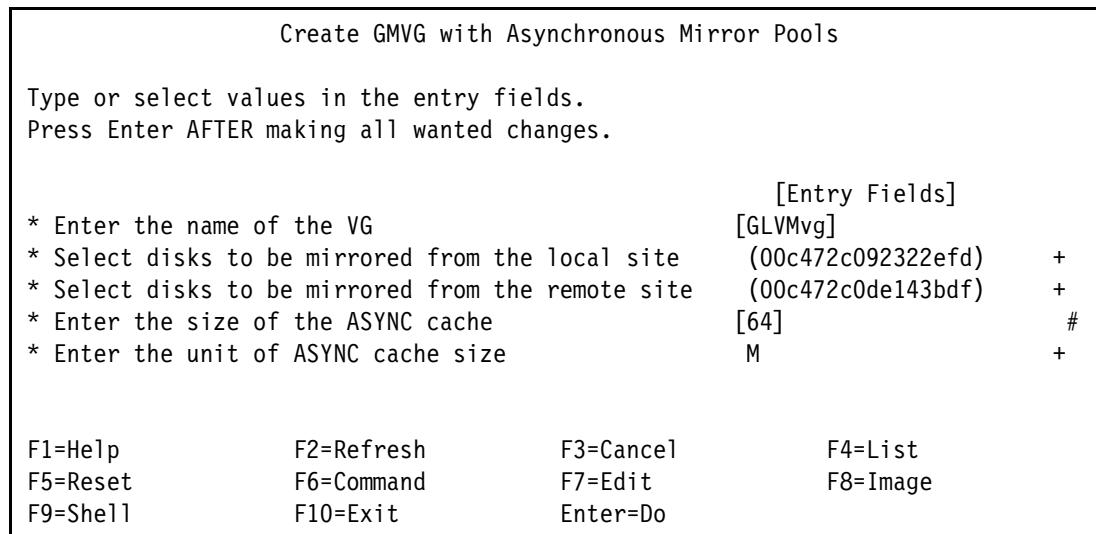


Figure 15-10 Asynchronous GLVM wizard by using SMIT

To add an asynchronous GMVG into an existing cluster by using the GLVM wizard, run **smitty sysmirror**, and select **Cluster Applications and Resources** → **Make Applications Highly Available (Use Smart Assists)** → **GLVM Configuration Assistant** → **Configure Asynchronous GMVG**.

After the process completes, an overview of all the tasks that ran is shown in the SMIT screen, as shown in Example 15-2 on page 586.

The overall configuration that is created is nearly identical to the SMUI results that are shown in 15.4, “GLVM cluster configuration” on page 595, except for the creation of a JFS2 file system and a jfs2log, as shown in Example 15-3.

Example 15-3 Logical volumes and JFS2 created from asynchronous GLVM wizard

| LVMvg:     |           |     |     |     |              |             |
|------------|-----------|-----|-----|-----|--------------|-------------|
| LV NAME    | TYPE      | LPs | PPs | PVs | LV STATE     | MOUNT POINT |
| GLVMvgLV   | jfs2      | 10  | 20  | 2   | closed/syncd | /GLVMvgfs0  |
| GLVMvgLV1  | jfs2log   | 10  | 20  | 2   | closed/syncd | N/A         |
| GLVMvgALV  | aio_cache | 8   | 8   | 1   | open/syncd   | N/A         |
| GLVMvgALV1 | aio_cache | 8   | 8   | 1   | closed/syncd | N/A         |

**Note:** The full command syntax for this example is as follows:

```
c1_glvm_configuration -v 'GLVMvg' -l '(00c472c092322efd)' -r
'(00c472c0de143bdf)' -s '64' -u 'M'
```

### 15.3.2 Using the SMUI to add an asynchronous GLVM configuration

To add an asynchronous GLVM configuration to the existing cluster by using the GLVM wizard through the SMUI, complete the following steps:

1. Log in to the SMUI server.
2. Find and click the cluster and select **Health Summary** → **Configure GLVM**, as shown in Figure 15-11.

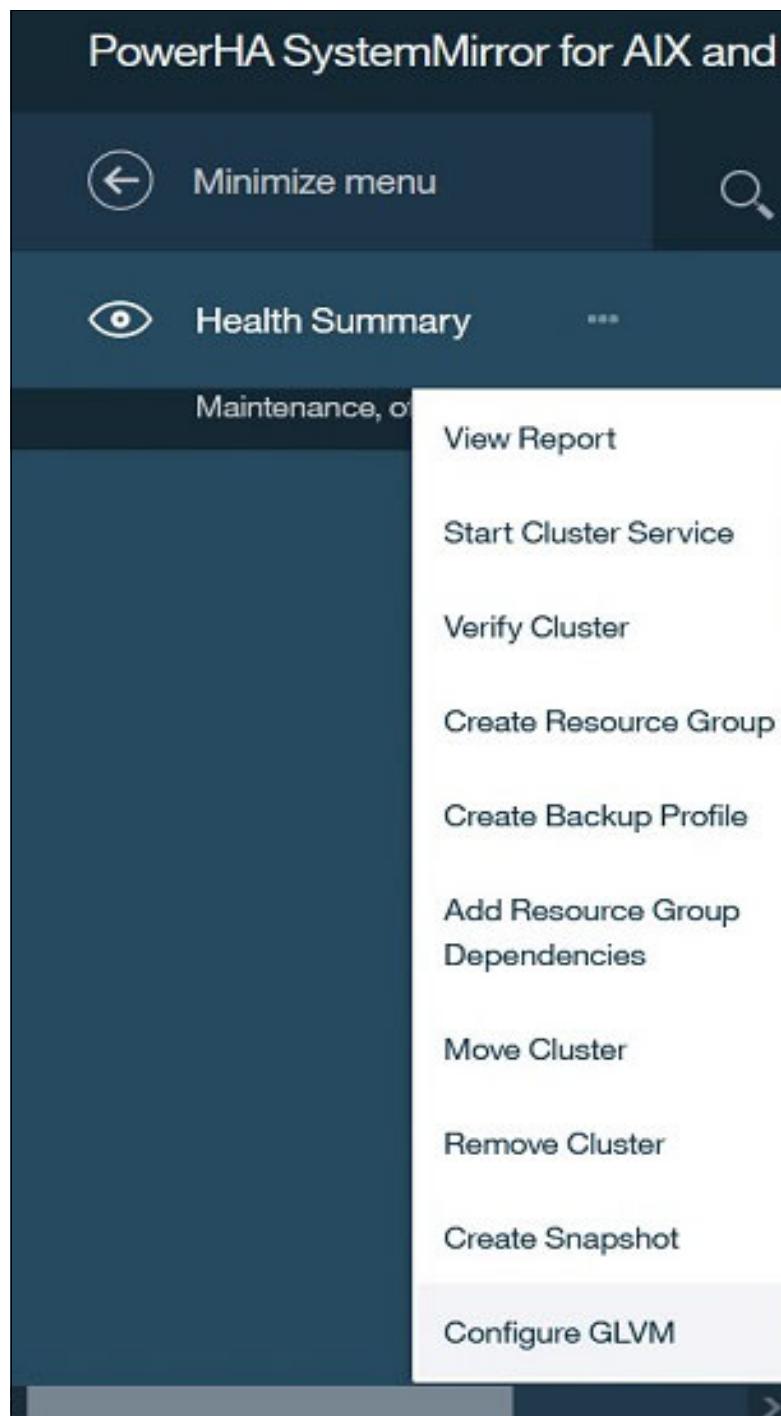


Figure 15-11 Adding a GLVM to a cluster

A GLVM informational window opens, as shown in Figure 15-12.

**Here is some information that can help you successfully configure a Geographical Logical Volume Manager (GLVM) replicated volume group.**

**i Volume group \***  
A geographically mirrored volume group contains one or more logical volumes that are automatically copied to remote physical volumes (RPV) on a remote site.

**i XD data network \***  
An IP-based network that is used by geographically mirrored volume groups in a cluster for transferring the data between the remote physical volume (RPV) devices. This network is also used for participation in Reliable Scalable Cluster Technology (RSCT) protocols and heartbeating.

**i Local site disk and remote site disk**  
The volume group that will be replicated must be defined on the local site. Disks at the remote site are automatically selected based on the disks included in the volume group on the local site. Ensure that disks of the same size are available in the remote site.

**Replication Example**

The diagram illustrates the replication setup. On the left, under 'Local Site', there are three icons representing 'Hdisk'. On the right, under 'Remote Site', there are also three icons representing 'Hdisk'. A large green arrow points from the Local Site to the Remote Site, indicating the direction of data transfer.

**i Cache size**  
The Asynchronous I/O (AIO) cache is a logical volume of type aio\_cache, which is used to store asynchronous writes.

*\* Ensure that a volume group and an XD data network are created before you configure GLVM.*

Figure 15-12 GLVM informational window

3. Specify the local site XD\_Data network and the persistent IP aliases if they are not already defined in the cluster topology. Our cluster already has the persistent aliases, so no further selection is required. Click **Continue**, as shown in Figure 15-13.

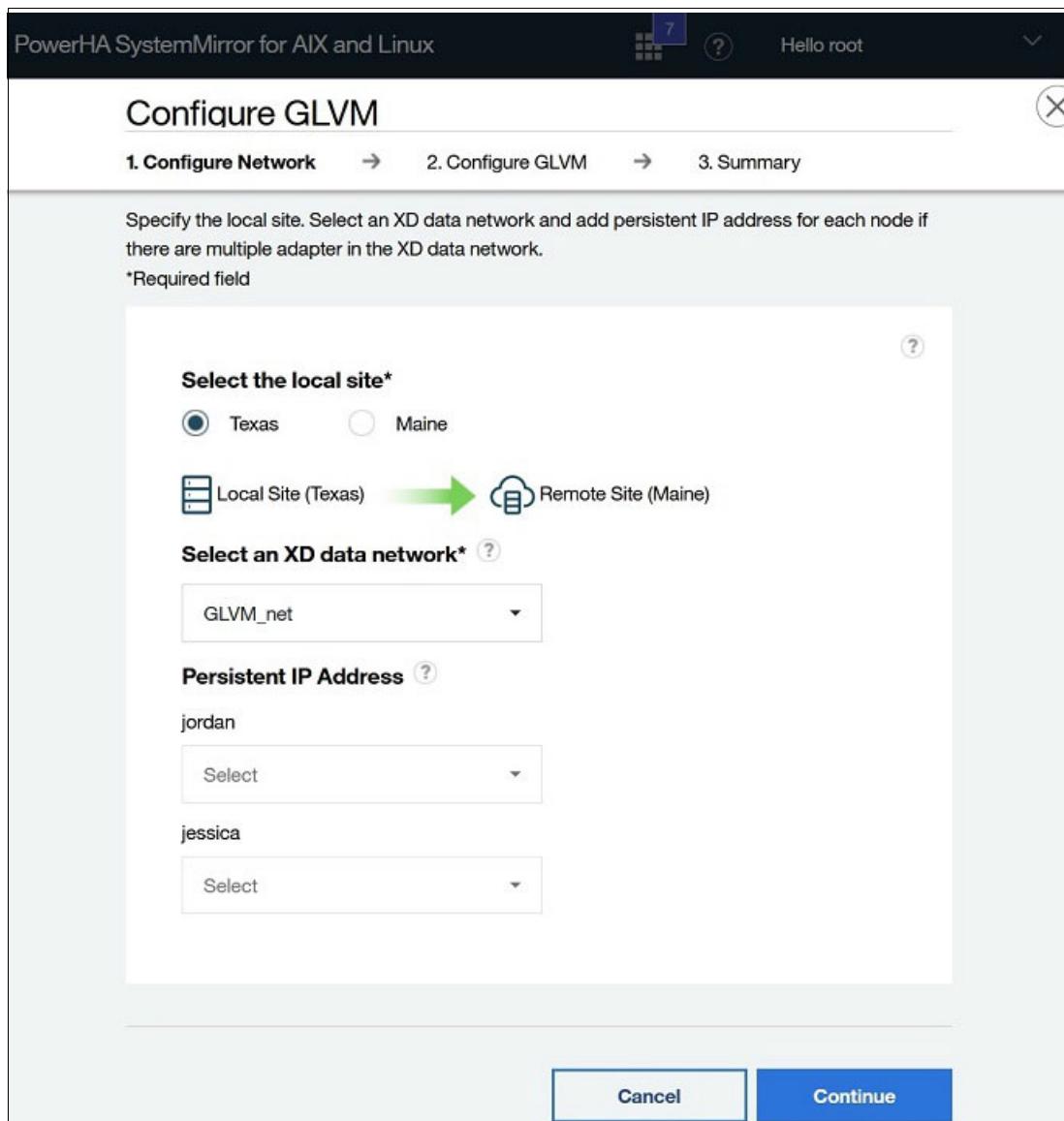


Figure 15-13 GLVM site and network configuration

4. Select a VG. Specify the asynchronous I/O cache size. Select **Disable** for the Compression radio button. Complete the I/O Group Latency and the Number of Parallel Logical volumes fields. Click **Continue**, as shown in Figure 15-14.

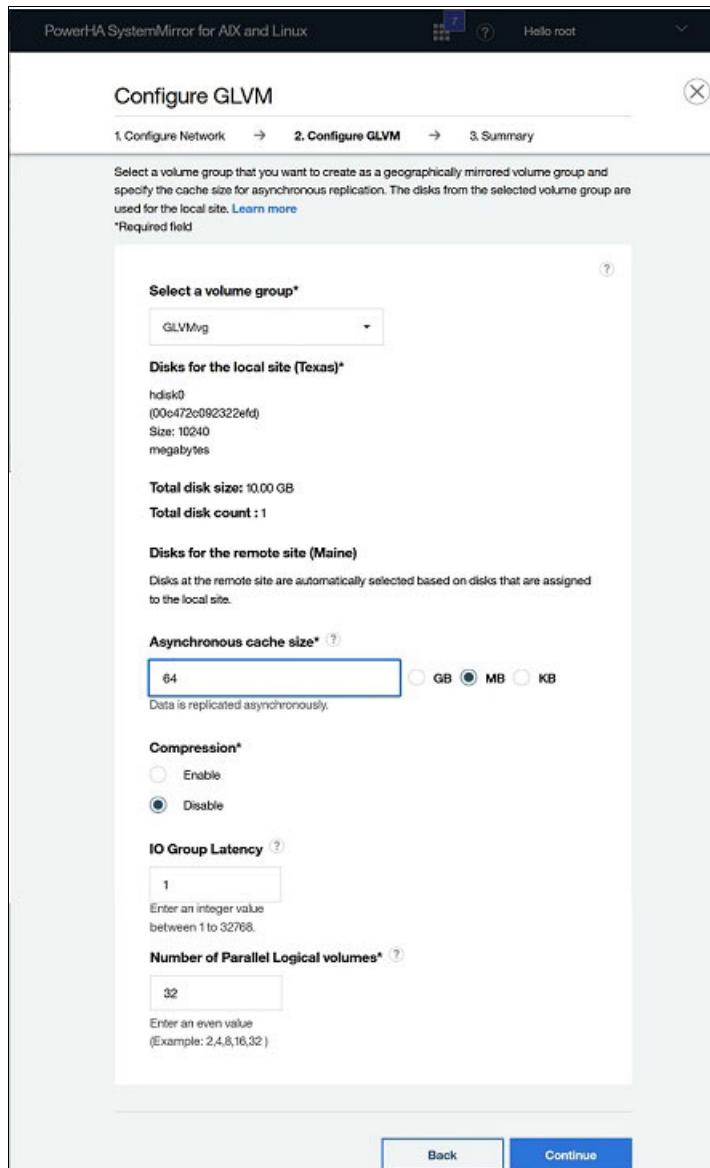


Figure 15-14 GMVG settings

Here are the descriptions for some of the options:

|                               |                                                                                                                                                                                                                                                                                                                                                                                                                      |
|-------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>Async I/O Cache LV</b>     | An LV of type aio_cache that is used to store asynchronous writes. Instead of waiting for the write to complete on the Remote Physical Volume (RPV), the write is recorded on the local cache, and then acknowledgment is returned to the application. The I/Os that are recorded in the cache are played in order against the remote disks and then deleted from the cache after a successful write acknowledgment. |
| <b>I/O Group Latency</b>      | This parameter indicates the maximum expected delay (in milliseconds) before you receive the I/O acknowledgment in a mirror pool. Specify the VG that is associated with the mirror pool by using the -v flag. The default value is 10 milliseconds. If you specify lower values, I/O performance might improve at the cost of higher CPU consumption. This attribute is a specific VG attribute.                    |
| <b>Number of Parallel LPs</b> | The number of logical partitions (LPARs) to be synchronized in parallel. The valid range is 1 - 32. The number of parallel LPARs must be tailored to the machine, disks in the VG, system resources, and the VG mode.                                                                                                                                                                                                |

**Note:** Even though the last option says “Number of Parallel Logical volumes”, it technically is “Number of Logical Partitions to Synchronize in Parallel”. It is directly related to the **syncvg** command and specified either by using the -P flag or the environment variable NUM\_PARALLEL\_LPS.

