

IBM PowerHA SystemMirror V7.2.1 for IBM AIX Updates

Dino Quintero
Shawn Bodily
Bernhard Buehler
Bunphot Chuprasertsuk
Bing He
Maria-Katharina Esser
Fabio Martins
Matthew W Radford
Antony Steel



Power Systems



International Technical Support Organization

IBM PowerHA SystemMirror V7.2.1 for IBM AIX Updates

May 2017

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

First Edition (May 2017)

This edition applies to:

IBM PowerHA SystemMirror Version 7.2.1

IBM Reliable Scalable Cluster Technology (RSCT) Version 3.2.1.0

IBM AIX 7.2.0 SP2

IBM Storwize V7000 Version 7.6.1.1

© Copyright International Business Machines Corporation 2017. All rights reserved.

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

Contents

Notices	ix
Trademarks	x
Preface	xi
Authors	xi
Now you can become a published author, too!	xiii
Comments welcome	xiii
Stay connected to IBM Redbooks	xiii
Chapter 1. Introduction to IBM PowerHA SystemMirror for AIX	1
1.1 What is PowerHA SystemMirror for AIX	2
1.1.1 High availability	2
1.1.2 Cluster multiprocessing	2
1.2 Availability solutions: An overview	3
1.2.1 Downtime	5
1.2.2 Single point of failure	5
1.3 History and evolution	6
1.3.1 PowerHA SystemMirror Version 7.1.1	7
1.3.2 PowerHA SystemMirror Version 7.1.2	8
1.3.3 PowerHA SystemMirror Version 7.1.3	8
1.3.4 PowerHA SystemMirror Version 7.2.0	9
1.3.5 PowerHA SystemMirror Version 7.2.1	10
1.4 High availability terminology and concepts	10
1.4.1 Terminology	10
1.5 Fault tolerance versus high availability	12
1.5.1 Fault-tolerant systems	12
1.5.2 High availability systems	12
1.6 Additional PowerHA resources	13
Chapter 2. IBM PowerHA SystemMirror V7.2.0 and V7.2.1 for IBM AIX new features	17
2.1 Resiliency enhancements	18
2.1.1 Integrated support for AIX Live Kernel Update	18
2.1.2 Automatic Repository Replacement	20
2.1.3 Verification enhancements	20
2.1.4 Using Logical Volume Manager rootvg failure monitoring	21
2.1.5 Live Partition Mobility automation	23
2.2 Cluster Aware AIX enhancements	25
2.2.1 Network failure detection tunable	25
2.2.2 Built-in NETMON logic	26
2.2.3 Traffic stimulation for better interface failure detection	26
2.2.4 Monitoring /var usage	26
2.2.5 New lscluster option -g	26
2.2.6 CAA level added to the lscluster -c output	26
2.3 Enhanced split-brain handling	27
2.4 Resource Optimized High Availability failovers by using enterprise pools	27
2.5 Nondisruptive upgrades	28
2.6 Geographic Logical Volume Manager wizard	28
2.7 New option for starting PowerHA by using clmgr	28
2.8 Graphical user interface	29

Chapter 3. Planning considerations	31
3.1 Introduction	32
3.1.1 Mirrored architecture.....	32
3.1.2 Single storage architecture.....	32
3.1.3 Stretched cluster.....	33
3.1.4 Linked cluster	34
3.2 Cluster Aware AIX repository disk.....	36
3.2.1 Preparing for a Cluster Aware AIX repository disk	36
3.2.2 Cluster Aware AIX with multiple storage devices	36
3.3 Cluster Aware AIX (CAA) tunables	41
3.3.1 CAA network monitoring	42
3.3.2 Network failure detection time.....	42
3.4 Important considerations for virtual input/output server	43
3.4.1 Using poll_uplink.....	43
3.4.2 Advantages for PowerHA when poll_uplink is used	46
3.5 Network considerations.....	46
3.5.1 Dual-adapter networks	46
3.5.2 Single-adapter network.....	47
3.5.3 The netmon.cf file	47
3.6 Network File System tie breaker.....	47
3.6.1 Introduction and concepts.....	47
3.6.2 Test environment setup	49
3.6.3 NFS server and client configuration	52
3.6.4 NFS tie-breaker configuration.....	54
3.6.5 NFS tie-breaker tests	58
3.6.6 Log entries for monitoring and debugging	62
Chapter 4. What is new with IBM Cluster Aware AIX and Reliable Scalable Clustering Technology	67
4.1 Cluster Aware AIX.....	68
4.1.1 Cluster Aware AIX tunables	68
4.1.2 What is new in Cluster Aware AIX: Overview	68
4.1.3 Monitoring /var usage	69
4.1.4 New lscluster option -g	71
4.1.5 Network Failure Detection Time (FDT)	80
4.2 Automatic repository update for the repository disk	82
4.2.1 Introduction to the Automatic Repository Update	82
4.2.2 Requirements for Automatic Repository Update.....	83
4.2.3 Configuring Automatic Repository Update	83
4.2.4 Automatic Repository Update operations	86
4.3 New manage option to start PowerHA	93
4.4 Reliable Scalable Cluster Technology overview	94
4.4.1 What Reliable Scalable Cluster Technology is	94
4.4.2 Reliable Scalable Cluster Technology components	94
4.5 PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX	103
4.5.1 Configuring PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX	104
4.5.2 Relationship between PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX	104
4.5.3 How to start and stop CAA and RSCT	110
Chapter 5. Migration	111
5.1 Migration planning.....	112
5.1.1 PowerHA SystemMirror V7.2.1 requirements	112