5. A summary is displayed. Review and click **Submit**, as shown in Figure 15-15.

Configure GLVM

1. Configure Network → 2. Configure GLVM → 3. Summary

Summary of your specifications for the new GLVM cluster configuration.

**Network Configuration**

**XD Data**  
**Network Name**  
GLVM\_net

**Texas**

**Node**  
jordan

**Maine**

**Node**  
jessica

**GLVM Configuration**

**Volume Group**  
GLVMvg

**Local Site (Texas)**      **Remote Site (Maine)**

hdisk0      hdisk1

**Asynchronous cache size**  
64 GB

| <b>Compression</b> | <b>IO Group Latency</b> | <b>Number of Parallel Logical volumes</b> |
|--------------------|-------------------------|-------------------------------------------|
| Disabled           | 1                       | 32                                        |

Back      Submit

Figure 15-15 GLVM cluster summary

**Note:** At the time of writing, the SMUI summary screen always displayed the asynchronous I/O cache size in GB regardless of what was chosen on previous windows. However, the properly specified size in GB, MB, or KB is used.

## 15.4 GLVM cluster configuration

Regardless of whether you use SMUI or SMIT, the resulting cluster configuration is the same. This section describes the configuration.

### 15.4.1 Topology

In our scenario, we have two networks. When you create the cluster, the default settings make both networks type XD\_data. There is also a persistent IP address that is associated with one of the XD\_data networks that were specified by the GLVM wizard. You might want to change which interface in which the persistent IP address is, or change one of the networks to type ether.

The exact topology configuration is shown in Example 15-4.

*Example 15-4 GLVM cluster topology*

---

```
Cluster Name: GLVM_redbook
Cluster Type: Linked
Heartbeat Type: Unicast
Repository Disks:
 Site 1 (Texas@jordan): hdisk1
 Site 2 (Maine@jessica): hdisk0
Cluster Nodes:
 Site 1 (Texas):
 jordan
 Site 2 (Maine):
 jessica

There are 2 nodes and 2 networks defined
NODE jessica:
 Network GLVM_net
 jessica 10.2.30.190
 Network net_ether_010
 jessica_xd 192.168.100.90
NODE jordan:
 Network GLVM_net
 jordan 10.2.30.191
 Network net_ether_010
 jordan_xd 192.168.100.91

cllsif -p
Adapter Type Network Net Type Attribute Node IP
Address Hardware Address Interface Name Global Name
 Netmask

jessica boot GLVM_net XD_data public jessica
10.2.30.190 en1 en1 255.255.0.0
jessica_pers persistent GLVM_net XD_data public jessica
192.168.100.190 en1 255.255.0.0
jessica_xd boot net_ether_010 XD_data public jessica
192.168.100.90 en0 255.255.255.0
jordan boot GLVM_net XD_data public jordan
10.2.30.191 en1 en1 255.255.0.0
jordan_pers persistent GLVM_net XD_data public jordan
192.168.100.191 en1 255.255.0.0
```

```

jordan_xd boot net_ether_010 XD_data public jordan
192.168.100.91 en0 255.255.255.0

clcmd netstat -in

NODE jessica

Name Mtu Network Address Ipkts Ierrs Opkts
en0 1500 link#2 96.d7.54.3b.2d.4 196703 0 838865
en0 1500 192.168.100 192.168.100.90 196703 0 838865
en1 1500 link#3 96.d7.54.3b.2d.2 912643 0 54204
en1 1500 10.2 10.2.30.190 912643 0 54204
en1 1500 192.168.0 192.168.100.190 912643 0 54204
lo0 16896 link#1
lo0 16896 127 127.0.0.1 20920 0 20920
lo0 16896 ::1%1
lo0 16896 ::1%1

```

```

NODE jordan

Name Mtu Network Address Ipkts Ierrs Opkts
en0 1500 link#2 96.d7.58.9d.39.4 197120 0 2320881
en0 1500 192.168.100 192.168.100.91 197120 0 2320881
en1 1500 link#3 96.d7.58.9d.39.2 824644 0 90585
en1 1500 10.2 10.2.30.191 824644 0 90585
en1 1500 192.168.0 192.168.100.191 824644 0 90585
lo0 16896 link#1
lo0 16896 127 127.0.0.1 26676 0 26676
lo0 16896 ::1%1
lo0 16896 ::1%1

```

### 15.4.2 Resource group

The RG is created with the resources and attributes that are shown in Example 15-5. The RG name primarily is the name of the GLVM VG with \_RG appended to it. The contents in the example were edited to show only the relevant resources.

*Example 15-5 GLVM resource group*

---

|                                                   |                                            |
|---------------------------------------------------|--------------------------------------------|
| Resource Group Name                               | GLVMvg_RG                                  |
| Participating Node Names                          | jessica jordan                             |
| Startup Policy                                    | Online on Home Node Only                   |
| Fallover Policy                                   | Fallover To Next Priority Node In The List |
| Fallback Policy                                   | Never Fallback                             |
| Site Relationship                                 | Prefer Primary Site                        |
| Node Priority                                     |                                            |
| Service IP Label                                  |                                            |
| Filesystems                                       | ALL                                        |
| Filesystems Consistency Check                     | fsck                                       |
| Filesystems Recovery Method                       | sequential                                 |
| Volume Groups                                     | GLVMvg                                     |
| Use forced varyon for volume groups, if necessary | true                                       |
| GMVG Replicated Resources                         | GLVMvg                                     |
| Application Servers                               | GLVMvg_RG_GLVM_serv                        |

---

Although all settings are important, we are most concerned with Site Relationship, Fallback Policy, and Service IP Label. For Site Relationship, the setting of Prefer Primary Site moves the RG back when the site rejoins the cluster. For Fallback Policy, the setting of Never Fallback is an intra-site policy.

The options for Site Relationship are as follows.

- ▶ Ignore

As the name implies, nothing is specified. Do not use this option with replicated resources, including GLVM configurations.

- ▶ Prefer Primary Site

On startup, only a node within the primary site activates the RG. When a site fails, the active site with the highest priority acquires the resource. When the failed site rejoins, the site with the highest priority acquires the resource.

- ▶ Either Site

On startup, a participating node in either site may activate the RG. When a site fails, the resource is acquired by the highest priority standby site. When the failed site rejoins, the resource remains with its new owner.

- ▶ Online on Both Sites

Resources are acquired by both sites. This option is for concurrent capable RGs, and does *not* apply to GLVM configurations.

For service IP addresses, it is common to define two addresses, one for each site, and add both to the RG. This approach is referred to as *site-specific service IP labels*. For more information about configuring and using this type of service IP address, see 12.4, “Site-specific service IP labels” on page 509.

### 15.4.3 Application controller

An application controller is auto-created (GLVMvg\_RG\_GLVM\_serv). The name is a combination of the RG name with GLVM\_serv appended to the end.

Example 15-6 shows an example.

*Example 15-6 GLVM application controller*

---

```
c1sserv
GLVMvg_RG_GLVM_serv
/usr/es/sbin/cluster/glvm/utils/c1_GLVMStart -a GLVMvg_RG
/usr/es/sbin/cluster/glvm/utils/c1_GLVMStop -a GLVMvg_RG
background
```

---

Both scripts are plain text ksh shell scripts that are used to export environmental variables and to enable the application monitor to gather GLVM statistics.

#### 15.4.4 Application monitor

A GLVM monitor is auto-created (GLVMvg\_RG\_GLVM\_mon). The name is a combination of the RG name with GLVM\_mon appended to the end. The primary purpose of this monitor is to gather GLVM statistics. It does so by running the **c1\_GLVMGetStatistics** utility in the background, which collects the GLVM statistics and write updates to a JavaScript Object Notation (JSON) file. This monitor always exits with success because any failure in collecting should not result in a GLVM RG failure. The JSON file is part of the file collection, as described in 15.4.5, “File collection” on page 598. It is created by using the attributes that are shown in Example 15-7.

Example 15-7 GLVM application monitor

---

Change/Show Custom Application Monitor

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

|                                    |                                                                 |
|------------------------------------|-----------------------------------------------------------------|
| [TOP]                              | [Entry Fields]                                                  |
| * Monitor Name                     | GLVMvg_RG_GLVM_mon                                              |
| New Name                           | []                                                              |
| Application Controllers to Monitor | GLVMvg_RG_GLVM_serv +                                           |
| * Monitor Mode                     | [Long-running monitoring] +                                     |
| * Monitor Method                   | [/usr/es/sbin/cluster/glvm/utils/c1_c1_GLVMMonitor -a GLVMvg_RG |
| Monitor Interval                   | [300] #                                                         |
| Hung Monitor Signal                | [9] #                                                           |
| * Stabilization Interval           | [20] #                                                          |
| Restart Count                      | [0] #                                                           |
| Restart Interval                   | [792] #                                                         |
| * Action on Application Failure    | [failover] +                                                    |
| Notify Method                      | []                                                              |
| Cleanup Method                     | [/usr/es/sbin/cluster/glvm/utils/c1_GLMStop -a GLVMvg_RG>       |
| Restart Method                     | [/usr/es/sbin/cluster/glvm/utils/c1_GLMStart -a GLVMvg_RG>      |
| Monitor Retry Count                | [0] #                                                           |
| * Enable AM logging                | No                                                              |

---

**Note:** The Action on Application Failure field is set to failover. The monitor script *should* always exit 0 and never fail. However, you might want to set it to *notify* along with a notification method to provide an extra level of insurance that a failover does *not* occur in a monitor failure.

#### 15.4.5 File collection

A GLVM file collection is auto-created (GLVMvg\_RG\_Json\_fc). The name is a combination of the RG name with Json\_fc appended to the end. This file collection consists of the contents in the /var/hacmp/log/glvm/GLVMvg\_RG directory. In our case, only a single JSON file, GLVMvg\_RG\_GLVM\_Stat.JSON, exists in that directory.

The file collection is created with the attributes that are shown in Example 15-8 on page 599. The example contains a combination of two different SMIT panels to show all in one view.

---

*Example 15-8 GLVM file collection details*

---

Add Files to a File Collection

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

|                                                          | [Entry Fields]               |
|----------------------------------------------------------|------------------------------|
| File Collection Name                                     | GLVM_GLMVg_RG_Json_fc        |
| File Collection Description                              |                              |
| Propagate files during cluster synchronization?          | yes                          |
| Propagate files automatically when changes are detected? | yes                          |
| Collection files                                         | /var/hacmp/log/glvm/GLMVg_RG |

Change/Show Automatic Update Time

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

|                                           | [Entry Fields] |
|-------------------------------------------|----------------|
| * Automatic File Update Time (in minutes) | [10]           |

---

### 15.4.6 RPV servers and clients

In our test scenario, because we have only a 2-disk (one at each site) VG, we have only a single RPV server and client that are defined at each site, as shown in Example 15-9. There is always a one-for-one ratio of the number of rpvservers and rpvclients to the number of disks to mirror.

---

*Example 15-9 GLVM RPV devices*

---

```
clcmd lsrpvserver

NODE jessica

rpvsrv0 00c472c0de143bdf hdisk1

NODE jordan

rpvsrv0 00c472c092322efd hdisk0

clcmd lsrpvclient

NODE jessica

hdisk2 00c472c092322efd Unknown

NODE jordan

hdisk2 00c472c0de143bdf
```

---

### 15.4.7 GMVG attributes

The GMVG in our test scenario is named GLVMvg. It consists of two disks, one at each site. During the creation of the GMVG, the following attributes are set from the GLVM wizard by using best practices:

|                              |                          |
|------------------------------|--------------------------|
| <b>Auto-activate:</b>        | Disabled/No              |
| <b>Bad block relocation:</b> | Disabled/Non-relocatable |
| <b>Enhanced Concurrent:</b>  | No                       |
| <b>Mirror Pools:</b>         | Superstrict              |
| <b>Scalable:</b>             | Yes                      |
| <b>Quorum:</b>               | Disabled/No              |

All the GMVG attributes are shown in Example 15-10.

*Example 15-10 GMVG (GLVMvg) attributes*

|                                   |                                                  |
|-----------------------------------|--------------------------------------------------|
| VOLUME GROUP: GLVMvg              | VG IDENTIFIER: 000c472c000004b00000001858b9bc6b7 |
| VG STATE: active                  | PP SIZE: 8 megabytes                             |
| VG PERMISSION: read/write         | TOTAL PPs: 2538 (20304 MBytes)                   |
| MAX LVs: 256                      | FREE PPs: 2266 (18128 MBytes)                    |
| LVs: 3                            | USED PPs: 272 (2176 megabytes)                   |
| OPEN LVs: 1                       | QUORUM: 1 (Disabled)                             |
| TOTAL PVs: 2                      | VG DESCRIPTORS: 4                                |
| STALE PVs: 0                      | STALE PPs: 0                                     |
| ACTIVE PVs: 0                     | AUTO ON: no                                      |
| MAX PPs per VG: 32768             | MAX PVs: 1024                                    |
| LTG size (Dynamic): 128 kilobytes | AUTO SYNC: no                                    |
| HOT SPARE: no                     | BB POLICY: non-relocatable                       |
| <b>MIRROR POOL STRICT:</b> super  |                                                  |
| PV RESTRICTION: none              | INFINITE RETRY: no                               |
| DISK BLOCK SIZE: 512              | CRITICAL VG: yes                                 |
| FS SYNC OPTION: no                | CRITICAL PVs: no                                 |
| ENCRYPTION: yes                   |                                                  |

### 15.4.8 Mirror pools

During GMVG creation, the GLVM wizard also creates uniquely named mirror pools. In our case, there are two mirror pools because there are only two mirrored copies. There is a limit of three mirror pools. The GLVM wizard automatically sets the mirroring mode to type `async`, which is a stated limitation in using the wizard.

The mirror pools' names are initially created with `MP` and `MP#` appended to the end of the VG name. The wizard also sets the `async` cache high water mark to 80.

The high water mark is the percent of I/O cache size that can be used before new write requests must wait for mirroring to catch up. The default is 100. All the mirror pool detailed attributes are shown in Example 15-11 on page 601.

*Example 15-11 GLVM async mirror pools attributes and member disks*

---

```
lsmmp -A GLVMvg
VOLUME GROUP: GLVMvg Mirror Pool Super Strict: yes

MIRROR POOL: GLVMvgMP Mirroring Mode: ASYNC
ASYNC MIRROR STATE: inactive ASYNC CACHE LV: GLVMvgALV1
ASYNC CACHE VALID: no ASYNC CACHE EMPTY: yes
ASYNC CACHE HWM: 80 ASYNC DATA DIVERGED: no

MIRROR POOL: GLVMvgMP1 Mirroring Mode: ASYNC
ASYNC MIRROR STATE: inactive ASYNC CACHE LV: GLVMvgALV
ASYNC CACHE VALID: yes ASYNC CACHE EMPTY: yes
ASYNC CACHE HWM: 80 ASYNC DATA DIVERGED: no

lspv -P|grep -i glvm
hdisk0 GLVMvg
hdisk2 GLVMvgMP
hdisk2 GLVMvgMP1
```

---

### 15.4.9 Logical volumes mirrored

During GMVG creation, the GLVM wizard ensures that every non-AIO cache LV is mirrored across the appropriate mirror pools or disks. It also sets the best practices attributes:

**Bad block relocation:** Disabled/Non-relocatable  
**Strictness:** Superstrict  
**Upper Bound:** Lowest common denominator allowed (in our case 1)  
**Mirror Write Consistency:** Passive

All the LV attributes are shown in Example 15-12.