5.1.2	Deprecated features	113
5.1.3	Migration options.	113
5.1.4	Migration steps	113
5.1.5	Migration matrix to PowerHA SystemMirror 7.2.1	115
5.2	Migration scenarios from PowerHA V7.1.3	116
5.2.1	PowerHA V7.1.3 test environment overview.	116
5.2.2	Rolling migration from PowerHA V7.1.3	116
5.2.3	Offline migration from PowerHA V7.1.3	120
5.2.4	Snapshot migration from PowerHA V7.1.3	122
5.2.5	Nondisruptive upgrade from PowerHA V7.1.3	124
5.3	Migration scenarios from PowerHA V7.2.0	126
5.3.1	PowerHA V7.2.0 test environment overview.	127
5.3.2	Rolling migration from PowerHA V7.2.0	127
5.3.3	Offline migration from PowerHA V7.2.0	135
5.3.4	Snapshot migration from PowerHA V7.2.0	137
5.3.5	Nondisruptive upgrade from PowerHA V7.2.0	139
Chapter 6.	Resource Optimized High Availability	143
6.1	Resource Optimized High Availability concept and terminology	144
6.1.1	Environment requirement for Resource Optimized High Availability.	145
6.2	New PowerHA SystemMirror SMIT configuration panels for Resource Optimized High Availability.	146
6.2.1	Entry point to Resource Optimized High Availability.	146
6.2.2	Resource Optimized High Availability panel	147
6.2.3	HMC configuration	148
6.2.4	Hardware resource provisioning for application controller	155
6.2.5	Change>Show Default Cluster Tunable.	160
6.3	New PowerHA SystemMirror verification enhancement for Resource Optimized High Availability.	161
6.4	Planning for one Resource Optimized High Availability cluster environment	163
6.4.1	Consideration before Resource Optimized High Availability configuration	163
6.4.2	Configuration steps for Resource Optimized High Availability	173
6.5	Resource acquisition and release process introduction	174
6.5.1	Steps for allocation and for release	174
6.6	Introduction to resource acquisition	175
6.6.1	Query	176
6.6.2	Resource computation	180
6.6.3	Identifying the method of resource allocation	181
6.6.4	Acquiring the resource	183
6.7	Introduction to release of resources	185
6.7.1	Query	186
6.7.2	Synchronous and asynchronous mode.	191
6.7.3	Automatic resource release process after an operating system crash	191
6.8	Example 1: Setting up one Resource Optimized High Availability cluster (without On/Off CoD)	192
6.8.1	Requirement	192
6.8.2	Hardware topology	192
6.8.3	Cluster configuration	193
6.8.4	Showing the Resource Optimized High Availability configuration.	196
6.9	Test scenarios of Example 1 (without On/Off CoD)	198
6.9.1	Bringing two resource groups online.	198
6.9.2	Moving one resource group to another node	205
6.9.3	Restarting with the current configuration after the primary node crashes.	213

6.10 Example 2: Setting up one Resource Optimized High Availability cluster (with On/Off CoD)	215
6.10.1 Requirements	215
6.10.2 Hardware topology	215
6.10.3 Cluster configuration	216
6.10.4 Showing the Resource Optimized High Availability configuration	217
6.11 Test scenarios for Example 2 (with On/Off CoD)	219
6.11.1 Bringing two resource groups online	220
6.11.2 Bringing one resource group offline	225
6.12 Hardware Management Console high availability introduction	226
6.12.1 Switching to the backup HMC for the Power Enterprise Pool	228
6.13 Test scenario for HMC failover	229
6.13.1 Hardware topology	229
6.13.2 Bringing one resource group offline when the primary HMC fails	232
6.13.3 Testing summary	237
6.14 Managing, monitoring, and troubleshooting	237
6.14.1 The clmgr interface to manage Resource Optimized High Availability	237
6.14.2 Changing the DLPAR and CoD resources dynamically	240
6.14.3 View the Resource Optimized High Availability report	241
6.14.4 Troubleshooting DLPAR and CoD operations	241
Chapter 7. Geographic Logical Volume Manager configuration assistant	245
7.1 Introduction	246
7.1.1 Geographical Logical Volume Manager	246
7.1.2 GLVM configuration assistant	249
7.2 Prerequisites	250
7.3 Using the GLVM wizard	251
7.3.1 Test environment overview	251
7.3.2 Synchronous configuration	252
7.3.3 Asynchronous configuration	260
Chapter 8. Automation adaptation for Live Partition Mobility	271
8.1 Concept	272
8.1.1 Prerequisites for PowerHA node support of Live Partition Mobility	274
8.1.2 PowerHA fix requirement	274
8.1.3 Reducing the Live Partition Mobility freeze time	274
8.2 Operation flow to support Live Partition Mobility on a PowerHA node	274
8.2.1 Pre-migration operation flow	275
8.2.2 Post-migration operation flow	277
8.3 Example: Live Partition Mobility scenario for PowerHA V7.1	279
8.3.1 Topology introduction	279
8.3.2 Initial status	280
8.3.3 Manual pre-Live Partition Mobility operations	284
8.3.4 Performing Live Partition Mobility	291
8.3.5 Manual post-Live Partition Mobility operations	292
8.4 Live Partition Mobility SMIT panel	296
8.5 PowerHA V7.2 scenario and troubleshooting	297
8.5.1 Troubleshooting	298
Chapter 9. IBM PowerHA SystemMirror User Interface	303
9.1 Introduction	304
9.2 Installation	305
9.2.1 Planning	305
9.2.2 SMUI Client: Cluster nodes	306

9.2.3 SMUI server	306
9.2.4 Adding and removing clusters.....	308
9.3 Navigating	310
9.3.1 Event summaries	310
9.3.2 Log files.....	313
9.3.3 General.....	314
9.3.4 Network.....	315
9.3.5 Terminal session.....	316
9.4 Troubleshooting	316
9.4.1 Log files.....	316
9.4.2 Login problems	317
9.4.3 Adding clusters	317
9.4.4 Status not updating.....	318
Chapter 10. Cluster partitioning management update.....	319
10.1 Introduction to cluster partitioning	320
10.1.1 Causes of a partitioned cluster	321
10.1.2 Terminology	321
10.2 PowerHA cluster split and merge policies (before PowerHA V7.2.1)	322
10.2.1 Split policy.....	322
10.2.2 Merge policy	324
10.2.3 Configuration for the split and merge policy	325
10.3 PowerHA quarantine policy.....	332
10.3.1 Active node halt quarantine policy	332
10.3.2 Disk fencing quarantine	333
10.3.3 Configuration of quarantine policies	334
10.4 Changes in split and merge policies in PowerHA V7.2.1	342
10.4.1 Configuring the split and merge policy by using SMIT	343
10.4.2 Configuring the split and merge policy by using clmgr	345
10.4.3 Starting cluster services after a split	346
10.4.4 Migration and limitation	346
10.5 Considerations for using split and merge quarantine policies.....	347
10.6 Split and merge policy testing environment	350
10.6.1 Basic configuration	351
10.6.2 Specific hardware configuration for some scenarios.....	351
10.6.3 Initial PowerHA service status for each scenario	351
10.7 Scenario: Default split and merge policy.....	357
10.7.1 Scenario description	357
10.7.2 Split and merge configuration in PowerHA	358
10.7.3 Cluster split	359
10.7.4 Cluster merge	362
10.7.5 Scenario summary	363
10.8 Scenario: Split and merge policy with a disk tie breaker.....	364
10.8.1 Scenario description	364
10.8.2 Split and merge configuration in PowerHA	365
10.8.3 Cluster split	367
10.8.4 How to change the tie breaker group leader manually	370
10.8.5 Cluster merge	370
10.8.6 Scenario summary	371
10.9 Scenario: Split and merge policy with the NFS tie breaker.....	372
10.9.1 Scenario description	372
10.9.2 Setting up the NFS environment.....	372
10.9.3 Setting the NFS split and merge policies	374

10.9.4 Cluster split	376
10.9.5 Cluster merge	379
10.9.6 Scenario summary	379
10.10 Scenario: Split and merge policy is manual	380
10.10.1 Scenario description	380
10.10.2 Split and merge configuration in PowerHA	380
10.10.3 Cluster split	382
10.10.4 Cluster merge	385
10.10.5 Scenario summary	386
10.11 Scenario: Active node halt policy quarantine	386
10.11.1 Scenario description	386
10.11.2 HMC password-less access configuration	387
10.11.3 HMC configuration in PowerHA	388
10.11.4 Quarantine policy configuration in PowerHA	391
10.11.5 Simulating a cluster split	393
10.11.6 Cluster merge occurs	394
10.11.7 Scenario summary	395
10.12 Scenario: Enabling the disk fencing quarantine policy	395
10.12.1 Scenario description	395
10.12.2 Quarantine policy configuration in PowerHA	396
10.12.3 Simulating a cluster split	399
10.12.4 Simulating a cluster merge	403
10.12.5 Scenario summary	405
Chapter 11. IBM PowerHA SystemMirror special features	407
11.1 New option for starting PowerHA by using the clmgr command	408
11.1.1 PowerHA Resource Group dependency settings	408
11.1.2 Use case for using manage=delayed	409
Appendix A. SCSI reservations	413
SCSI reservations	414
ODM reserve policy	415
Persistent Reserve IN	417
Persistent Preserve OUT	417
Understanding register, reserve, and preempt	418
Unregister request	421
Release request	421
Clear request	421
Storage	422
More about PR reservations	422
Persistent reservation commands	423
Appendix B. IBM PowerHA: Live kernel update support	425
Live kernel update support	426
Example of live kernel update patching a kernel interim fix in a PowerHA environment	426
Related publications	435
IBM Redbooks	435
Other publications	435
Online resources	435
Help from IBM	436

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®	Power Systems™	Redpaper™
DS8000®	POWER6®	Redbooks (logo)  ®
GPFS™	POWER7®	RS/6000®
HACMP™	POWER8®	Storwize®
HyperSwap®	PowerHA®	SystemMirror®
IBM®	PowerVM®	XIV®
IBM Spectrum™	PureSystems®	
POWER®	Redbooks®	

The following terms are trademarks of other companies:

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

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

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 helps strengthen the position of the IBM PowerHA® SystemMirror® solution with a well-defined and documented deployment models within an IBM Power Systems™ virtualized environment, which provides customers with a planned foundation for business resilience and disaster recovery for their IBM Power Systems infrastructure solutions.

This publication addresses topics to help meet customers' complex high availability and disaster recovery requirements on IBM Power Systems servers to help maximize their systems' availability and resources, and provide technical documentation to transfer the how-to-skills to users and support teams.

This book is targeted at technical professionals (consultants, technical support staff, IT Architects, and IT Specialists) who are responsible for providing high availability and disaster recovery solutions and support with IBM PowerHA SystemMirror Standard and Enterprise Editions on IBM Power Systems servers.

Authors

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

Dino Quintero is a Complex Solutions Project Leader and an IBM Level 3 Certified Senior IT Specialist with the ITSO in Poughkeepsie, New York. His areas of expertise include enterprise continuous availability, enterprise systems management, system virtualization, technical computing, and clustering solutions. He is an Open Group Distinguished IT Specialist. Dino holds a Master of Computing Information Systems degree and a Bachelor of Science degree in Computer Science from Marist College.

Shawn Bodily is an IBM Champion for Power Systems and a Senior IT Consultant for Clear Technologies in Dallas, Texas. He has 24 years of IBM AIX® experience with the last 20 years specializing in high availability and disaster recovery that is primarily focused around PowerHA SystemMirror. He is a double AIX Advanced Technical Expert, and is certified in IBM POWER® Systems and IBM Storage. He has written and presented extensively about high availability and storage at technical conferences, webinars, and onsite to customers. He is an IBM Redbooks platinum author who has co-authored 10 other IBM Redbooks publications and three IBM Redpaper™ publications.

Bernhard Buehler is an IT Specialist in Germany. He works for IBM STG Lab Services in La Gaude, France. He has worked at IBM for 35 years and has 26 years of experience in AIX and the availability field. His areas of expertise include AIX, PowerHA SystemMirror, HA architecture, script programming, and AIX security. He is a co-author of several IBM Redbooks publications. He is also a co-author of several courses in the IBM AIX curriculum.

Bunphot Chuprasertsuk is a Senior Technical Specialist in Bangkok, Thailand. He has a Bachelor of Science degree in Information and Communication Technology from Mahidol University. He has five years of experience in AIX, POWER, and storage. He has extensive experience in the banking and retail industries, and with large, complex enterprise organizations and dynamic environments.

Bing He is a Consulting IT Specialist of the IBM Advanced Technical Skills (ATS) team in China. He has 15 years of experience with IBM Power Systems. He has worked at IBM for about 10 years. His areas of expertise include PowerHA SystemMirror, IBM PowerVM®, Disaster Recovery, HANA on Power, and performance tuning on AIX.

Maria-Katharina Esser is an IT Specialist for pre-sales technical support and works for the IBM System and Technology Group (STG) in Munich, Germany. She has worked for IBM for 28 years, and has 17 years of experience in AIX, POWER, and storage.

Fabio Martins is a Senior Software Support Specialist with IBM Technical Support Services in Brazil. He has worked at IBM for 12+ years. His areas of expertise include IBM AIX, IBM PowerVM, IBM PowerKVM, IBM PowerHA SystemMirror, PowerVC, IBM PureSystems®, IBM DS8000®, IBM Storwize®, Linux, and Brocade SAN switches and directors. He is a Certified Product Services Professional in Software Support Specialization and a Certified Advanced Technical Expert on IBM Power Systems. He has worked extensively on IBM Power Systems for Brazilian customers, providing technical leadership and support, including how-to questions, problem determination, root cause analysis, performance concerns, and other general complex issues. He holds a bachelor degree in Computer Science from Universidade Paulista (UNIP).

Matthew W Radford is a UNIX support specialist in the United Kingdom. He has worked in IBM for 18 years and has eight years of experience in AIX and High Availability Cluster Multi-Processing (IBM HACMP™). He holds a degree in Information Technology from the University of Glamorgan. Matt has co-authored two other IBM Redbooks publications.