*Example 15-12 GLVM mirrored logical volume attributes*

---

```
lslv fs1v00
LOGICAL VOLUME: fs1v00 VOLUME GROUP: GLVMvg
LV IDENTIFIER: 00c472c000004b00000001858b9bc6b7.1 PERMISSION: read/write
VG STATE: active/complete LV STATE: opened/syncd
TYPE: jfs2 WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
COPIES: 2 SCHED POLICY: parallel
LPs: 128 PPs: 256
STALE PPs: 0 BB POLICY: non-relocatable
INTER-POLICY: minimum RELOCATABLE: yes
INTRA-POLICY: middle UPPER BOUND: 1
MOUNT POINT: /GLVMfs LABEL: /GLVMfs
DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/PASSIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?: NO
INFINITE RETRY: no PREFERRED READ: 0
DEVICE SUBTYPE: DS_LVZ
COPY 1 MIRROR POOL: GLVMvgMP
COPY 2 MIRROR POOL: GLVMvgMP1
COPY 3 MIRROR POOL: None
ENCRYPTION: no
```

---

### 15.4.10 AIO cache logical volumes

During GMVG creation, the GLVM wizard also creates uniquely named async I/O LVs. There are two (one for each site). They are *not* mirrored. The LVs have some attributes that are similar to LVs that are mirrored:

|                                 |                                                   |
|---------------------------------|---------------------------------------------------|
| <b>Bad block relocation:</b>    | Disabled/Non-relocatable                          |
| <b>Strictness:</b>              | Superstrict                                       |
| <b>Upper Bound:</b>             | Lowest common denominator allowed (in our case 1) |
| <b>Mirror Write Consistency</b> | Passive                                           |

The AIO LV names are initially created with ALV and ALV1 appended to the end of the LV name. The names and attributes are shown in Example 15-13.

*Example 15-13 AIO logical volume names and attributes*

```
lsvg -l GLVMvg
GLVMvg:
LV NAME TYPE LPs PPs PVs LV STATE MOUNT POINT
fs1v00 jfs2 128 256 2 open/syncd /GLVMfs
GLVMvgALV aio_cache 8 8 1 closed/syncd N/A
GLVMvgALV1 aio_cache 8 8 1 closed/syncd N/A

ls1v GLVMvgALV
LOGICAL VOLUME: GLVMvgALV VOLUME GROUP: GLVMvg
LV IDENTIFIER: 00c472c000004b00000001858b9bc6b7.2 PERMISSION: read/write
VG STATE: active/complete LV STATE: closed/syncd
TYPE: aio_cache WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
COPIES: 1 SCHED POLICY: parallel
LPs: 8 PPs: 8
STALE PPs: 0 BB POLICY: non-relocatable
INTER-POLICY: minimum RELOCATABLE: yes
INTRA-POLICY: middle UPPER BOUND: 1
MOUNT POINT: N/A LABEL: None
DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/PASSIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?: NO
INFINITE RETRY: no PREFERRED READ: 0
DEVICE SUBTYPE: DS_LVZ
COPY 1 MIRROR POOL: GLVMvgMP
COPY 2 MIRROR POOL: None
COPY 3 MIRROR POOL: None
ENCRYPTION: no

ls1v GLVMvgALV1
LOGICAL VOLUME: GLVMvgALV1 VOLUME GROUP: GLVMvg
LV IDENTIFIER: 00c472c000004b00000001858b9bc6b7.3 PERMISSION: read/write
VG STATE: active/complete LV STATE: closed/syncd
TYPE: aio_cache WRITE VERIFY: off
MAX LPs: 512 PP SIZE: 8 megabytes
COPIES: 1 SCHED POLICY: parallel
LPs: 8 PPs: 8
STALE PPs: 0 BB POLICY: non-relocatable
INTER-POLICY: minimum RELOCATABLE: yes
INTRA-POLICY: middle UPPER BOUND: 1
MOUNT POINT: N/A LABEL: None
```

```

DEVICE UID: 0 DEVICE GID: 0
DEVICE PERMISSIONS: 432
MIRROR WRITE CONSISTENCY: on/PASSIVE
EACH LPAR COPY ON A SEPARATE PV ?: yes (superstrict)
Serialize IO ?: NO
INFINITE RETRY: no
DEVICESUBTYPE: DS_LVZ
COPY 1 MIRROR POOL: GLVMvgMP1
COPY 2 MIRROR POOL: None
COPY 3 MIRROR POOL: None
ENCRYPTION: no

```

---

### 15.4.11 Split and merge policy

The GLVM wizard leaves the split and merge policy as the defaults none and majority, as shown in Example 15-14.

*Example 15-14 Split and merge settings*

---

```
clmgr query cluster|egrep "SPLIT|MERGE"
SPLIT_POLICY="none"
MERGE_POLICY="majority"
```

---

If these settings do not comply with your cluster configuration, you can change these policies and their possible values. For more information, see Table 2-1 on page 37.

## 15.5 Using the command-line interface

As stated in 15.1, “Introduction” on page 576, the GLVM wizard is primarily based on the **c1\_glvmp Configuration** command, which both the SMUI and SMIT run with the flags that are based on the responses to the prompts. When you use the SMUI, the VG must already exist. SMIT wants to create the VG.

If you want to be specific about what options to use, consider using the CLI. Example 15-15 shows the **c1\_glvmp Configuration** command reference.

*Example 15-15 GLVM CLI command reference*

Usage:

```
c1_glvmp Configuration [-v <VolumeGroup> [-l <LocalSitePvidList> -r
<RemoteSitePvidList>]]
[-g <ExistingResourceGroup>]
[-s <cache_size> -u <cache_size_unit>]
[-o compression=<0|1> | io_grp_latency=## |
no_parallel_lps=<2|4|8|16|32>]
[[-A | -R] <VolumeGroup> -D <PVname...>]
[-p <ExistingVolumeGroup> -n RemoteSiteNodes]
[-d <VolumeGroup>] [-j] [-h]
```

Options:

- A Volume group name, existing GLVM volume group to which new disks are adding.

- c Adds GLVM statistics application monitor and application controller to the provided resource group.

- d Unconfigure the volume group.

```

-D hdisk1,hdisk2.
-g Resource Group name, it can be a new resource group or an existing
resource group to convert into a GLVM resource group.
-h Help
-j Display GLVM Configuration details in JSON format.
-l Local site disk PVIDs to create rpvservers and rpvclients for new volume
group.
-n Remotesitenode1,Remotesitenode2.
-o Update the glvm tunables.
 compression to enable or disable
 io_grp_latency in milliseconds
 no_parallel_lps can be given in 2, 4, 8, 16, or 32.
-p Designed to call from GUI GLVM only.
-r Remote site disk PVIDs to create rpvservers and rpvclients for new volume
group.
-R Volume group name, existing GLVM volume group to which new disks are
removing.
-s Asynchronous Cache size.
-u To specify units(GB G, MB M, KB K) for cache size, if unit is not
specified then cache size would become cache_size*PP size.
-v Volume group name, it can be a new volume group or an existing volume
group to convert into a GLVM volume group.

Example:
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4)
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -o
compression=1
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -o
io_grp_latency=5
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -o
compression=1 -o no_parallel_lps=2
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -s 3 -u G
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -s 3 -u G -o
compression=1
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -s 3 -u G -o
no_parallel_lps=2
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -s 3 -u G -o
io_grp_latency=5
 cl_glvm_configuration -v vg1 -l (pvid1,pvid2) -r (pvid3,pvid4) -s 3 -u G -o
compression=1 -o io_grp_latency=5 -o no_parallel_lps=2
 cl_glvm_configuration -v vg1
 cl_glvm_configuration -v vg1 -o compression=1
 cl_glvm_configuration -v vg1 -o no_parallel_lps=2
 cl_glvm_configuration -v vg1 -o compression=1 -o no_parallel_lps=2
 cl_glvm_configuration -v vg1 -s 10
 cl_glvm_configuration -v vg1 -s 3 -u G
 cl_glvm_configuration -v vg1 -s 3072 -u M
 cl_glvm_configuration -v vg1 -s 3072 -u M -o compression=1
 cl_glvm_configuration -v vg1 -s 3072 -u M -o io_grp_latency=5
 cl_glvm_configuration -v vg1 -s 3072 -u M -o no_parallel_lps=2
 cl_glvm_configuration -v vg1 -s 3072 -u M -o compression=1 -o io_grp_latency=5
-o no_parallel_lps=2
 cl_glvm_configuration -g rg1
 cl_glvm_configuration -g rg1 -o compression=1
 cl_glvm_configuration -g rg1 -o no_parallel_lps=2
 cl_glvm_configuration -g rg1 -o compression=1 -o no_parallel_lps=2

```

```
cl_glvm_configuration -g rg1 -s 10
cl_glvm_configuration -g rg1 -s 3 -u G
cl_glvm_configuration -g rg1 -s 3072 -u M
cl_glvm_configuration -g rg1 -s 3072 -u M -o compression=1
cl_glvm_configuration -g rg1 -s 3072 -u M -o io_grp_latency=5
cl_glvm_configuration -g rg1 -s 3072 -u M -o no_parallel_lps=2
cl_glvm_configuration -g rg1 -s 3072 -u M -o compression=1 -o io_grp_latency=5
-o no_parallel_lps=2
cl_glvm_configuration -p vg1 -n node3,node4
cl_glvm_configuration -d vg1
cl_glvm_configuration -A vg1 -D hdisk1,hdisk2
cl_glvm_configuration -R vg1 -D hdisk1,hdisk2
cl_glvm_configuration -c rg1
cl_glvm_configuration -j
```

---





## Part 5

# Appendices

This part includes the following appendixes:

- ▶ Appendix A, “Paper planning worksheets” on page 609
- ▶ Appendix B, “Cluster Test Tool log” on page 621





A

# Paper planning worksheets

The following planning worksheets can be used to guide you through the planning and implementation of a PowerHA cluster. The worksheets cover all the important aspects of the cluster configuration and follow a logical planning flow.

For more information about planning, see [Planning PowerHA SystemMirror](#).

# Internet Protocol network planning worksheets

To record your network information, use the worksheet in Table A-1.

*Table A-1 Cluster Ethernet networks*

# Internet Protocol network interface worksheet

To record your inference information, use the worksheet in Table A-2.

*Table A-2 Network interface worksheet*

Table A-3 shows the cluster repository disk worksheet.

*Table A-3 Cluster repository disk*

# Fibre Channel disks worksheets

To record information about the Fibre Channel (FC) disks to include in the cluster, use the worksheet in Table A-4. Complete a separate worksheet for each cluster node.

*Table A-4 Fibre Channel disks worksheet*

## Shared volume group and file system worksheet

To record the shared volume groups (VGs) and file systems in a non-concurrent access configuration, use the worksheet in Table A-5. Use a separate worksheet for each shared VG, print a worksheet for each VG, and complete the names of the nodes that share the VG on each worksheet.

*Table A-5 Shared volume group and file system worksheet*

|                                        | Node A | Node B |
|----------------------------------------|--------|--------|
| Node Names                             |        |        |
| Shared Volume Group Name               |        |        |
| Major Number                           |        |        |
| Log Logical Volume Name                |        |        |
| Physical Volumes                       |        |        |
|                                        |        |        |
|                                        |        |        |
|                                        |        |        |
|                                        |        |        |
|                                        |        |        |
|                                        |        |        |
| Cross-site LVM Mirror                  |        |        |
| Logical Volume Name                    |        |        |
| Number Of Copies Of Logical Partitions |        |        |
| On Separate Physical Volumes           |        |        |
| File System Mount Point                |        |        |
| Size                                   |        |        |
| Cross-site LVM Mirroring Enabled       |        |        |

## NFS-exported file system or directory worksheet

To record the file systems and directories that are Network File System (NFS)-exported by a node in a non-concurrent access configuration, use the worksheet in Table A-6. Use a separate worksheet for each node that is defined in the cluster, print a worksheet for each node, and specify a node name on each worksheet.

*Table A-6 NFS-export file system or directory worksheet*

| Resource group name                          |                                  |
|----------------------------------------------|----------------------------------|
| Network for NFS mount                        |                                  |
| File System Mounted before IP Configured?    |                                  |
| For export options                           | See the <b>exports</b> man page. |
| File System or Directory to Export (NFSv2/3) |                                  |
| Export options                               |                                  |
| File System or Directory to Export (NFSv4)   |                                  |
| Export Options                               |                                  |
| File System or Directory to Export (NFSv2/3) |                                  |
| Export Options                               |                                  |
| File System or Directory to Export (NFSv4)   |                                  |
| Export Options                               |                                  |
| Stable Storage Path (NFSv4)                  |                                  |

## Application worksheet

To record information about applications in the cluster, use the worksheet in Table A-7.

*Table A-7 Application worksheet*

| <b>Application name</b>                         |  |
|-------------------------------------------------|--|
| Directory                                       |  |
| Executable Files                                |  |
| Configuration Files                             |  |
| Data files or devices                           |  |
| Log files or devices                            |  |
| Cluster Name                                    |  |
| <b>Failover strategy (P=Primary T=Takeover)</b> |  |
| Node                                            |  |
| Strategy                                        |  |
| <b>Normal Start Commands</b>                    |  |
|                                                 |  |
| <b>Normal stop Commands</b>                     |  |
|                                                 |  |
| <b>Verification Commands</b>                    |  |
|                                                 |  |
| <b>Node Reintegration Caveats</b>               |  |
| Node A                                          |  |
| Node B                                          |  |

## Application server worksheet

To record information about application servers in the cluster, use the worksheet in Table A-8.

*Table A-8 Application server worksheet*

| Cluster name                                            |  |
|---------------------------------------------------------|--|
| Note: Use full path names for all user-defined scripts. |  |
| Server name                                             |  |
| Start Script                                            |  |
| Stop Script                                             |  |
| Server name                                             |  |
| Start Script                                            |  |
| Stop Script                                             |  |
| Server name                                             |  |
| Start Script                                            |  |
| Stop Script                                             |  |

## Application monitor worksheet (custom)

To record information about custom application monitors in the cluster, use the worksheet in Table A-9.

*Table A-9 Application server worksheet*

| Cluster name                         |  |
|--------------------------------------|--|
| Application server name              |  |
| Monitor Method                       |  |
| Monitor Interval                     |  |
| <b>Hung Monitor Signal</b>           |  |
| Stabilization Interval               |  |
| Restart Count                        |  |
| Restart Interval                     |  |
| <b>Action on Application Failure</b> |  |
| Notify Method                        |  |
| Cleanup Method                       |  |
| Restart Method                       |  |

## Resource group worksheet

To record information about resource groups (RGs) in the cluster, use the worksheet in Table A-10.

Table A-10 Resource groups

| Cluster name                                       |  |
|----------------------------------------------------|--|
| RESOURCE Group Name                                |  |
| Participating Node Names                           |  |
| Inter-Site Management Policy                       |  |
| Startup Policy                                     |  |
| Fallover Policy                                    |  |
| Fallback Policy                                    |  |
| Delayed Fallback Timer                             |  |
| Settling Time                                      |  |
| Runtime Policies                                   |  |
| Dynamic Node Priority Policy                       |  |
| Processing Order (Parallel, Serial, or Customized) |  |
| Service IP Label                                   |  |
| File systems                                       |  |
| File system Consistency Check                      |  |
| File systems Recovery Method                       |  |
| File systems/Directories to Export                 |  |
| File systems/Directories to NFS mount (NFSv2/3)    |  |
| File systems/Directories to NFS mount (NFSv4)      |  |
| Storage Stable Path (NFSv4)                        |  |
| Network for NFS mount                              |  |
| Volume Groups                                      |  |
| Concurrent Volume Groups                           |  |
| Use forced varyon for volume groups, if necessary  |  |
| Raw Disk PVIDS                                     |  |
| Disk Error Management                              |  |
| Tape Resources                                     |  |
| Application Servers                                |  |
| Primary WLM Class                                  |  |

| Cluster name                              |  |
|-------------------------------------------|--|
| Secondary WLM Class                       |  |
| Miscellaneous Data                        |  |
| Delayed Fallback Timer                    |  |
| Auto Import Volume Groups                 |  |
| file systems Mounted before IP Configured |  |
| User-Defined Resources                    |  |
| WPAR Name                                 |  |
| COMMENTS                                  |  |

## Cluster events worksheet

To record information about cluster events in the cluster, use the worksheet in Table A-11.