Anthony Steel is a Senior IT Specialist in ITS Singapore. He has 22 years experience in the UNIX field, predominately AIX and Linux. He holds an honors degree in Theoretical Chemistry from the University of Sydney. His areas of expertise include scripting, system customization, performance, networking, high availability, and problem solving. He has written and presented on LVM, TCP/IP, and high availability both in Australia and throughout Asia Pacific.

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

Octavian Lascu

International Technical Support Organization, Poughkeepsie Center

Paul Moyer, Mike Coffey, Rajeev Nimmagadda, Prasad Dasari, Sharath Kacham
Aricent Technologies, an IBM Business Partner

Paul Desgranges

Groupe BULL

Kwan Ho Yau

IBM China

Ravi Shankar, Steven Finnes, Minh Pham, Alex Mcleod, Tom Weaver, Teresa Pham, Gus Schlachter, Isaac Silva, Alexa Mcleod, Timothy Thornal, PI Ganesh, Gary Lowther, Gary Domrow, Esdras E Cruz-Aguilar
IBM US

Jes Kiran, Srikanth Thanneeru, Madhusudhanan Duraisamy, Prabhanjan Gururaj
IBM India

Now you can become a published author, too!

Here's an opportunity to spotlight your skills, grow your career, and become a published author—all at the same time! Join an ITSO 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, International Technical Support Organization
Dept. HYTD Mail Station P099
2455 South Road
Poughkeepsie, NY 12601-5400

Stay connected to IBM Redbooks

- ▶ Find us on Facebook:
<http://www.facebook.com/IBMRedbooks>
- ▶ Follow us on Twitter:
<http://twitter.com/ibmredbooks>
- ▶ Look for 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>



Introduction to IBM PowerHA SystemMirror for AIX

This chapter provides an introduction to IBM PowerHA SystemMirror for newcomers to this solution and a refresher for those users that have implemented PowerHA SystemMirror and used it for many years.

This chapter covers the following topics:

- ▶ What is PowerHA SystemMirror for AIX
- ▶ Availability solutions: An overview
- ▶ History and evolution
- ▶ High availability terminology and concepts
- ▶ Fault tolerance versus high availability
- ▶ Additional PowerHA resources

1.1 What is PowerHA SystemMirror for AIX

PowerHA SystemMirror for AIX (also referred to as PowerHA) is the IBM Power Systems data center solution that helps protect critical business applications from outages, both planned and unplanned. One of the major objectives of PowerHA is to offer automatically continued business services by providing redundancy despite different component failures. PowerHA depends on Reliable Scalable Cluster Technology (RSCT) and Cluster Aware AIX (CAA).

RSCT is a set of low-level operating system components that allow the implementation of clustering technologies, such as IBM Spectrum™ Scale (formerly GPFS™). RSCT is distributed with AIX. On the current AIX release, AIX 7.2, RSCT is Version 3.2.1.0. After installing PowerHA and CAA file sets, the RSCT topology services subsystem is deactivated and all its functions are performed by CAA.

PowerHA Version 7.1 and later relies heavily on the CAA infrastructure that was introduced in AIX 6.1 TL6 and AIX 7.1. CAA provides communication interfaces and monitoring provision for PowerHA and execution by using CAA commands with `c1cmd`.

PowerHA Enterprise Edition also provides disaster recovery functions such as cross-site mirroring, IBM HyperSwap®, Geographical Logical Volume Mirroring, and many storage-based replication methods. These cross-site clustering methods support PowerHA functions between two geographic sites. For more information, see the *IBM PowerHA SystemMirror 7.1.2 Enterprise Edition for AIX*, SG24-8106.

For more information about features that are added in PowerHA V7.1.1 and later, see 1.3, “History and evolution” on page 6.

1.1.1 High availability

In today’s complex environments, providing continuous service for applications is a key component of a successful IT implementation. High availability is one of the components that contributes to providing continuous service for the application clients, by masking or eliminating both planned and unplanned systems and application downtime. A high availability 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.

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

In short, you can define *high availability* as the process of ensuring, by using duplicated or shared hardware resources that are managed by a specialized software component, that an application stays up and available for use.

1.1.2 Cluster multiprocessing

In addition to high availability, 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.

Although desirable, 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. This solution must be implemented by starting with the cluster planning and design phase.

PowerHA is only one of the high availability technologies, and it builds on increasingly reliable operating systems, hot-swappable hardware, and increasingly resilient applications, by offering monitoring and automated response.

A high availability solution that is based on PowerHA provides automated failure detection, diagnosis, application recovery, and node reintegration. PowerHA can also provide excellent horizontal and vertical scalability by combining other advanced functions, such as dynamic logical partitioning (DLPAR) and Capacity on Demand (CoD).

1.2 Availability solutions: An overview

Many solutions can provide a wide range of availability 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
Enhanced standalone	Hours	Until last transaction	Double most hardware components
High availability clustering	Seconds	Until last transaction	Double hardware and additional software costs
Fault-tolerant	Zero	No loss of data	Specialized hardware and software, and expensive

High availability 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 a reasonable cost

The highly available solution for IBM Power Systems offers distinct benefits:

- ▶ Proven solution with 27 years of product development
- ▶ Using *off-the-shelf* hardware components
- ▶ Proven commitment for supporting your customers
- ▶ IP version 6 (IPv6) support for both internal and external cluster communication
- ▶ Smart Assist technology enabling high availability 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 high availability (HA) design and detailed planning from end to end
- ▶ Elimination of single points of failure
- ▶ Selection of appropriate hardware
- ▶ Correct implementation (do not take *shortcuts*)
- ▶ Disciplined system administration practices and change control
- ▶ Documented operational procedures
- ▶ Comprehensive test plan and thorough testing

Figure 1-1 shows a typical PowerHA environment with both IP and non-IP heartbeat networks. Non-IP heartbeat uses the cluster repository disk and an optional storage area network (SAN).

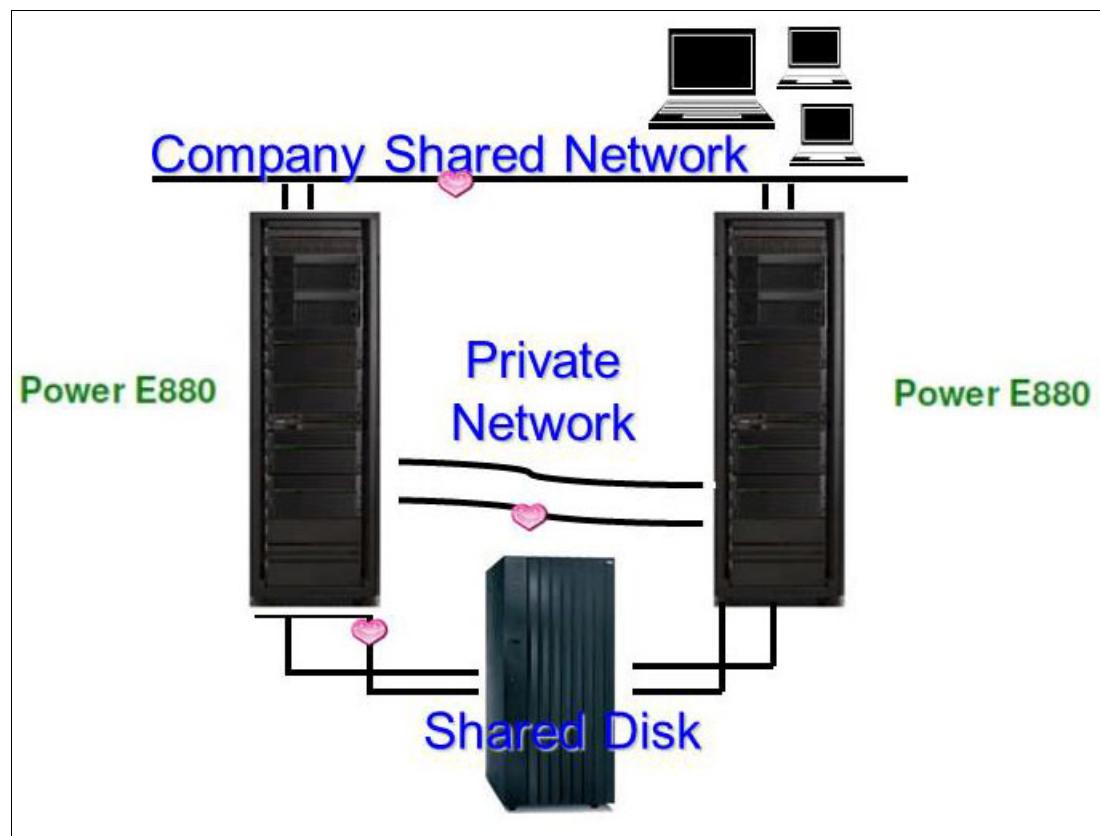


Figure 1-1 PowerHA cluster example

1.2.1 Downtime

Downtime is the period when an application is not available to serve its clients. Downtime can be classified in two categories: planned and unplanned.

- ▶ Planned
 - Hardware upgrades
 - Hardware or software repair or replacement
 - Software updates or upgrades
 - Backups (offline backups)
 - Testing (Periodic testing is required for good cluster maintenance.)
 - Development
- ▶ Unplanned
 - Administrator errors
 - Application failures
 - Hardware failures
 - Operating system errors
 - Environmental disasters

The role of PowerHA is to manage the application recovery after the outage. PowerHA provides monitoring and automatic recovery of the resources on which your application depends.

1.2.2 Single point of failure

A single point of failure (SPOF) is any individual component that is integrated into a cluster that, if it fails, renders the application unavailable for users.

Good design can remove single points of failure in the cluster: Nodes, storage, and networks. PowerHA manages these components and also the resources that are required by the application (including the application start/stop scripts).

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

To avoid single points of failure, use the following items:

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

A good design avoids single points of failure, and PowerHA can manage the availability of the application through the individual component failures. Table 1-2 lists each cluster object, which, if it fails, can result in loss of availability of the application. Each cluster object can be a physical or logical component.

Table 1-2 Single points of failure

Cluster object	SPOF eliminated by
Node (servers)	Multiple nodes.
Power/power supply	Multiple circuits, power supplies, or uninterruptible power supply (UPS).
Network	Multiple networks that are connected to each node, redundant network paths with independent hardware between each node and the clients.
Network adapters	Redundant adapters, and use other HA type features such as Etherchannel or Shared Ethernet Adapters (SEAs) by way of the Virtual I/O Server (VIOS).
I/O adapters	Redundant I/O adapters and multipathing software.
Controllers	Redundant controllers.
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	Use of more than one site for disaster recovery.
Resource groups	A resource group (RG) is a container of resources that are required to run the application. The SPOF is removed by moving the RG around the cluster to avoid failed components.

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

1.3 History and evolution

IBM High Availability Cluster Multi-Processing (HACMP) development started in 1990 to provide high availability 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 this publication was written. Instead, we provide highlights about the most recent versions.

Originally designed as a stand-alone product (known as HACMP classic) after the IBM high availability infrastructure known as RSCT became available, HACMP adopted this technology and became HACMP Enhanced Scalability (HACMP/ES) because it provides performance and functional advantages over the classic version. Starting with HACMP V5.1, there are no more classic versions. Later HACMP terminology was replaced with PowerHA in Version 5.5 and then PowerHA SystemMirror V6.1.

Starting with PowerHA V7.1, the CAA feature of the operating system is used to configure, verify, and monitor the cluster services. This major change improves the reliability of PowerHA because the cluster service functions now run in kernel space rather than user space. CAA was introduced in AIX 6.1 TL6. At the time of writing, the current release is PowerHA V7.2.1.

1.3.1 PowerHA SystemMirror Version 7.1.1

Released in September 2010, PowerHA V7.1.1 introduced improvements to PowerHA in terms of administration, security, and simplification of management tasks. The following list summarizes the improvements in PowerHA V7.1.1:

- ▶ Federated security allows cluster-wide single point of control, such as:
 - Encrypted file system (EFS) support
 - Role-based access control (RBAC) support
 - Authentication by using LDAP methods
- ▶ Logical volume manager (LVM) and C-SPOC enhancements:
 - EFS management by C-SPOC
 - Support for mirror pools
 - Disk renaming inside the cluster
 - Support for EMC, Hitachi, and HP disk subsystems multipathing LUN as a clustered repository disk
 - Capability to display a disk Universally Unique Identifier (UUID)
 - File system mounting feature (JFS2 Mount Guard), which prevents simultaneous mounting of the same file system by two nodes, which can cause data corruption
- ▶ Repository resiliency
- ▶ Dynamic automatic reconfiguration (DARE) progress indicator
- ▶ Application management improvements, such as a new application startup option

When you add an application controller, you can choose the application start mode. Now, you can choose background startup mode, which is the default and where the cluster activation moves forward with an application start script that runs in the background, or you can choose foreground start mode. When you choose the application controller option, the cluster activation is sequential, which means that cluster events hold application-start-script execution. If the application script ends with a failure (nonzero return code), the cluster activation also is considered to have failed.