*Table A-11 Cluster event worksheet*

| Cluster name                         |  |
|--------------------------------------|--|
| Cluster Event Description            |  |
| Cluster Event Method                 |  |
| Cluster Event Name                   |  |
| Event Command                        |  |
| Notify Command                       |  |
| Remote Notification Message Text     |  |
| Remote Notification Message Location |  |
| Pre-Event Command                    |  |
| Post-Event Command                   |  |
| Event Recovery Command               |  |
| Recovery Counter                     |  |
| Time Until Warning                   |  |

## Cluster file collections worksheet

To record information about file collections in the cluster, use the worksheet in Table A-12.

*Table A-12 Cluster file collections worksheet*

| Cluster name                        |  |
|-------------------------------------|--|
| File Collection name                |  |
| File Collection description         |  |
| Propagate files before verification |  |
| Propagate files automatically       |  |
| Files to include in this collection |  |
| Automatic check time limit          |  |





B

# Cluster Test Tool log

This appendix contains the output of the test tool log (`var/hacmp/log/cl_testtool.log`).

## Sample output from the Cluster Test Tool log

Example B-1 shows the output from the test tool log (var/hacmp/log/cl\_testtool.log). This output is from a PowerHA 7.2.7 cluster.

*Example B-1 Output from the test tool log of var/hacmp/log/cl\_testtool.log*

---

```
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Initializing Variable Table
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Using Process Environment for Variable Table
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Reading Static Configuration Data
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Cluster Name: jessica_cluster
26/11/2022_21:07:14: Cluster Version: 23
26/11/2022_21:07:14: Local Node Name: jessica
26/11/2022_21:07:14: Cluster Nodes: jessica jordan
26/11/2022_21:07:14: Found 1 Cluster Networks
26/11/2022_21:07:14: Found 4 Cluster Interfaces/Device/Labels
26/11/2022_21:07:14: Found 1 Cluster Resource Groups
26/11/2022_21:07:14: Found 10 Cluster Resources
26/11/2022_21:07:14: Event Timeout Value: 720
26/11/2022_21:07:14: Maximum Timeout Value: 2880
26/11/2022_21:07:14: Found 2 Cluster Sites
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Building Test Queue
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Test Plan: /usr/es/sbin/cluster/cl_testtool/auto_topology
26/11/2022_21:07:14: Event 1: NODE_UP: NODE_UP,ALL,Start cluster services on all available nodes
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Validate NODE_UP
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: ALL
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14: Event 2: NODE_DOWN_GRACEFUL: NODE_DOWN_GRACEFUL,node1,Stop cluster services gracefully on a node
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Validate NODE_DOWN_GRACEFUL
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: jessica
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14: Event 3: NODE_UP: NODE_UP,node1,Restart cluster services on the node that was stopped
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Validate NODE_UP
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: jessica
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14: Event 4: NODE_DOWN_TAKEOVER: NODE_DOWN_TAKEOVER,node2,Stop cluster services with takeover on a node
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Validate NODE_DOWN_TAKEOVER
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: jordan
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14: Event 5: NODE_UP: NODE_UP,node2,Restart cluster services on the node that was stopped
26/11/2022_21:07:14: -----
```

```

26/11/2022_21:07:14: | Validate NODE_UP
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: jordan
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14: Event 6: NODE_DOWN_FORCED: NODE_DOWN_FORCED,node3,Stop cluster
services forced on a node
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Validate NODE_DOWN_FORCED
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: jessica
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14: Event 7: NODE_UP: NODE_UP,node3,Restart cluster services on the node
that was stopped
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: | Validate NODE_UP
26/11/2022_21:07:14: -----
26/11/2022_21:07:14: Event node: jessica
26/11/2022_21:07:14: Configured nodes: jessica jordan
26/11/2022_21:07:14:
#####
26/11/2022_21:07:14: ##
Starting Cluster Test Tool: -e /usr/es/sbin/cluster/cl_testtool/auto_topology
##
26/11/2022_21:07:14:
#####
26/11/2022_21:07:14: ||
|| Starting Test 1 - NODE_UP,ALL,Start cluster services on all available nodes
||
26/11/2022_21:07:14:
=====
26/11/2022_21:07:15: -----
26/11/2022_21:07:15: | is_rational NODE_UP
26/11/2022_21:07:15: -----
26/11/2022_21:07:15: Checking cluster stability
26/11/2022_21:07:15: jessica: ST_INIT
26/11/2022_21:07:15: jordan: ST_INIT
26/11/2022_21:07:15: Cluster is stable
26/11/2022_21:07:15: Active Nodes:
26/11/2022_21:07:15: -----
26/11/2022_21:07:15: | Executing Command for NODE_UP
26/11/2022_21:07:15: -----
26/11/2022_21:07:15: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e NODE_UP -m
execute 'jessica,jordan'
26/11/2022_21:07:42: -----
26/11/2022_21:07:42: | Entering wait_for_stable
26/11/2022_21:07:42: -----
26/11/2022_21:07:42: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:08:12: Checking Node States:
26/11/2022_21:08:12: Node jessica: ST_STABLE
26/11/2022_21:08:12: Active Timers: None
26/11/2022_21:08:12: Node jordan: ST_STABLE
26/11/2022_21:08:12: Active Timers: None
26/11/2022_21:08:12: -----
26/11/2022_21:08:12: | NODE_UP: Checking post-event status
26/11/2022_21:08:12: -----
26/11/2022_21:08:12: Event Nodes: jessica jordan
26/11/2022_21:08:13: pre-event online nodes:
26/11/2022_21:08:13: post-event online nodes: jessica jordan

```

```

26/11/2022_21:08:13: Checking node states
26/11/2022_21:08:13: jessica: Preevent state: ST_INIT, Postevent state: ST_STABLE
26/11/2022_21:08:13: jordan: Preevent state: ST_INIT, Postevent state: ST_STABLE
26/11/2022_21:08:13: Checking RG states
26/11/2022_21:08:13: Resource Group: redbookrg
26/11/2022_21:08:13: Node: jessica Pre Event State: OFFLINE, Post Event State: ONLINE
26/11/2022_21:08:13: Node: jordan Pre Event State: OFFLINE, Post Event State: OFFLINE
26/11/2022_21:08:13: Checking event history
26/11/2022_21:08:13: Begin Event History records:
26/11/2022_21:08:13: NODE: jessica
 Nov 26 2022 21:07:38 EVENT COMPLETED: admin_op clrm_start_request 12922 0 0
 <LAT>|2022-11-26T21:07:38|12922|EVENT COMPLETED: admin_op clrm_start_request
12922 0 0|</LAT>
 Nov 26 2022 21:07:43 EVENT COMPLETED: site_up_local Dallas 0
 <LAT>|2022-11-26T21:07:43|12923|EVENT COMPLETED: site_up_local Dallas 0|</LAT>
 Nov 26 2022 21:07:43 EVENT COMPLETED: site_up Dallas 0
 <LAT>|2022-11-26T21:07:43|12923|EVENT COMPLETED: site_up Dallas 0|</LAT>
 Nov 26 2022 21:07:45 EVENT COMPLETED: node_up jessica 0
 <LAT>|2022-11-26T21:07:45|12923|EVENT COMPLETED: node_up jessica 0|</LAT>
 Nov 26 2022 21:07:47 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:07:47|12924|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:07:48 EVENT COMPLETED: acquire_service_addr 0
 <LAT>|2022-11-26T21:07:49|12924|EVENT COMPLETED: acquire_service_addr 0|</LAT>
 Nov 26 2022 21:07:50 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
 <LAT>|2022-11-26T21:07:50|12924|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
 Nov 26 2022 21:07:50 EVENT COMPLETED: rg_move_acquire jessica 1 0
 <LAT>|2022-11-26T21:07:50|12924|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
 Nov 26 2022 21:07:52 EVENT COMPLETED: start_server demoapp 0
 <LAT>|2022-11-26T21:07:52|12924|EVENT COMPLETED: start_server demoapp 0|</LAT>
 Nov 26 2022 21:07:52 EVENT COMPLETED: rg_move_complete jessica 1 0
 <LAT>|2022-11-26T21:07:52|12924|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
 Nov 26 2022 21:07:54 EVENT COMPLETED: node_up_complete jessica 0
 <LAT>|2022-11-26T21:07:54|12925|EVENT COMPLETED: node_up_complete jessica
0|</LAT>
 Nov 26 2022 21:07:54 EVENT COMPLETED: site_up_local_complete Dallas 0
 <LAT>|2022-11-26T21:07:55|12925|EVENT COMPLETED: site_up_local_complete Dallas
0|</LAT>
 Nov 26 2022 21:07:55 EVENT COMPLETED: site_up_complete Dallas 0
 <LAT>|2022-11-26T21:07:55|12925|EVENT COMPLETED: site_up_complete Dallas 0|</LAT>
 Nov 26 2022 21:08:00 EVENT COMPLETED: site_up_remote FortWorth 0
 <LAT>|2022-11-26T21:08:00|7835|EVENT COMPLETED: site_up_remote FortWorth 0|</LAT>
 Nov 26 2022 21:08:00 EVENT COMPLETED: site_up FortWorth 0
 <LAT>|2022-11-26T21:08:00|7835|EVENT COMPLETED: site_up FortWorth 0|</LAT>
 Nov 26 2022 21:08:02 EVENT COMPLETED: node_up jordan 0
 <LAT>|2022-11-26T21:08:02|7835|EVENT COMPLETED: node_up jordan 0|</LAT>
 Nov 26 2022 21:08:05 EVENT COMPLETED: node_up_complete jordan 0
 <LAT>|2022-11-26T21:08:05|7836|EVENT COMPLETED: node_up_complete jordan 0|</LAT>
 Nov 26 2022 21:08:05 EVENT COMPLETED: site_up_remote_complete FortWorth 0
 <LAT>|2022-11-26T21:08:05|7836|EVENT COMPLETED: site_up_remote_complete FortWorth
0|</LAT>
 Nov 26 2022 21:08:05 EVENT COMPLETED: site_up_complete FortWorth 0
 <LAT>|2022-11-26T21:08:05|7836|EVENT COMPLETED: site_up_complete FortWorth
0|</LAT>
26/11/2022_21:08:13: NODE: jordan
 Nov 26 2022 21:07:58 EVENT COMPLETED: admin_op clrm_start_request 7834 0 0
 <LAT>|2022-11-26T21:07:58|7834|EVENT COMPLETED: admin_op clrm_start_request 7834
0 0|</LAT>

```

```

Nov 26 2022 21:08:16 EVENT COMPLETED: site_up_local FortWorth 0
<LAT>|2022-11-26T21:08:16|7835|EVENT COMPLETED: site_up_local FortWorth 0|</LAT>
Nov 26 2022 21:08:16 EVENT COMPLETED: site_up FortWorth 0
<LAT>|2022-11-26T21:08:16|7835|EVENT COMPLETED: site_up FortWorth 0|</LAT>
Nov 26 2022 21:08:17 EVENT COMPLETED: node_up jordan 0
<LAT>|2022-11-26T21:08:17|7835|EVENT COMPLETED: node_up jordan 0|</LAT>
Nov 26 2022 21:08:19 EVENT COMPLETED: node_up_complete jordan 0
<LAT>|2022-11-26T21:08:20|7836|EVENT COMPLETED: node_up_complete jordan 0|</LAT>
Nov 26 2022 21:08:20 EVENT COMPLETED: site_up_local_complete FortWorth 0
<LAT>|2022-11-26T21:08:20|7836|EVENT COMPLETED: site_up_local_complete FortWorth
0|</LAT>
Nov 26 2022 21:08:20 EVENT COMPLETED: site_up_complete FortWorth 0
<LAT>|2022-11-26T21:08:20|7836|EVENT COMPLETED: site_up_complete FortWorth
0|</LAT>
26/11/2022_21:08:13: End Event History records
26/11/2022_21:08:13:
=====
26/11/2022_21:08:13: ||
|| Test 1 Complete - NODE_UP: Start cluster services on all available nodes
||
26/11/2022_21:08:13: || Test Completion Status: PASSED
||
26/11/2022_21:08:13:
=====
26/11/2022_21:08:13:
=====
26/11/2022_21:08:13: ||
|| Starting Test 2 - NODE_DOWN_GRACEFUL,jessica,Stop cluster services gracefully on a node
||
26/11/2022_21:08:13:
=====
26/11/2022_21:08:13: -----
26/11/2022_21:08:13: | is_rational NODE_DOWN_GRACEFUL
26/11/2022_21:08:13: -----
26/11/2022_21:08:13: Checking cluster stability
26/11/2022_21:08:13: jessica: ST_STABLE
26/11/2022_21:08:13: jordan: ST_STABLE
26/11/2022_21:08:13: Cluster is stable
26/11/2022_21:08:13: Node: jessica, State: ST_STABLE
26/11/2022_21:08:13: -----
26/11/2022_21:08:13: | Executing Command for NODE_DOWN_GRACEFUL
26/11/2022_21:08:13: -----
26/11/2022_21:08:13: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e
NODE_DOWN_GRACEFUL -m execute 'jessica'
26/11/2022_21:08:15: -----
26/11/2022_21:08:15: | Entering wait_for_stable
26/11/2022_21:08:15: -----
26/11/2022_21:08:15: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:08:46: Checking Node States:
26/11/2022_21:08:46: Node jessica: ST_INIT
26/11/2022_21:08:46: Node jordan: ST_STABLE
26/11/2022_21:08:46: Active Timers: None
26/11/2022_21:08:46: -----
26/11/2022_21:08:46: | NODE_DOWN_GRACEFUL: Checking post-event status
26/11/2022_21:08:46: -----
26/11/2022_21:08:46: Event Nodes: jessica
26/11/2022_21:08:46: pre-event online nodes: jessica jordan
26/11/2022_21:08:46: post-event online nodes: jordan
26/11/2022_21:08:46: Checking node states
26/11/2022_21:08:46: jessica: Preevent state: ST_STABLE, Postevent state: ST_INIT

```

```

26/11/2022_21:08:46: jordan: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:08:46: Checking RG states
26/11/2022_21:08:46: Resource Group: redbookrg
26/11/2022_21:08:46: Node: jessica Pre Event State: ONLINE, Post Event State: OFFLINE
26/11/2022_21:08:46: Node: jordan Pre Event State: OFFLINE, Post Event State: OFFLINE
26/11/2022_21:08:46: Checking event history
26/11/2022_21:08:46: Begin Event History records:
26/11/2022_21:08:46: NODE: jessica
 Nov 26 2022 21:08:15 EVENT COMPLETED: admin_op clrm_stop_request 7837 0 0
 <LAT>|2022-11-26T21:08:15|7837|EVENT COMPLETED: admin_op clrm_stop_request 7837 0
0|</LAT>
 Nov 26 2022 21:08:16 EVENT COMPLETED: site_down_local 0
 <LAT>|2022-11-26T21:08:16|7837|EVENT COMPLETED: site_down_local 0|</LAT>
 Nov 26 2022 21:08:16 EVENT COMPLETED: site_down Dallas 0
 <LAT>|2022-11-26T21:08:16|7837|EVENT COMPLETED: site_down Dallas 0|</LAT>
 Nov 26 2022 21:08:16 EVENT COMPLETED: node_down jessica graceful 0
 <LAT>|2022-11-26T21:08:16|7837|EVENT COMPLETED: node_down jessica graceful
0|</LAT>
 Nov 26 2022 21:08:19 EVENT COMPLETED: stop_server demoapp 0
 <LAT>|2022-11-26T21:08:19|7838|EVENT COMPLETED: stop_server demoapp 0|</LAT>
 Nov 26 2022 21:08:22 EVENT COMPLETED: release_service_addr 0
 <LAT>|2022-11-26T21:08:22|7838|EVENT COMPLETED: release_service_addr 0|</LAT>
 Nov 26 2022 21:08:22 EVENT COMPLETED: rg_move jessica 1 RELEASE 0
 <LAT>|2022-11-26T21:08:22|7838|EVENT COMPLETED: rg_move jessica 1 RELEASE
0|</LAT>
 Nov 26 2022 21:08:22 EVENT COMPLETED: rg_move_release jessica 1 0
 <LAT>|2022-11-26T21:08:22|7838|EVENT COMPLETED: rg_move_release jessica 1
0|</LAT>
 Nov 26 2022 21:08:25 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:08:25|7838|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:08:28 EVENT COMPLETED: node_down_complete jessica 0
 <LAT>|2022-11-26T21:08:28|7839|EVENT COMPLETED: node_down_complete jessica
0|</LAT>
 Nov 26 2022 21:08:28 EVENT COMPLETED: site_down_local_complete 0
 <LAT>|2022-11-26T21:08:28|7839|EVENT COMPLETED: site_down_local_complete 0|</LAT>
 Nov 26 2022 21:08:28 EVENT COMPLETED: site_down_complete Dallas 0
 <LAT>|2022-11-26T21:08:28|7839|EVENT COMPLETED: site_down_complete Dallas
0|</LAT>
26/11/2022_21:08:46: NODE: jordan
 Nov 26 2022 21:08:30 EVENT COMPLETED: site_down_remote Dallas 0
 <LAT>|2022-11-26T21:08:30|7837|EVENT COMPLETED: site_down_remote Dallas 0|</LAT>
 Nov 26 2022 21:08:31 EVENT COMPLETED: site_down Dallas 0
 <LAT>|2022-11-26T21:08:31|7837|EVENT COMPLETED: site_down Dallas 0|</LAT>
 Nov 26 2022 21:08:31 EVENT COMPLETED: node_down jessica graceful 0
 <LAT>|2022-11-26T21:08:31|7837|EVENT COMPLETED: node_down jessica graceful
0|</LAT>
 Nov 26 2022 21:08:33 EVENT COMPLETED: rg_move jessica 1 RELEASE 0
 <LAT>|2022-11-26T21:08:34|7838|EVENT COMPLETED: rg_move jessica 1 RELEASE
0|</LAT>
 Nov 26 2022 21:08:34 EVENT COMPLETED: rg_move_release jessica 1 0
 <LAT>|2022-11-26T21:08:34|7838|EVENT COMPLETED: rg_move_release jessica 1
0|</LAT>
 Nov 26 2022 21:08:39 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:08:39|7838|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:08:42 EVENT COMPLETED: node_down_complete jessica 0
 <LAT>|2022-11-26T21:08:42|7839|EVENT COMPLETED: node_down_complete jessica
0|</LAT>
 Nov 26 2022 21:08:43 EVENT COMPLETED: site_down_remote_complete Dallas 0
 <LAT>|2022-11-26T21:08:43|7839|EVENT COMPLETED: site_down_remote_complete Dallas
0|</LAT>

```

```

Nov 26 2022 21:08:43 EVENT COMPLETED: site_down_complete Dallas 0
<LAT>|2022-11-26T21:08:43|7839|EVENT COMPLETED: site_down_complete Dallas
0|</LAT>
26/11/2022_21:08:46: End Event History records
26/11/2022_21:08:46:
=====
26/11/2022_21:08:46: ||
|| Test 2 Complete - NODE_DOWN_GRACEFUL: Stop cluster services gracefully on a node
||
26/11/2022_21:08:46: || Test Completion Status: PASSED
||
26/11/2022_21:08:46:
=====
26/11/2022_21:08:46:
=====
26/11/2022_21:08:46: ||
|| Starting Test 3 - NODE_UP,jessica,Restart cluster services on the node that was stopped
||
26/11/2022_21:08:46:
=====
26/11/2022_21:08:47: -----
26/11/2022_21:08:47: | is_rational NODE_UP
26/11/2022_21:08:47: -----
26/11/2022_21:08:47: Checking cluster stability
26/11/2022_21:08:47: jessica: ST_INIT
26/11/2022_21:08:47: jordan: ST_STABLE
26/11/2022_21:08:47: Cluster is stable
26/11/2022_21:08:47: Node: jessica, State: ST_INIT
26/11/2022_21:08:47: -----
26/11/2022_21:08:47: | Executing Command for NODE_UP
26/11/2022_21:08:47: -----
26/11/2022_21:08:47: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e NODE_UP -m
execute 'jessica'
26/11/2022_21:09:08: -----
26/11/2022_21:09:08: | Entering wait_for_stable
26/11/2022_21:09:08: -----
26/11/2022_21:09:08: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:09:38: Checking Node States:
26/11/2022_21:09:38: Node jessica: ST_STABLE
26/11/2022_21:09:38: Active Timers: None
26/11/2022_21:09:38: Node jordan: ST_STABLE
26/11/2022_21:09:38: Active Timers: None
26/11/2022_21:09:38: -----
26/11/2022_21:09:38: | NODE_UP: Checking post-event status
26/11/2022_21:09:38: -----
26/11/2022_21:09:38: Event Nodes: jessica
26/11/2022_21:09:39: pre-event online nodes: jordan
26/11/2022_21:09:39: post-event online nodes: jessica jordan
26/11/2022_21:09:39: Checking node states
26/11/2022_21:09:39: jessica: Preevent state: ST_INIT, Postevent state: ST_STABLE
26/11/2022_21:09:39: jordan: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:09:39: Checking RG states
26/11/2022_21:09:39: Resource Group: redbookrg
26/11/2022_21:09:39: Node: jessica Pre Event State: OFFLINE, Post Event State: ONLINE
26/11/2022_21:09:39: Node: jordan Pre Event State: OFFLINE, Post Event State: OFFLINE
26/11/2022_21:09:39: Checking event history
26/11/2022_21:09:39: Begin Event History records:
26/11/2022_21:09:39: NODE: jessica
Nov 26 2022 21:09:09 EVENT COMPLETED: admin_op clrm_start_request 24436 0 0

```

```

<LAT>|2022-11-26T21:09:09|24436|EVENT COMPLETED: admin_op clrm_start_request
24436 0 0|</LAT>
 Nov 26 2022 21:09:14 EVENT COMPLETED: site_up_local Dallas 0
 <LAT>|2022-11-26T21:09:14|24437|EVENT COMPLETED: site_up_local Dallas 0|</LAT>
 Nov 26 2022 21:09:14 EVENT COMPLETED: site_up Dallas 0
 <LAT>|2022-11-26T21:09:14|24437|EVENT COMPLETED: site_up Dallas 0|</LAT>
 Nov 26 2022 21:09:15 EVENT COMPLETED: node_up jessica 0
 <LAT>|2022-11-26T21:09:15|24437|EVENT COMPLETED: node_up jessica 0|</LAT>
 Nov 26 2022 21:09:17 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:09:17|24438|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:09:18 EVENT COMPLETED: acquire_service_addr 0
 <LAT>|2022-11-26T21:09:18|24438|EVENT COMPLETED: acquire_service_addr 0|</LAT>
 Nov 26 2022 21:09:20 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
 <LAT>|2022-11-26T21:09:20|24438|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
 Nov 26 2022 21:09:20 EVENT COMPLETED: rg_move_acquire jessica 1 0
 <LAT>|2022-11-26T21:09:20|24438|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
 Nov 26 2022 21:09:21 EVENT COMPLETED: start_server demoapp 0
 <LAT>|2022-11-26T21:09:21|24438|EVENT COMPLETED: start_server demoapp 0|</LAT>
 Nov 26 2022 21:09:21 EVENT COMPLETED: rg_move_complete jessica 1 0
 <LAT>|2022-11-26T21:09:21|24438|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
 Nov 26 2022 21:09:24 EVENT COMPLETED: node_up_complete jessica 0
 <LAT>|2022-11-26T21:09:24|24439|EVENT COMPLETED: node_up_complete jessica
0|</LAT>
 Nov 26 2022 21:09:24 EVENT COMPLETED: site_up_local_complete Dallas 0
 <LAT>|2022-11-26T21:09:24|24439|EVENT COMPLETED: site_up_local_complete Dallas
0|</LAT>
 Nov 26 2022 21:09:24 EVENT COMPLETED: site_up_complete Dallas 0
 <LAT>|2022-11-26T21:09:24|24439|EVENT COMPLETED: site_up_complete Dallas 0|</LAT>
26/11/2022_21:09:39: NODE: jordan
 Nov 26 2022 21:09:26 EVENT COMPLETED: site_up_remote Dallas 0
 <LAT>|2022-11-26T21:09:26|24437|EVENT COMPLETED: site_up_remote Dallas 0|</LAT>
 Nov 26 2022 21:09:26 EVENT COMPLETED: site_up Dallas 0
 <LAT>|2022-11-26T21:09:26|24437|EVENT COMPLETED: site_up Dallas 0|</LAT>
 Nov 26 2022 21:09:29 EVENT COMPLETED: node_up jessica 0
 <LAT>|2022-11-26T21:09:29|24437|EVENT COMPLETED: node_up jessica 0|</LAT>
 Nov 26 2022 21:09:32 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:09:32|24438|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:09:32 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
 <LAT>|2022-11-26T21:09:32|24438|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
 Nov 26 2022 21:09:32 EVENT COMPLETED: rg_move_acquire jessica 1 0
 <LAT>|2022-11-26T21:09:32|24438|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
 Nov 26 2022 21:09:35 EVENT COMPLETED: rg_move_complete jessica 1 0
 <LAT>|2022-11-26T21:09:35|24438|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
 Nov 26 2022 21:09:38 EVENT COMPLETED: node_up_complete jessica 0
 <LAT>|2022-11-26T21:09:38|24439|EVENT COMPLETED: node_up_complete jessica
0|</LAT>
 Nov 26 2022 21:09:39 EVENT COMPLETED: site_up_remote_complete Dallas 0
 <LAT>|2022-11-26T21:09:39|24439|EVENT COMPLETED: site_up_remote_complete Dallas
0|</LAT>
 Nov 26 2022 21:09:39 EVENT COMPLETED: site_up_complete Dallas 0
 <LAT>|2022-11-26T21:09:39|24439|EVENT COMPLETED: site_up_complete Dallas 0|</LAT>
26/11/2022_21:09:39: End Event History records
26/11/2022_21:09:39:
=====

```

```

26/11/2022_21:09:39: ||
|| Test 3 Complete - NODE_UP: Restart cluster services on the node that was stopped
||
26/11/2022_21:09:39: || Test Completion Status: PASSED
||
26/11/2022_21:09:39:
=====
26/11/2022_21:09:39:
=====
26/11/2022_21:09:39: ||
|| Starting Test 4 - NODE_DOWN_TAKEOVER,jordan,Stop cluster services with takeover on a
node
||
26/11/2022_21:09:39:
=====
26/11/2022_21:09:39: -----
26/11/2022_21:09:39: | is_rational NODE_DOWN_TAKEOVER
26/11/2022_21:09:39: -----
26/11/2022_21:09:39: Checking cluster stability
26/11/2022_21:09:39: jessica: ST_STABLE
26/11/2022_21:09:39: jordan: ST_STABLE
26/11/2022_21:09:39: Cluster is stable
26/11/2022_21:09:39: -----
26/11/2022_21:09:39: | Executing Command for NODE_DOWN_TAKEOVER
26/11/2022_21:09:39: -----
26/11/2022_21:09:39: /usr/es/sbin/cluster/utilities/cl_rsh -n jordan
/usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e NODE_DOWN_TAKEOVER -m execute 'jordan'
26/11/2022_21:09:42: -----
26/11/2022_21:09:42: | Entering wait_for_stable
26/11/2022_21:09:42: -----
26/11/2022_21:09:42: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:10:13: Checking Node States:
26/11/2022_21:10:13: Node jessica: ST_STABLE
26/11/2022_21:10:13: Active Timers: None
26/11/2022_21:10:13: Node jordan: ST_INIT
26/11/2022_21:10:13: -----
26/11/2022_21:10:13: | NODE_DOWN_TAKEOVER: Checking post-event status
26/11/2022_21:10:13: -----
26/11/2022_21:10:13: pre-event online nodes: jessica jordan
26/11/2022_21:10:13: post-event online nodes: jessica
26/11/2022_21:10:13: Checking node states
26/11/2022_21:10:13: jessica: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:10:13: jordan: Preevent state: ST_STABLE, Postevent state: ST_INIT
26/11/2022_21:10:13: Checking RG states
26/11/2022_21:10:13: Resource Group: redbookrg
26/11/2022_21:10:13: Node: jessica Pre Event State: ONLINE, Post Event State: ONLINE
26/11/2022_21:10:13: Node: jordan Pre Event State: OFFLINE, Post Event State: OFFLINE
26/11/2022_21:10:13: Checking event history
26/11/2022_21:10:13: Begin Event History records:
26/11/2022_21:10:13: NODE: jessica
 Nov 26 2022 21:09:42 EVENT COMPLETED: site_down_remote FortWorth 0
 <LAT>|2022-11-26T21:09:42|24439|EVENT COMPLETED: site_down_remote FortWorth
0|</LAT>
 Nov 26 2022 21:09:42 EVENT COMPLETED: site_down FortWorth 0
 <LAT>|2022-11-26T21:09:42|24439|EVENT COMPLETED: site_down FortWorth 0|</LAT>
 Nov 26 2022 21:09:43 EVENT COMPLETED: node_down jordan 0
 <LAT>|2022-11-26T21:09:43|24439|EVENT COMPLETED: node_down jordan 0|</LAT>
 Nov 26 2022 21:09:45 EVENT COMPLETED: node_down_complete jordan 0
 <LAT>|2022-11-26T21:09:45|24440|EVENT COMPLETED: node_down_complete jordan
0|</LAT>

```

```

Nov 26 2022 21:09:46 EVENT COMPLETED: site_down_remote_complete FortWorth 0
<LAT>|2022-11-26T21:09:46|24440|EVENT COMPLETED: site_down_remote_complete
FortWorth 0|</LAT>
Nov 26 2022 21:09:46 EVENT COMPLETED: site_down_complete FortWorth 0
<LAT>|2022-11-26T21:09:46|24440|EVENT COMPLETED: site_down_complete FortWorth
0|</LAT>
26/11/2022_21:10:13: NODE: jordan
Nov 26 2022 21:09:55 EVENT COMPLETED: admin_op clrm_stop_request 24440 0 0
<LAT>|2022-11-26T21:09:55|24440|EVENT COMPLETED: admin_op clrm_stop_request 24440
0 0|</LAT>
Nov 26 2022 21:09:56 EVENT COMPLETED: site_down_local 0
<LAT>|2022-11-26T21:09:56|24440|EVENT COMPLETED: site_down_local 0|</LAT>
Nov 26 2022 21:09:56 EVENT COMPLETED: site_down FortWorth 0
<LAT>|2022-11-26T21:09:56|24440|EVENT COMPLETED: site_down FortWorth 0|</LAT>
Nov 26 2022 21:09:57 EVENT COMPLETED: node_down jordan 0
<LAT>|2022-11-26T21:09:57|24440|EVENT COMPLETED: node_down jordan 0|</LAT>
Nov 26 2022 21:10:00 EVENT COMPLETED: node_down_complete jordan 0
<LAT>|2022-11-26T21:10:00|24441|EVENT COMPLETED: node_down_complete jordan
0|</LAT>
Nov 26 2022 21:10:00 EVENT COMPLETED: site_down_local_complete 0
<LAT>|2022-11-26T21:10:00|24441|EVENT COMPLETED: site_down_local_complete
0|</LAT>
Nov 26 2022 21:10:00 EVENT COMPLETED: site_down_complete FortWorth 0
<LAT>|2022-11-26T21:10:00|24441|EVENT COMPLETED: site_down_complete FortWorth
0|</LAT>
26/11/2022_21:10:13: End Event History records
26/11/2022_21:10:13:
=====
26/11/2022_21:10:13: ||
|| Test 4 Complete - NODE_DOWN_TAKEOVER: Stop cluster services with takeover on a node
||
26/11/2022_21:10:13: || Test Completion Status: PASSED
||
26/11/2022_21:10:13:
=====
26/11/2022_21:10:13:
=====
26/11/2022_21:10:13: ||
|| Starting Test 5 - NODE_UP,jordan,Restart cluster services on the node that was stopped
||
26/11/2022_21:10:13:
=====
26/11/2022_21:10:14: -----
26/11/2022_21:10:14: | is_rational NODE_UP
26/11/2022_21:10:14: -----
26/11/2022_21:10:14: Checking cluster stability
26/11/2022_21:10:14: jessica: ST_STABLE
26/11/2022_21:10:14: jordan: ST_INIT
26/11/2022_21:10:14: Cluster is stable
26/11/2022_21:10:14: Node: jordan, State: ST_INIT
26/11/2022_21:10:14: -----
26/11/2022_21:10:14: | Executing Command for NODE_UP
26/11/2022_21:10:14: -----
26/11/2022_21:10:14: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e NODE_UP -m
execute 'jordan'
26/11/2022_21:10:37: -----
26/11/2022_21:10:37: | Entering wait_for_stable
26/11/2022_21:10:37: -----
26/11/2022_21:10:37: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:11:07: Checking Node States:

```

```

26/11/2022_21:11:07: Node jessica: ST_STABLE
26/11/2022_21:11:07: Active Timers: None
26/11/2022_21:11:07: Node jordan: ST_STABLE
26/11/2022_21:11:07: Active Timers: None
26/11/2022_21:11:07: -----
26/11/2022_21:11:07: | NODE_UP: Checking post-event status
26/11/2022_21:11:07: -----
26/11/2022_21:11:07: Event Nodes: jordan
26/11/2022_21:11:08: pre-event online nodes: jessica
26/11/2022_21:11:08: post-event online nodes: jessica jordan
26/11/2022_21:11:08: Checking node states
26/11/2022_21:11:08: jessica: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:11:08: jordan: Preevent state: ST_INIT, Postevent state: ST_STABLE
26/11/2022_21:11:08: Checking RG states
26/11/2022_21:11:08: Resource Group: redbookrg
26/11/2022_21:11:08: Node: jessica Pre Event State: ONLINE, Post Event State: ONLINE
26/11/2022_21:11:08: Node: jordan Pre Event State: OFFLINE, Post Event State: OFFLINE
26/11/2022_21:11:08: Checking event history
26/11/2022_21:11:08: Begin Event History records:
26/11/2022_21:11:08: NODE: jessica
 Nov 26 2022 21:10:40 EVENT COMPLETED: site_up_remote FortWorth 0
 <LAT>|2022-11-26T21:10:40|30963|EVENT COMPLETED: site_up_remote FortWorth
0|</LAT>
 Nov 26 2022 21:10:40 EVENT COMPLETED: site_up FortWorth 0
 <LAT>|2022-11-26T21:10:41|30963|EVENT COMPLETED: site_up FortWorth 0|</LAT>
 Nov 26 2022 21:10:43 EVENT COMPLETED: node_up jordan 0
 <LAT>|2022-11-26T21:10:43|30963|EVENT COMPLETED: node_up jordan 0|</LAT>
 Nov 26 2022 21:10:46 EVENT COMPLETED: node_up_complete jordan 0
 <LAT>|2022-11-26T21:10:46|30964|EVENT COMPLETED: node_up_complete jordan 0|</LAT>
 Nov 26 2022 21:10:46 EVENT COMPLETED: site_up_remote_complete FortWorth 0
 <LAT>|2022-11-26T21:10:46|30964|EVENT COMPLETED: site_up_remote_complete
FortWorth 0|</LAT>
 Nov 26 2022 21:10:46 EVENT COMPLETED: site_up_complete FortWorth 0
 <LAT>|2022-11-26T21:10:46|30964|EVENT COMPLETED: site_up_complete FortWorth
0|</LAT>
26/11/2022_21:11:08: NODE: jordan
 Nov 26 2022 21:10:53 EVENT COMPLETED: admin_op clrm_start_request 30962 0 0
 <LAT>|2022-11-26T21:10:53|30962|EVENT COMPLETED: admin_op clrm_start_request
30962 0 0|</LAT>
 Nov 26 2022 21:10:57 EVENT COMPLETED: site_up_local FortWorth 0
 <LAT>|2022-11-26T21:10:57|30963|EVENT COMPLETED: site_up_local FortWorth 0|</LAT>
 Nov 26 2022 21:10:57 EVENT COMPLETED: site_up FortWorth 0
 <LAT>|2022-11-26T21:10:57|30963|EVENT COMPLETED: site_up FortWorth 0|</LAT>
 Nov 26 2022 21:10:58 EVENT COMPLETED: node_up jordan 0
 <LAT>|2022-11-26T21:10:58|30963|EVENT COMPLETED: node_up jordan 0|</LAT>
 Nov 26 2022 21:11:00 EVENT COMPLETED: node_up_complete jordan 0
 <LAT>|2022-11-26T21:11:00|30964|EVENT COMPLETED: node_up_complete jordan 0|</LAT>
 Nov 26 2022 21:11:01 EVENT COMPLETED: site_up_local_complete FortWorth 0
 <LAT>|2022-11-26T21:11:01|30964|EVENT COMPLETED: site_up_local_complete FortWorth
0|</LAT>
 Nov 26 2022 21:11:01 EVENT COMPLETED: site_up_complete FortWorth 0
 <LAT>|2022-11-26T21:11:01|30964|EVENT COMPLETED: site_up_complete FortWorth
0|</LAT>
26/11/2022_21:11:08: End Event History records
26/11/2022_21:11:08:
=====
26/11/2022_21:11:08: ||
|| Test 5 Complete - NODE_UP: Restart cluster services on the node that was stopped
||
26/11/2022_21:11:08: || Test Completion Status: PASSED

```

```

||

26/11/2022_21:11:08:

=====

26/11/2022_21:11:08:

=====

26/11/2022_21:11:08: ||

|| Starting Test 6 - NODE_DOWN_FORCED,jessica,Stop cluster services forced on a node

||

26/11/2022_21:11:08:

=====

26/11/2022_21:11:09: -----

26/11/2022_21:11:09: | is_rational NODE_DOWN_FORCED

26/11/2022_21:11:09: -----

26/11/2022_21:11:09: Checking cluster stability

26/11/2022_21:11:09: jessica: ST_STABLE

26/11/2022_21:11:09: jordan: ST_STABLE

26/11/2022_21:11:09: Cluster is stable

26/11/2022_21:11:09: Node: jessica, Force Down:

26/11/2022_21:11:09: Node: jordan, Force Down:

26/11/2022_21:11:09: -----

26/11/2022_21:11:09: | Executing Command for NODE_DOWN_FORCED

26/11/2022_21:11:09: -----

26/11/2022_21:11:09: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e NODE_DOWN_FORCED

-m execute 'jessica'

26/11/2022_21:11:11: -----

26/11/2022_21:11:11: | Entering wait_for_stable

26/11/2022_21:11:11: -----

26/11/2022_21:11:11: Waiting 30 seconds for cluster to stabilize.

26/11/2022_21:11:41: Checking Node States:

26/11/2022_21:11:41: Node jessica: ST_STABLE

26/11/2022_21:11:41: Active Timers: None

26/11/2022_21:11:41: Node jordan: ST_STABLE

26/11/2022_21:11:41: Active Timers: None

26/11/2022_21:11:41: -----

26/11/2022_21:11:41: | NODE_DOWN_FORCED: Checking post-event status

26/11/2022_21:11:41: -----

26/11/2022_21:11:42: pre-event online nodes: jessica jordan

26/11/2022_21:11:42: post-event online nodes: jessica jordan

26/11/2022_21:11:42: Checking forced down node lists

26/11/2022_21:11:42: Node: jessica, Force Down: jessica

26/11/2022_21:11:42: Checking node states

26/11/2022_21:11:42: jessica: Preevent state: ST_STABLE, Postevent state: ST_STABLE

26/11/2022_21:11:42: jordan: Preevent state: ST_STABLE, Postevent state: ST_STABLE

26/11/2022_21:11:42: Checking RG states

26/11/2022_21:11:42: Resource Group: redbookrg

26/11/2022_21:11:42: Node: jessica Pre Event State: ONLINE, Post Event State: UNMANAGED

26/11/2022_21:11:42: Node: jordan Pre Event State: OFFLINE, Post Event State: UNMANAGED

26/11/2022_21:11:42: Checking event history

26/11/2022_21:11:42: Begin Event History records:

26/11/2022_21:11:42: NODE: jessica
 Nov 26 2022 21:11:10 EVENT COMPLETED: admin_op clrm_stop_request 30965 0 0
 <LAT>|2022-11-26T21:11:10|30965|EVENT COMPLETED: admin_op clrm_stop_request 30965
 0 0|</LAT>
 Nov 26 2022 21:11:13 EVENT COMPLETED: site_down_local 0
 <LAT>|2022-11-26T21:11:13|30965|EVENT COMPLETED: site_down_local 0|</LAT>
 Nov 26 2022 21:11:13 EVENT COMPLETED: site_down Dallas 0
 <LAT>|2022-11-26T21:11:13|30965|EVENT COMPLETED: site_down Dallas 0|</LAT>
 Nov 26 2022 21:11:14 EVENT COMPLETED: node_down jessica forced 0
 <LAT>|2022-11-26T21:11:14|30965|EVENT COMPLETED: node_down jessica forced
 0|</LAT>
```

```

Nov 26 2022 21:11:16 EVENT COMPLETED: node_down_complete jessica forced 0
<LAT>|2022-11-26T21:11:16|30966|EVENT COMPLETED: node_down_complete jessica
forced 0|</LAT>
 Nov 26 2022 21:11:16 EVENT COMPLETED: site_down_local_complete 0
 <LAT>|2022-11-26T21:11:16|30966|EVENT COMPLETED: site_down_local_complete
0|</LAT>
 Nov 26 2022 21:11:16 EVENT COMPLETED: site_down_complete Dallas 0
 <LAT>|2022-11-26T21:11:17|30966|EVENT COMPLETED: site_down_complete Dallas
0|</LAT>
26/11/2022_21:11:42: NODE: jordan
 Nov 26 2022 21:11:28 EVENT COMPLETED: site_down_remote Dallas 0
 <LAT>|2022-11-26T21:11:28|30965|EVENT COMPLETED: site_down_remote Dallas 0|</LAT>
 Nov 26 2022 21:11:28 EVENT COMPLETED: site_down Dallas 0
 <LAT>|2022-11-26T21:11:28|30965|EVENT COMPLETED: site_down Dallas 0|</LAT>
 Nov 26 2022 21:11:28 EVENT COMPLETED: node_down jessica forced 0
 <LAT>|2022-11-26T21:11:28|30965|EVENT COMPLETED: node_down jessica forced
0|</LAT>
 Nov 26 2022 21:11:31 EVENT COMPLETED: node_down_complete jessica forced 0
 <LAT>|2022-11-26T21:11:31|30966|EVENT COMPLETED: node_down_complete jessica
forced 0|</LAT>
 Nov 26 2022 21:11:31 EVENT COMPLETED: site_down_remote_complete Dallas 0
 <LAT>|2022-11-26T21:11:31|30966|EVENT COMPLETED: site_down_remote_complete Dallas
0|</LAT>
 Nov 26 2022 21:11:31 EVENT COMPLETED: site_down_complete Dallas 0
 <LAT>|2022-11-26T21:11:31|30966|EVENT COMPLETED: site_down_complete Dallas
0|</LAT>
26/11/2022_21:11:42: End Event History records
26/11/2022_21:11:42:
=====
26/11/2022_21:11:42: ||
|| Test 6 Complete - NODE_DOWN_FORCED: Stop cluster services forced on a node
||
26/11/2022_21:11:42: || Test Completion Status: PASSED
||
26/11/2022_21:11:42:
=====
26/11/2022_21:11:42:
=====
26/11/2022_21:11:42: ||
|| Starting Test 7 - NODE_UP,jessica,Restart cluster services on the node that was stopped
||
26/11/2022_21:11:42:
=====
26/11/2022_21:11:42: -----
26/11/2022_21:11:42: | is_rational NODE_UP
26/11/2022_21:11:42: -----
26/11/2022_21:11:42: Checking cluster stability
26/11/2022_21:11:42: jessica: ST_STABLE
26/11/2022_21:11:42: jordan: ST_STABLE
26/11/2022_21:11:42: Cluster is stable
26/11/2022_21:11:42: Node: jessica, State: ST_STABLE
26/11/2022_21:11:42: Node: jessica, Force Down: jessica
26/11/2022_21:11:42: -----
26/11/2022_21:11:42: | Executing Command for NODE_UP
26/11/2022_21:11:42: -----
26/11/2022_21:11:42: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e NODE_UP -m
execute 'jessica'
26/11/2022_21:11:46: -----
26/11/2022_21:11:46: | Entering wait_for_stable
26/11/2022_21:11:46: -----

```

```

26/11/2022_21:11:46: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:12:17: Checking Node States:
26/11/2022_21:12:17: Node jessica: ST_STABLE
26/11/2022_21:12:17: Active Timers: None
26/11/2022_21:12:17: Node jordan: ST_STABLE
26/11/2022_21:12:17: Active Timers: None
26/11/2022_21:12:17: -----
26/11/2022_21:12:17: | NODE_UP: Checking post-event status
26/11/2022_21:12:17: -----
26/11/2022_21:12:17: Event Nodes: jessica
26/11/2022_21:12:17: pre-event online nodes: jessica jordan
26/11/2022_21:12:17: post-event online nodes: jessica jordan
26/11/2022_21:12:17: Checking node states
26/11/2022_21:12:17: jessica: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:12:17: jordan: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:12:17: Checking RG states
26/11/2022_21:12:17: Resource Group: redbookrg
26/11/2022_21:12:17: Node: jessica Pre Event State: UNMANAGED, Post Event State: ONLINE
26/11/2022_21:12:17: Node: jordan Pre Event State: UNMANAGED, Post Event State: OFFLINE
26/11/2022_21:12:17: Checking event history
26/11/2022_21:12:17: Begin Event History records:
26/11/2022_21:12:17: NODE: jessica
 Nov 26 2022 21:11:46 EVENT COMPLETED: admin_op clrm_start_request 30967 0 0
 <LAT>|2022-11-26T21:11:46|30967|EVENT COMPLETED: admin_op clrm_start_request
30967 0|</LAT>
 Nov 26 2022 21:11:47 EVENT COMPLETED: site_up_local Dallas 0
 <LAT>|2022-11-26T21:11:47|30967|EVENT COMPLETED: site_up_local Dallas 0|</LAT>
 Nov 26 2022 21:11:47 EVENT COMPLETED: site_up Dallas 0
 <LAT>|2022-11-26T21:11:47|30967|EVENT COMPLETED: site_up Dallas 0|</LAT>
 Nov 26 2022 21:11:48 EVENT COMPLETED: node_up jessica 0
 <LAT>|2022-11-26T21:11:48|30967|EVENT COMPLETED: node_up jessica 0|</LAT>
 Nov 26 2022 21:11:50 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:11:50|30968|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:11:51 EVENT COMPLETED: acquire_service_addr 0
 <LAT>|2022-11-26T21:11:51|30968|EVENT COMPLETED: acquire_service_addr 0|</LAT>
 Nov 26 2022 21:11:51 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
 <LAT>|2022-11-26T21:11:51|30968|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
 Nov 26 2022 21:11:51 EVENT COMPLETED: rg_move_acquire jessica 1 0
 <LAT>|2022-11-26T21:11:51|30968|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
 Nov 26 2022 21:11:55 EVENT COMPLETED: start_server demoapp 0
 <LAT>|2022-11-26T21:11:55|30968|EVENT COMPLETED: start_server demoapp 0|</LAT>
 Nov 26 2022 21:11:55 EVENT COMPLETED: rg_move_complete jessica 1 0
 <LAT>|2022-11-26T21:11:55|30968|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
 Nov 26 2022 21:11:57 EVENT COMPLETED: node_up_complete jessica 0
 <LAT>|2022-11-26T21:11:57|30969|EVENT COMPLETED: node_up_complete jessica
0|</LAT>
 Nov 26 2022 21:11:57 EVENT COMPLETED: site_up_local_complete Dallas 0
 <LAT>|2022-11-26T21:11:57|30969|EVENT COMPLETED: site_up_local_complete Dallas
0|</LAT>
 Nov 26 2022 21:11:58 EVENT COMPLETED: site_up_complete Dallas 0
 <LAT>|2022-11-26T21:11:58|30969|EVENT COMPLETED: site_up_complete Dallas 0|</LAT>
26/11/2022_21:12:17: NODE: jordan
 Nov 26 2022 21:12:01 EVENT COMPLETED: site_up_remote Dallas 0
 <LAT>|2022-11-26T21:12:01|30967|EVENT COMPLETED: site_up_remote Dallas 0|</LAT>
 Nov 26 2022 21:12:01 EVENT COMPLETED: site_up Dallas 0
 <LAT>|2022-11-26T21:12:01|30967|EVENT COMPLETED: site_up Dallas 0|</LAT>
 Nov 26 2022 21:12:02 EVENT COMPLETED: node_up jessica 0