- ▶ New network features, such as defining a network as private, use of netmon.cf file, and more network tunables

Note: Additional details and examples of implementing these features are found in *IBM PowerHA SystemMirror Standard Edition 7.1.1 for AIX Update*, SG24-8030.

1.3.2 PowerHA SystemMirror Version 7.1.2

Released in October 2012, PowerHA V7.1.2 continued to add features and functions:

- ▶ Two new cluster types (stretched and linked clusters):
 - Stretched cluster refers to a cluster that has sites that are defined in the same geographic location. It uses a shared repository disk. Extended distance sites with only IP connectivity are not possible with this cluster.
 - Linked cluster refers to a cluster with only IP connectivity across sites and is usually for PowerHA Enterprise Edition.
- ▶ IPv6 support reintroduced
- ▶ Backup repository disk
- ▶ Site support that is reintroduced with Standard Edition
- ▶ PowerHA Enterprise Edition reintroduced:
 - New HyperSwap support added for DS88XX:
All previous storage replication options that were supported in PowerHA 6.1 are supported:
 - IBM DS8000 Metro Mirror and Global Mirror
 - SAN Volume Controller Metro Mirror and Global Mirror
 - IBM Storwize v7000 Metro Mirror and Global Mirror
 - EMC SRDF synchronous and asynchronous replication
 - Hitachi TrueCopy and HUR replication
 - HP Continuous Access synchronous and asynchronous replication
 - Geographic Logical Volume Manager (GLVM)

Note: Additional details and examples of implementing some of these features are found in *IBM PowerHA SystemMirror 7.1.2 Enterprise Edition for AIX*, SG24-8106.

1.3.3 PowerHA SystemMirror Version 7.1.3

Released in October 2013, PowerHA V7.1.3 continued the development of PowerHA SystemMirror by adding further improvements in management, configuration simplification, automation, and performance areas. The following list summarizes the improvements in PowerHA V7.1.3:

- ▶ Unicast heartbeat
- ▶ Dynamic host name change
- ▶ Cluster split and merge handling policies
- ▶ **c1mgr** command enhancements:
 - Embedded hyphen and leading digit support in node labels
 - Native HTML report
 - Cluster copying through snapshots
 - Syntactical built-in help
 - Split and merge support
- ▶ CAA enhancements:
 - Scalability up to 32 nodes
 - Support for unicast and multicast
 - Dynamic host name or IP address support

- ▶ HyperSwap enhancements:
 - Active-active sites
 - One node HyperSwap
 - Auto resynchronization of mirroring
 - Node level unmanaged mode support
 - Enhanced repository disk swap management
- ▶ PowerHA plug-in enhancements for IBM Systems Director:
 - Restore snapshot wizard
 - Cluster simulator
 - Cluster split/merge support
- ▶ Smart Assist for SAP enhancements

Note: Additional details and examples of implementing some of these features are found in *IBM PowerHA SystemMirror for AIX Cookbook*, SG24-7739.

1.3.4 PowerHA SystemMirror Version 7.2.0

Released in December 2015, PowerHA V7.2 continued the development of PowerHA SystemMirror by adding further improvements in management, configuration simplification, automation, and performance areas. The following list summarizes the improvements in PowerHA V7.2:

- ▶ Resiliency enhancements:
 - Integrated support for AIX Live Kernel Update (LKU)
 - Automatic Repository Replacement (ARR)
 - Verification enhancements
 - Exploitation of LVM rootvg failure monitoring
 - Live Partition Mobility (LPM) automation
- ▶ CAA enhancements:
 - Network Failure Detection Tunable per interface
 - Built-in netmon logic
 - Traffic stimulation for better interface failure detection
- ▶ Enhanced split-brain handling:
 - Quarantine protection against “sick but not dead” nodes
 - NFS Tie Breaker support for split and merge policies
- ▶ Resource Optimized failovers by way of the Enterprise Pools (Resource Optimized High Availability (ROHA))
- ▶ Non-disruptive upgrades

The Systems Director plug-in was discontinued in PowerHA V7.2.0.

1.3.5 PowerHA SystemMirror Version 7.2.1

Released in December 2016, PowerHA V7.2.1 added the following additional improvements:

- ▶ Verification enhancements, some that are carried over from Version 7.2.0:
 - The reserve policy value must not be single path.
 - Checks for the consistency of /etc/filesystem. Do mount points exist and so on?
 - LVM PVID checks across LVM and ODM on various nodes.
 - Uses AIX Runtime Expert checks for LVM and NFS.
 - Checks for network errors. If they cross a threshold (5% of packet count receive and transmit), warn the administrator about the network issue.
 - GLVM buffer size checks.
 - Security configuration (password rules).
 - Kernel parameters: Tunables that are related to AIX network, VMM, and security.
- ▶ Expanded support of resource optimized failovers by way of the enterprise pools (ROHA).
- ▶ Browser-based GUI, which is called System Mirror User Interface (SMUI). The initial release is for monitoring and troubleshooting, not configuring clusters.
- ▶ All split/merge policies are now available to both standard and stretched clusters when using AIX 7.2.1.

1.4 High availability terminology and concepts

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

1.4.1 Terminology

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

Node	An IBM Power Systems (or LPAR) running AIX and PowerHA that are 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.
Cluster	A loosely coupled collection of independent systems (nodes) or logical partitions (LPARs) that are organized into a network for the purpose of sharing resources and communicating with each other.
Client	PowerHA defines relationships among cooperating systems where peer cluster nodes provide the services that are offered by a cluster node if that node cannot do so. These individual nodes are responsible for maintaining the functions of one or more applications in case of a failure of any cluster component.

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. All resources that together form a highly available application or service are grouped in RGs.
	PowerHA keeps the RG highly available as a single entity that can be moved from node to node in the event of a component or node failure. 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 efficient use of the cluster nodes.
Service IP label	A label that matches to a service IP address and is used for communications between clients and the node. A service IP label is part of an RG, which means that PowerHA can monitor it and keep it highly available.
IP address takeover (IPAT)	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	This is 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	This represents the movement of an RG from one active node to another node (backup node) in response to a failure on that active node.
Fallback	This represents 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	These consist of two types of processes: topology and group services. PowerHA uses group services, but depends on CAA for topology services. The cluster manager receives event information that is generated by these daemons and takes corresponding (response) actions in case of any failure.
Smart assists	A set of high availability agents, called <i>smart assists</i> , are bundled with the PowerHA SystemMirror Standard Edition to help discover and define high availability policies for most common middleware products.

1.5 Fault tolerance versus high availability

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

- ▶ Fault-tolerant systems
- ▶ High availability 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 shutdown 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 high availability are a combination of hardware and software components that are configured to work together to ensure automated recovery in case of failure with 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 node).

Therefore, eliminating all single points of failure (SPOF) 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 that are 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 to be integrated in a PowerHA cluster can require a fair amount of customization, possibly both at the application level and at the PowerHA and AIX platform level. 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.

PowerHA is not a fault-tolerant solution and should not be implemented as such.

1.6 Additional PowerHA resources

Here is a list of additional PowerHA resources and descriptions of each one:

- ▶ [Entitled Software Support \(download images\)](#)
- ▶ [PowerHA fixes](#)
- ▶ [PowerHA, CAA, and RSCT migration interim fixes](#)
- ▶ [PowerHA wiki](#)

This comprehensive resource contains links to all of the following references and much more.

- ▶ [PowerHA LinkedIn group](#)
- ▶ Base publications

All of the following PowerHA v7 publications are available at [IBM Knowledge Center](#):

- *Administering PowerHA SystemMirror*
- *Developing Smart Assist applications for PowerHA SystemMirror*
- *Geographic Logical Volume Manager for PowerHA SystemMirror Enterprise Edition*
- *Installing PowerHA SystemMirror*
- *Planning PowerHA SystemMirror*
- *PowerHA SystemMirror concepts*
- *PowerHA SystemMirror for IBM Systems Director*
- *Programming client applications for PowerHA SystemMirror*
- *Quick reference: clmgr command*
- *Smart Assists for PowerHA SystemMirror*
- *Storage-based high availability and disaster recovery for PowerHA SystemMirror Enterprise Edition*
- *Troubleshooting PowerHA SystemMirror*

- ▶ [PowerHA and Capacity Backup](#)

- ▶ Videos

Shawn Bodily has several PowerHA related videos on his [YouTube channel](#)

- ▶ [DeveloperWorks Discussion forum](#)
- ▶ IBM Redbooks publications

The main focus of each IBM PowerHA Redbooks publication differs a bit, but usually their main focus is covering what is new in a particular release. They generally have more details and advanced tips than the base publications.

Each new publication is rarely a complete replacement for the last. The only exception to this is *IBM PowerHA SystemMirror for AIX Cookbook*, SG24-7739. It was updated to Version 7.1.3 after replacing two previous cookbooks. It is probably the most comprehensive of all the current IBM Redbooks publications with regard to PowerHA Standard Edition specifically. Although there is some overlap across them, with multiple versions supported, it is important to reference the version of the book that is relevant to the version that you are using.

Figure 1-2 shows a list of relevant PowerHA IBM Redbooks publications. Although it still includes PowerHA 6.1 Enterprise Edition, which is no longer supported, that exact book is still the best reference for configuring EMC SRDF and Hitachi TrueCopy.

Redbooks Publication	Redbooks Publication Title	Exploiting IBM PowerHA SystemMirror V6.1 for AIX Enterprise Edition	IBM PowerHA SystemMirror Standard Edition 7.1.1 for AIX Update	IBM PowerHA SystemMirror 7.1.2 Enterprise Edition for AIX	Guide to IBM PowerHA SystemMirror for AIX Version 7.1.3	IBM PowerHA SystemMirror for AIX Cookbook	IBM PowerHA SystemMirror for AIX 7.1.3 Best Practices and Migration Guide
Topics	Publish Date	01 May 2013	24 October 2012	06 May 2013	28 September 2014	30 October 2014	02 February 2015
	Last Update	18 February 2014	23 July 2014	06 May 2015	16 June 2015	13 April 2015	-
	IBM Form Number	SG24-7841-01	SG24-8030-00	SG24-8106-00	SG24-8167-00	SG24-7739-01	SG24-8234-00
General information							
Concepts and overview		x	x	x	x	x	x
What's new		x	x	x	x		
Differences			x	x		x	
Cluster technology and components							
Cluster Aware AIX			x			x	
RSCT						x	
Planning							
Infrastructure considerations		x	x	x		x	x
Hardware and software requirements		x	x	x		x	
Design considerations			x			x	x
Disaster recovery							
Campus-style disaster recovery solutions		x					
Cross-site logical volume mirroring		x	x	x			
Extended distance disaster recovery solutions		x		x			
Metro Mirror and Global Mirror		x					
ESS/DS Metro Mirror		x					
SRDF replication		x					
Geographic Logical Volume Manager		x					
Disaster recovery with DS8700 Global Mirror		x					
Hitachi TrueCopy and Universal Replicator		x					
HyperSwap				x	x		
SVC Replication		x		x			
XIV Replication				x			
Installation and configuration							
Installation and configuration			x			x	
Resources and resources groups			x			x	
Networking			x			x	
Smart Assist		x					
Smart Assist for SAP		x			x		
Workload partitions		x				x	
DB2 with PowerHA					x		
Administration, monitoring, maintenance and management							
Administration, maintenance, management		x	x	x		x	
Security						x	
Monitoring					x		
Migration		x	x	x	x	x	
Cluster test tool						x	
IBM Systems Director plugin			x			x	
Cluster partitioning			x				
IBM PowerHA cluster simulator					x		
Other topics							
RBAC integration and implementation					x		
Dynamic host name change					x		
PowerHA and PowerVM						x	
Extending resource group capabilities						x	
Customizing resources and events						x	
File system conversion and migration							x
Symantec Cluster Server							x
PowerHA SE to EE cluster conversion							x

Figure 1-2 PowerHA IBM Redbooks publications reference

- ▶ White papers
 - [PowerHA V7.1 quick config guide](#)
 - [Implementing PowerHA with Storwize V7000](#)
 - [PowerHA with EMC V-Plex](#)
 - [Tips and Consideration with Oracle 11gR2 with PowerHA on AIX](#)

- *Tips and Consideration with Oracle 12cR1 with PowerHA on AIX*
- *Edison Group Report on the value of deep integration of PowerHA V7.1 and AIX*
- *PowerHA Case Study of Robert Wood Johnson University Hospital*
- *Performance Implications of LVM Mirroring*
- *AIX Higher Availability by using SAN services*



IBM PowerHA SystemMirror V7.2.0 and V7.2.1 for IBM AIX new features

This chapter covers the specific features that are new to IBM PowerHA SystemMirror for IBM AIX for Version 7.2 and Version 7.2.1.