```

```

<LAT>|2022-11-26T21:12:02|30967|EVENT COMPLETED: node_up jessica 0|</LAT>
Nov 26 2022 21:12:04 EVENT COMPLETED: rg_move_fence jessica 1 0
<LAT>|2022-11-26T21:12:04|30968|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
Nov 26 2022 21:12:05 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
<LAT>|2022-11-26T21:12:05|30968|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
Nov 26 2022 21:12:05 EVENT COMPLETED: rg_move_acquire jessica 1 0
<LAT>|2022-11-26T21:12:05|30968|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
Nov 26 2022 21:12:08 EVENT COMPLETED: rg_move_complete jessica 1 0
<LAT>|2022-11-26T21:12:08|30968|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
Nov 26 2022 21:12:12 EVENT COMPLETED: node_up_complete jessica 0
<LAT>|2022-11-26T21:12:12|30969|EVENT COMPLETED: node_up_complete jessica
0|</LAT>
Nov 26 2022 21:12:12 EVENT COMPLETED: site_up_remote_complete Dallas 0
<LAT>|2022-11-26T21:12:12|30969|EVENT COMPLETED: site_up_remote_complete Dallas
0|</LAT>
Nov 26 2022 21:12:12 EVENT COMPLETED: site_up_complete Dallas 0
<LAT>|2022-11-26T21:12:12|30969|EVENT COMPLETED: site_up_complete Dallas 0|</LAT>
26/11/2022_21:12:17: End Event History records
26/11/2022_21:12:17:
=====
26/11/2022_21:12:17: ||
|| Test 7 Complete - NODE_UP: Restart cluster services on the node that was stopped
||
26/11/2022_21:12:17: || Test Completion Status: PASSED
||
26/11/2022_21:12:17:
=====
26/11/2022_21:12:17:
#####
26/11/2022_21:12:17: ##
Cluster Testing Complete: Exit Code 0
##
26/11/2022_21:12:17:
#####
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: | Initializing Variable Table
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: Using Process Environment for Variable Table
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: | Reading Static Configuration Data
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: Cluster Name: jessica_cluster
26/11/2022_21:12:18: Cluster Version: 23
26/11/2022_21:12:18: Local Node Name: jessica
26/11/2022_21:12:18: Cluster Nodes: jessica jordan
26/11/2022_21:12:18: Found 1 Cluster Networks
26/11/2022_21:12:18: Found 4 Cluster Interfaces/Device/Labels
26/11/2022_21:12:18: Found 1 Cluster Resource Groups
26/11/2022_21:12:18: Found 10 Cluster Resources
26/11/2022_21:12:18: Event Timeout Value: 720
26/11/2022_21:12:18: Maximum Timeout Value: 2880
26/11/2022_21:12:18: Found 2 Cluster Sites
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: | Building Test Queue
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: Test Plan: /usr/es/sbin/cluster/cl_testtool/auto_vg
26/11/2022_21:12:18: Event 1: VG_DOWN: VG_DOWN,vg1,ANY,Bring down volume group

```

```

26/11/2022_21:12:18: -----
26/11/2022_21:12:18: | Validate VG_DOWN
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: Event node: ANY
26/11/2022_21:12:18: Configured nodes: jessica jordan
26/11/2022_21:12:18: VG: leevg, RG Name: redbookrg
26/11/2022_21:12:18:
#####
26/11/2022_21:12:18: ##
Starting Cluster Test Tool: -c -e /usr/es/sbin/cluster/cl_testtool/auto_vg
##
26/11/2022_21:12:18:
#####
26/11/2022_21:12:18: ||
|| Starting Test 1 - VG_DOWN,ANY,leevg
||
26/11/2022_21:12:18:
=====
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: | is_rational VG_DOWN
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: Checking cluster stability
26/11/2022_21:12:18: jessica: ST_STABLE
26/11/2022_21:12:18: jordan: ST_STABLE
26/11/2022_21:12:18: Cluster is stable
26/11/2022_21:12:18: VG: leevg, RG: redbookrg, ONLINE NODES: jessica
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: | Executing Command for VG_DOWN
26/11/2022_21:12:18: -----
26/11/2022_21:12:18: /usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e VG_DOWN -m
execute 'leevg'
26/11/2022_21:12:19: -----
26/11/2022_21:12:19: | Entering wait_for_stable
26/11/2022_21:12:19: -----
26/11/2022_21:12:19: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:12:49: Checking Node States:
26/11/2022_21:12:49: Node jessica: ST_STABLE
26/11/2022_21:12:49: Active Timers: None
26/11/2022_21:12:49: Node jordan: ST_STABLE
26/11/2022_21:12:49: Active Timers: None
26/11/2022_21:12:49: -----
26/11/2022_21:12:49: | VG_DOWN: Checking post-event status
26/11/2022_21:12:49: -----
26/11/2022_21:12:50: RESID: 11, RG: redbookrg, Rgid: 1, TYPE: 0
26/11/2022_21:12:50: Checking node states
26/11/2022_21:12:50: jessica: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:12:50: jordan: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:12:50: Volume Group: leevg Failure Action: fallover
26/11/2022_21:12:50: Checking RG states
26/11/2022_21:12:50: Resource Group: redbookrg
26/11/2022_21:12:50: Node: jessica Pre Event State: ONLINE, Post Event State: OFFLINE
26/11/2022_21:12:50: Node: jordan Pre Event State: OFFLINE, Post Event State: ONLINE
26/11/2022_21:12:50: Checking event history
26/11/2022_21:12:50: Begin Event History records:
26/11/2022_21:12:50: NODE: jessica
 Nov 26 2022 21:12:20 EVENT COMPLETED: resource_state_change jessica 0
 <LAT>|2022-11-26T21:12:20|30970|EVENT COMPLETED: resource_state_change jessica
 0|</LAT>

```

```

Nov 26 2022 21:12:21 EVENT COMPLETED: stop_server demoapp 0
<LAT>|2022-11-26T21:12:21|30971|EVENT COMPLETED: stop_server demoapp 0|</LAT>
Nov 26 2022 21:12:24 EVENT COMPLETED: release_service_addr 0
<LAT>|2022-11-26T21:12:24|30971|EVENT COMPLETED: release_service_addr 0|</LAT>
Nov 26 2022 21:12:24 EVENT COMPLETED: rg_move jessica 1 RELEASE 0
<LAT>|2022-11-26T21:12:24|30971|EVENT COMPLETED: rg_move jessica 1 RELEASE
0|</LAT>
Nov 26 2022 21:12:24 EVENT COMPLETED: rg_move_release jessica 1 0
<LAT>|2022-11-26T21:12:24|30971|EVENT COMPLETED: rg_move_release jessica 1
0|</LAT>
Nov 26 2022 21:12:26 EVENT COMPLETED: rg_move_fence jessica 1 0
<LAT>|2022-11-26T21:12:26|30971|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
Nov 26 2022 21:12:29 EVENT COMPLETED: rg_move_fence jessica 1 0
<LAT>|2022-11-26T21:12:29|30972|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
Nov 26 2022 21:12:29 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
<LAT>|2022-11-26T21:12:29|30972|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
Nov 26 2022 21:12:29 EVENT COMPLETED: rg_move_acquire jessica 1 0
<LAT>|2022-11-26T21:12:29|30972|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
Nov 26 2022 21:12:32 EVENT COMPLETED: rg_move_complete jessica 1 0
<LAT>|2022-11-26T21:12:32|30972|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
Nov 26 2022 21:12:35 EVENT COMPLETED: resource_state_change_complete jessica 0
<LAT>|2022-11-26T21:12:35|30973|EVENT COMPLETED: resource_state_change_complete
jessica 0|</LAT>
26/11/2022_21:12:50: NODE: jordan
Nov 26 2022 21:12:34 EVENT COMPLETED: resource_state_change jessica 0
<LAT>|2022-11-26T21:12:34|30970|EVENT COMPLETED: resource_state_change jessica
0|</LAT>
Nov 26 2022 21:12:35 EVENT COMPLETED: rg_move jessica 1 RELEASE 0
<LAT>|2022-11-26T21:12:35|30971|EVENT COMPLETED: rg_move jessica 1 RELEASE
0|</LAT>
Nov 26 2022 21:12:35 EVENT COMPLETED: rg_move_release jessica 1 0
<LAT>|2022-11-26T21:12:35|30971|EVENT COMPLETED: rg_move_release jessica 1
0|</LAT>
Nov 26 2022 21:12:41 EVENT COMPLETED: rg_move_fence jessica 1 0
<LAT>|2022-11-26T21:12:41|30971|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
Nov 26 2022 21:12:43 EVENT COMPLETED: rg_move_fence jessica 1 0
<LAT>|2022-11-26T21:12:43|30972|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
Nov 26 2022 21:12:46 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
<LAT>|2022-11-26T21:12:46|30972|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
Nov 26 2022 21:12:46 EVENT COMPLETED: rg_move_acquire jessica 1 0
<LAT>|2022-11-26T21:12:46|30972|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
Nov 26 2022 21:12:47 EVENT COMPLETED: start_server demoapp 0
<LAT>|2022-11-26T21:12:47|30972|EVENT COMPLETED: start_server demoapp 0|</LAT>
Nov 26 2022 21:12:48 EVENT COMPLETED: rg_move_complete jessica 1 0
<LAT>|2022-11-26T21:12:48|30972|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>
Nov 26 2022 21:12:50 EVENT COMPLETED: resource_state_change_complete jessica 0
<LAT>|2022-11-26T21:12:50|30973|EVENT COMPLETED: resource_state_change_complete
jessica 0|</LAT>
26/11/2022_21:12:50: End Event History records
26/11/2022_21:12:50:
=====
26/11/2022_21:12:50: ||
|| Test 1 Complete - VG_DOWN: Bring down volume group
||
```

```

26/11/2022_21:12:50: || Test Completion Status: PASSED
||
26/11/2022_21:12:50:
=====
26/11/2022_21:12:50:
#####
26/11/2022_21:12:50: ##
Cluster Testing Complete: Exit Code 0
##
26/11/2022_21:12:50:
#####
26/11/2022_21:12:50: -----
26/11/2022_21:12:50: | Initializing Variable Table
26/11/2022_21:12:50: -----
26/11/2022_21:12:50: Using Process Environment for Variable Table
26/11/2022_21:12:50: -----
26/11/2022_21:12:50: | Reading Static Configuration Data
26/11/2022_21:12:50: -----
26/11/2022_21:12:50: Cluster Name: jessica_cluster
26/11/2022_21:12:50: Cluster Version: 23
26/11/2022_21:12:50: Local Node Name: jessica
26/11/2022_21:12:51: Cluster Nodes: jessica jordan
26/11/2022_21:12:51: Found 1 Cluster Networks
26/11/2022_21:12:51: Found 4 Cluster Interfaces/Device/Labels
26/11/2022_21:12:51: Found 1 Cluster Resource Groups
26/11/2022_21:12:51: Found 10 Cluster Resources
26/11/2022_21:12:51: Event Timeout Value: 720
26/11/2022_21:12:51: Maximum Timeout Value: 2880
26/11/2022_21:12:51: Found 2 Cluster Sites
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: | Building Test Queue
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: Test Plan: /usr/es/sbin/cluster/cl_testtool/auto_cluster_kill
26/11/2022_21:12:51: Event 1: CLSTRMGR_KILL: CLSTRMGR_KILL,node1,Kill the cluster manager
on a node
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: | Validate CLSTRMGR_KILL
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: Event node: jordan
26/11/2022_21:12:51: Configured nodes: jessica jordan
26/11/2022_21:12:51:
#####
26/11/2022_21:12:51: ##
Starting Cluster Test Tool: -c -e /usr/es/sbin/cluster/cl_testtool/auto_cluster_kill
##
26/11/2022_21:12:51:
=====
26/11/2022_21:12:51: ||
|| Starting Test 1 - CLSTRMGR_KILL,jordan,Kill the cluster manager on a node
||
26/11/2022_21:12:51:
=====
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: | is_rational CLSTRMGR_KILL
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: Checking cluster stability
26/11/2022_21:12:51: jessica: ST_STABLE
26/11/2022_21:12:51: jordan: ST_STABLE

```

```

26/11/2022_21:12:51: Cluster is stable
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: | Executing Command for CLSTRMGR_KILL
26/11/2022_21:12:51: -----
26/11/2022_21:12:51: /usr/es/sbin/cluster/utilities/cl_rsh -n jordan
/usr/es/sbin/cluster/cl_testtool/cl_testtool_ctrl -e CLSTRMGR_KILL -m execute 'jordan'
26/11/2022_21:12:52: -----
26/11/2022_21:12:52: | Entering wait_for_stable
26/11/2022_21:12:52: -----
26/11/2022_21:12:52: Waiting 30 seconds for cluster to stabilize.
26/11/2022_21:13:28: Checking Node States:
26/11/2022_21:13:28: Node jessica: ST_STABLE
26/11/2022_21:13:28: Active Timers
26/11/2022_21:13:28: monitor demoapp:::RestartTimer
26/11/2022_21:13:28: Node jordan:
26/11/2022_21:13:28: -----
26/11/2022_21:13:28: | CLSTRMGR_KILL: Checking post-event status
26/11/2022_21:13:28: -----
26/11/2022_21:13:34: pre-event online nodes: jessica jordan
26/11/2022_21:13:34: post-event online nodes: jessica
26/11/2022_21:13:34: Checking node states
26/11/2022_21:13:34: jessica: Preevent state: ST_STABLE, Postevent state: ST_STABLE
26/11/2022_21:13:34: jordan: Preevent state: ST_STABLE, Postevent state:
26/11/2022_21:13:34: Checking RG states
26/11/2022_21:13:34: Resource Group: redbookrg
26/11/2022_21:13:34: Node: jessica Pre Event State: OFFLINE, Post Event State: ONLINE
26/11/2022_21:13:34: Node: jordan Pre Event State: ONLINE, Post Event State: OFFLINE
26/11/2022_21:13:34: Checking event history
26/11/2022_21:13:34: Begin Event History records:
26/11/2022_21:13:34: NODE: jessica
 Nov 26 2022 21:12:52 EVENT COMPLETED: site_down_remote FortWorth 0
 <LAT>|2022-11-26T21:12:52|30974|EVENT COMPLETED: site_down_remote FortWorth
0|</LAT>
 Nov 26 2022 21:12:52 EVENT COMPLETED: site_down FortWorth 0
 <LAT>|2022-11-26T21:12:52|30974|EVENT COMPLETED: site_down FortWorth 0|</LAT>
 Nov 26 2022 21:12:52 EVENT COMPLETED: node_down jordan 0
 <LAT>|2022-11-26T21:12:52|30974|EVENT COMPLETED: node_down jordan 0|</LAT>
 Nov 26 2022 21:12:55 EVENT COMPLETED: rg_move jessica 1 RELEASE 0
 <LAT>|2022-11-26T21:12:55|30976|EVENT COMPLETED: rg_move jessica 1 RELEASE
0|</LAT>
 Nov 26 2022 21:12:55 EVENT COMPLETED: rg_move_release jessica 1 0
 <LAT>|2022-11-26T21:12:55|30976|EVENT COMPLETED: rg_move_release jessica 1
0|</LAT>
 Nov 26 2022 21:12:55 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:12:55|30976|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:12:57 EVENT COMPLETED: rg_move_fence jessica 1 0
 <LAT>|2022-11-26T21:12:57|30975|EVENT COMPLETED: rg_move_fence jessica 1 0|</LAT>
 Nov 26 2022 21:12:58 EVENT COMPLETED: acquire_service_addr 0
 <LAT>|2022-11-26T21:12:58|30975|EVENT COMPLETED: acquire_service_addr 0|</LAT>
 Nov 26 2022 21:13:00 EVENT COMPLETED: rg_move jessica 1 ACQUIRE 0
 <LAT>|2022-11-26T21:13:00|30975|EVENT COMPLETED: rg_move jessica 1 ACQUIRE
0|</LAT>
 Nov 26 2022 21:13:00 EVENT COMPLETED: rg_move_acquire jessica 1 0
 <LAT>|2022-11-26T21:13:00|30975|EVENT COMPLETED: rg_move_acquire jessica 1
0|</LAT>
 Nov 26 2022 21:13:10 EVENT COMPLETED: start_server demoapp 0
 <LAT>|2022-11-26T21:13:10|30975|EVENT COMPLETED: start_server demoapp 0|</LAT>
 Nov 26 2022 21:13:13 EVENT COMPLETED: rg_move_complete jessica 1 0
 <LAT>|2022-11-26T21:13:13|30975|EVENT COMPLETED: rg_move_complete jessica 1
0|</LAT>

```

```
Nov 26 2022 21:13:18 EVENT COMPLETED: node_down_complete jordan 0
<LAT>|2022-11-26T21:13:18|30977|EVENT COMPLETED: node_down_complete jordan
0|</LAT>
Nov 26 2022 21:13:18 EVENT COMPLETED: site_down_remote_complete FortWorth 0
<LAT>|2022-11-26T21:13:18|30977|EVENT COMPLETED: site_down_remote_complete
FortWorth 0|</LAT>
Nov 26 2022 21:13:18 EVENT COMPLETED: site_down_complete FortWorth 0
<LAT>|2022-11-26T21:13:18|30977|EVENT COMPLETED: site_down_complete FortWorth
0|</LAT>
26/11/2022_21:13:34: End Event History records
26/11/2022_21:13:34:
=====
26/11/2022_21:13:34: ||
|| Test 1 Complete - CLSTRMGR_KILL: Kill the cluster manager on a node
||
26/11/2022_21:13:34: || Test Completion Status: PASSED
||
26/11/2022_21:13:34:
=====
26/11/2022_21:13:34:
#####
26/11/2022_21:13:34: ##
Cluster Testing Complete: Exit Code 0
##
26/11/2022_21:13:34:
#####
```

---

# Abbreviations and acronyms

|               |                                         |                 |                                             |
|---------------|-----------------------------------------|-----------------|---------------------------------------------|
| <b>AES</b>    | Advanced Encryption Standard            | <b>FDDI</b>     | Fiber Distributed Data Interface            |
| <b>AFM</b>    | Active File Management                  | <b>GASA</b>     | General Application Smart Assist            |
| <b>APAR</b>   | Authorized Program Analysis Report      | <b>GLVM</b>     | Geographic Logical Volume Manager           |
| <b>API</b>    | application programming interface       | <b>GMVG</b>     | Geographic Mirrored Volume Group            |
| <b>ARAF</b>   | Automatic Release After a Failure       | <b>GPFS</b>     | General Parallel File System                |
| <b>ARP</b>    | Address Resolution Protocol             | <b>GRS</b>      | Global Replication Service                  |
| <b>ARU</b>    | Automatic Repository Update             | <b>HA</b>       | high availability                           |
| <b>AWS</b>    | Amazon Web Services                     | <b>HACMP</b>    | High Availability Cluster Multiprocessing   |
| <b>BOS</b>    | base operating system                   | <b>HACMP/ES</b> | HACMP Enhanced Scalability                  |
| <b>C-SPOC</b> | Cluster Single Point of Control         | <b>HADR</b>     | high availability and disaster recovery     |
| <b>CAA</b>    | Cluster Aware AIX                       | <b>HMC</b>      | Hardware Management Console                 |
| <b>CBM</b>    | Cloud Backup Management                 | <b>HPC</b>      | high-performance computing                  |
| <b>CBR</b>    | Cloud Backup and Recovery               | <b>HUR</b>      | Hitachi Universal Replicator                |
| <b>CCV</b>    | Cross-Cluster Verification              | <b>IaaS</b>     | infrastructure as a service                 |
| <b>CLI</b>    | command-line interface                  | <b>IBM</b>      | International Business Machines Corporation |
| <b>CoD</b>    | Capacity on Demand                      | <b>IBM HPS</b>  | IBM High Performance Switch                 |
| <b>CPC</b>    | central processing complex              | <b>IGMP</b>     | Internet Group Management Protocol          |
| <b>CtSec</b>  | Cluster Security                        | <b>IPAT</b>     | IP Address Takeover                         |
| <b>CuAT</b>   | Customized Attribute                    | <b>IPL</b>      | initial program load                        |
| <b>CUoD</b>   | Capacity Upgrade on Demand              | <b>IPv4</b>     | IP version 4                                |
| <b>CWDM</b>   | Coarse Wavelength Division Multiplexing | <b>IPv6</b>     | IP version 6                                |
| <b>DARE</b>   | dynamic automatic reconfiguration       | <b>iSCSI</b>    | Internet Small Computer System Interface    |
| <b>DCE</b>    | Distributed Computing Environment       | <b>IVM</b>      | Integrated Virtualization Manager           |
| <b>DER</b>    | Distinguished Encoding Rules            | <b>JFS</b>      | Journalized File System                     |
| <b>DLPAR</b>  | dynamic logical partition               | <b>JFS2</b>     | Journalized File System 2                   |
| <b>DNP</b>    | dynamic node priority                   | <b>JFSlog</b>   | Journalized File System log                 |
| <b>DNS</b>    | domain name server                      | <b>JSON</b>     | JavaScript Object Notation                  |
| <b>DR</b>     | disaster recovery                       | <b>LAN</b>      | local area network                          |
| <b>DVE</b>    | dynamic volume expansion                | <b>LDAP</b>     | Lightweight Directory Access Protocol       |
| <b>DWDM</b>   | Dense Wave length Division Multiplexing | <b>LPAR</b>     | logical partition or logical partitioning   |
| <b>ECM</b>    | enhanced concurrent mode                | <b>LPM</b>      | Live Partition Mobility                     |
| <b>ECVG</b>   | enhanced concurrent volume group        | <b>LUN</b>      | logical unit number                         |
| <b>EFS</b>    | Encrypted File System                   | <b>LV</b>       | logical volume                              |
| <b>EPCoD</b>  | Enterprise Pool CoD                     | <b>LVCB</b>     | logical volume control block                |
| <b>ESB</b>    | Enterprise Service Bus                  |                 |                                             |
| <b>FBHP</b>   | Fallback to Higher Priority Node        |                 |                                             |
| <b>FC</b>     | Fibre Channel                           |                 |                                             |
| <b>FCS</b>    | Fibre Channel Switch                    |                 |                                             |

|             |                                               |               |                                  |
|-------------|-----------------------------------------------|---------------|----------------------------------|
| <b>LVM</b>  | Logical Volume Manager                        | <b>SMUI</b>   | SystemMirror User Interface      |
| <b>MAN</b>  | metropolitan area network                     | <b>SMUX</b>   | SNMP multiplexing                |
| <b>MD5</b>  | Message Digest Version 5                      | <b>SOCC</b>   | Serial Optical Channel Converter |
| <b>MIB</b>  | Management Information Base                   | <b>SP</b>     | Service Pack                     |
| <b>MPIO</b> | multipath I/O                                 | <b>SPOF</b>   | single point of failure          |
| <b>NDD</b>  | network device driver                         | <b>SR-IOV</b> | single root I/O virtualization   |
| <b>NDP</b>  | Neighbor Discovery Protocol                   | <b>SRDF</b>   | Symmetrix Remote Data Facility   |
| <b>NDU</b>  | non-disruptive upgrade                        | <b>SRR</b>    | Simplified Remote Restart        |
| <b>NFS</b>  | Network File System                           | <b>SSH</b>    | Secure Shell                     |
| <b>NIB</b>  | network interface backup                      | <b>SSL</b>    | Secure Sockets Layer             |
| <b>NIC</b>  | network interface card                        | <b>SVC</b>    | SAN Volume Controller            |
| <b>NIM</b>  | Network Installation Management               | <b>SymKey</b> | symmetric key                    |
| <b>NIS</b>  | Network Information Service                   | <b>TL</b>     | Technology Level                 |
| <b>NPIV</b> | N_Port ID Virtualization                      | <b>UI</b>     | user interface                   |
| <b>NSC</b>  | No Site Cluster                               | <b>UPS</b>    | uninterruptible power supply     |
| <b>OCS</b>  | Operation Center Support                      | <b>UUID</b>   | universally unique identifier    |
| <b>ODM</b>  | Object Data Manager                           | <b>VFS</b>    | Veritas Foundation Suite         |
| <b>OFAN</b> | Online on First Available Node                | <b>VG</b>     | volume group                     |
| <b>OHNO</b> | Online on Home Node Only                      | <b>VGDA</b>   | Volume Group Descriptor Area     |
| <b>OTP</b>  | one-time password                             | <b>VIOS</b>   | Virtual I/O Server               |
| <b>PKS</b>  | platform keystore                             | <b>VIPA</b>   | Virtual IP Address               |
| <b>PP</b>   | physical partition                            | <b>VLAN</b>   | virtual local area network       |
| <b>PPRC</b> | Peer-to-Peer Remote Copy                      | <b>VM</b>     | virtual machine                  |
| <b>PV</b>   | physical volume                               | <b>VMRM</b>   | VM Recovery Manager              |
| <b>PVID</b> | physical volume identifier                    | <b>vNIC</b>   | virtual network interface card   |
| <b>RAC</b>  | Real Application Cluster                      | <b>VPN</b>    | virtual private network          |
| <b>RAID</b> | redundant array of independent disks          | <b>VxVM</b>   | Veritas Volume Manager           |
| <b>RAS</b>  | reliability, availability, and serviceability | <b>WAN</b>    | wide area network                |
| <b>RBAC</b> | role-based access control                     | <b>WDM</b>    | wave division multiplexing       |
| <b>RC</b>   | return code                                   | <b>WWPN</b>   | worldwide port name              |
| <b>RG</b>   | resource group                                |               |                                  |
| <b>RMC</b>  | Resource Monitoring and Control               |               |                                  |
| <b>ROHA</b> | Resource Optimized High Availability          |               |                                  |
| <b>RPV</b>  | Remote Physical Volume                        |               |                                  |
| <b>RSCT</b> | Reliable Scalable Cluster Technology          |               |                                  |
| <b>SAN</b>  | storage area network                          |               |                                  |
| <b>SAS</b>  | serial-attached SCSI                          |               |                                  |
| <b>SEA</b>  | Shared Ethernet Adapter                       |               |                                  |
| <b>SLIP</b> | Serial Line Internet Protocol                 |               |                                  |
| <b>SMIT</b> | System Management Interface Tool              |               |                                  |
| <b>SMS</b>  | Short Message Service                         |               |                                  |

# Related publications

The publications that are listed in this section are considered suitable for a more detailed description of the topics that are covered in this book.

## IBM Redbooks

The following IBM Redbooks publications provide additional information about the topics in this document. Some publications in this list might be available in softcopy only.

- ▶ *Asynchronous Geographic Logical Volume Mirroring Best Practices for Cloud Deployment*, REDP-5665
- ▶ *Guide to IBM PowerHA SystemMirror for AIX Version 7.1.3*, SG24-8167
- ▶ *IBM PowerHA SystemMirror 7.1.2 Enterprise Edition for AIX*, SG24-8106
- ▶ *IBM PowerHA SystemMirror Standard Edition 7.1.1 for AIX Update*, SG24-8030
- ▶ *IBM Power Systems High Availability and Disaster Recovery Updates: Planning for a Multicloud Environment*, REDP-5663
- ▶ *IBM Power Systems Private Cloud with Shared Utility Capacity: Featuring Power Enterprise Pools 2.0*, SG24-8478
- ▶ *IBM PowerVM Virtualization Introduction and Configuration*, SG24-7940
- ▶ *IBM System Storage DS8000 Copy Services for Open Systems*, SG24-6788
- ▶ *IBM System Storage SAN Volume Controller and Storwize V7000 Replication Family Services*, SG24-7574
- ▶ *ILM Library: Information Lifecycle Management Best Practices Guide*, SG24-7251
- ▶ *Implementing High Availability Cluster Multi-Processing (HACMP) Cookbook*, SG24-6769
- ▶ *Understanding LDAP - Design and Implementation*, SG24-4986

You can search for, view, download, or order these documents and other Redbooks, Redpapers, web docs, drafts, and additional materials, at the following website:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Online resources

These websites are also relevant as further information sources:

- ▶ *AIX Disaster Recovery with IBM Power Virtual Server: An IBM Systems Lab Services Tutorial*  
[https://cloud.ibm.com/media/docs/downloads/power-iaas-tutorials/PowerVS\\_AIX\\_DR\\_Tutorial\\_v1.pdf](https://cloud.ibm.com/media/docs/downloads/power-iaas-tutorials/PowerVS_AIX_DR_Tutorial_v1.pdf)
- ▶ *Global Replication Services Solution using IBM Power Virtual Server*  
[https://cloud.ibm.com/media/docs/downloads/power-iaas/Global\\_Replication\\_Services\\_Solution\\_using\\_IBM\\_Power\\_Virtual\\_Server.pdf](https://cloud.ibm.com/media/docs/downloads/power-iaas/Global_Replication_Services_Solution_using_IBM_Power_Virtual_Server.pdf)

- ▶ IBM PowerHA SystemMirror for AIX documentation  
<https://www.ibm.com/docs/en/powerha-aix>
- ▶ IBM Systems Director download page  
<http://www-03.ibm.com/systems/software/director/downloads/>
- ▶ List of current Service Packs (SPs) for PowerHA  
<http://www14.software.ibm.com/webapp/set2/sas/f/hacmp/home.html>
- ▶ List of supported devices by PowerHA  
<http://ibm.co/1EvK8cG>
- ▶ *Planning PowerHA SystemMirror*  
[http://public.dhe.ibm.com/systems/power/docs/powerha/72/hacmplangd\\_pdf.pdf](http://public.dhe.ibm.com/systems/power/docs/powerha/72/hacmplangd_pdf.pdf)
- ▶ PowerHA Enterprise Edition Cross Reference  
<http://tinyurl.com/haEEcompat>
- ▶ PowerHA frequently asked questions  
<http://www-03.ibm.com/systems/power/software/availability/aix/faq/index.html>

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)

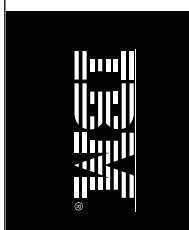


**Redbooks**

# IBM PowerHA SystemMirror for AIX Cookbook

SG24-7739-02

ISBN 0738461385



(1.0" spine)  
0.875" <-> 1.498"  
460 <-> 788 pages







SG24-7739-02

ISBN 0738461385

Printed in U.S.A.

Get connected