This chapter covers the following topics:

PowerHA V7.2 Related:

- ▶ Resiliency enhancements:
 - Integrated support for AIX Live Kernel Update
 - Automatic Repository Replacement
 - Verification enhancements
 - Using Logical Volume Manager rootvg failure monitoring
 - Live Partition Mobility automation
- ▶ Cluster Aware AIX enhancements:
 - Network Failure Detection Tunable
 - Built-in NETMON logic
 - Traffic stimulation for better interface failure detection
 - Monitor /var usage
 - New `lscuster` option -g
- ▶ Enhanced split-brain handling:
 - Quarantine protection against “sick but not dead” nodes
 - NFS Tie Breaker support for split and merge policies
- ▶ Resource Optimized High Availability fallovers by using enterprise pools
- ▶ Nondisruptive upgrades
- ▶ Geographic Logical Volume Manager (GLVM) wizard

PowerHA V7.2.1 related:

- ▶ New option for starting PowerHA by using `clmgr`
- ▶ Graphical user interface
- ▶ Split and Merge enhancements
- ▶ PowerHA SystemMirror Resource Optimized High Availability (ROHA) enhancements

2.1 Resiliency enhancements

Every release of PowerHA SystemMirror aims to make the product even more resilient than its predecessors. PowerHA SystemMirror for AIX 7.2 continues this tradition.

2.1.1 Integrated support for AIX Live Kernel Update

AIX 7.2 introduced a new capability to allow concurrent patching without interruption to the applications. This capability is known as AIX Live Kernel Update (LNU). Initially, this capability is supported only for interim fixes, but it is the foundation for broader patching of service packs and eventually technologies levels in the future.

Tip: For more information about LKU, see [AIX Live Updates](#).

A demonstration of LKU is available in this [YouTube video](#).

Consider the following key points about PowerHA integrated support for LKUs:

- ▶ LKU can be performed on only one cluster node at a time.
- ▶ Support includes all PowerHA SystemMirror Enterprise Edition Storage replication features, including HyperSwap and GLVM.

However, for asynchronous GLVM, you must swap to sync mode before LKU is performed, and then swap back to async mode upon LKU completion.

- ▶ During LKU operation, enhanced concurrent volume groups (VGs) cannot be changed.
- ▶ Workloads continue to run without interruption.

PowerHA scripts and checks during Live Kernel Update

PowerHA provides scripts that are called during different phases of the AIX LKU notification mechanism. An overview of the PowerHA operations that are performed at which phase follows:

- ▶ Check phase:
 - Verifies that no other concurrent AIX Live Update is in progress in the cluster.
 - Verifies that the cluster is in stable state.
 - Verifies that there are no GLVM active asynchronous mirror pools.
- ▶ Pre-phase:
 - Switches the active Enhanced Concurrent VGs to *silent* mode.
 - Stops the cluster services and SRC daemons.
 - Stops GLVM traffic if required.

- ▶ Post phase:
 - Restarts GLVM traffic.
 - Restarts System Resource Controller (SRC) daemons and cluster services.
 - Restores the state of the Enhanced Concurrent VGs.

Enabling and disabling AIX Live Kernel Update support of PowerHA

As is the case for most of the features and functions of PowerHA, the feature can be enabled and disabled by using both the System Management Interface Tool (SMIT) and the **clmgr** command. In either case, it must be set on each node.

When enabling AIX LKU through SMIT, the option is set to either yes or no. However, when you use the **clmgr** command, the settings are true or false. The default is for it to be enabled (yes/true).

To modify by using SMIT, complete the following steps, as shown in Figure 2-1:

1. Run **smitty sysmirror** and select **Cluster Nodes and Networks → Manage Nodes → Change>Show a Node**.
2. Select the wanted node.
3. Set the Enable AIX Live Update operation field as wanted.
4. Press Enter.

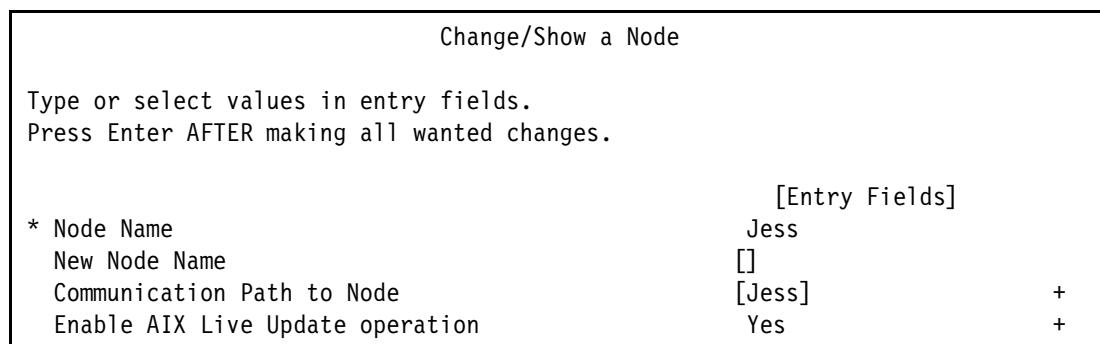


Figure 2-1 Enabling the AIX Live Kernel Update operation

Here is an example of how to check the current value of this setting by using the **clmgr** command:

```
[root@Jess] /# clmgr view node Jess |grep LIVE
ENABLE_LIVE_UPDATE="true"
```

Here is an example of how to disable this setting by using the **clmgr** command:

```
[root@Jess] /# clmgr modify node Jess ENABLE_LIVE_UPDATE=false
```

In order for the change to take effect, the cluster must be synchronized.

Logs that are generated during the AIX Live Kernel Update operation

The two logs that are used during the operation of an AIX LKU are both in the `/var/hacmp/log` directory:

- | | |
|--------------------------------|---|
| <code>lvupdate_orig.log</code> | This log file keeps information from the original source system logical partition (LPAR). |
| <code>lvupdate_surr.log</code> | This log file keeps information from the target surrogate system LPAR. |

Tip: A demonstration of performing an LKU is available in this [YouTube video](#).

2.1.2 Automatic Repository Replacement

Cluster Aware AIX (CAA) detects when a repository disk failure occurs and generates a notification message. The notification messages continue until the failed repository disk is replaced. PowerHA V7.1.1 introduced the ability to define a backup repository disk. However, the replacement procedure was a manual one. Beginning in PowerHA V7.2 and combined with AIX 7.1.4 or 7.2.0, Automatic Repository Update (ARU) can automatically swap a failed repository disk with the 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 clconfd to verify that they are still viable for an ARU operation. The steps to define a backup repository disk are the same as in previous versions of PowerHA. These steps and examples of failure situations can be found in 4.2, “Automatic repository update for the repository disk” on page 82.

Tip: An overview of configuring and a demonstration of Automatic Repository Replacement (ARR) can be found in this [YouTube video](#).

2.1.3 Verification enhancements

Cluster verification is the framework to check environmental conditions across all nodes in the cluster. Its purpose is to try to ensure proper operation of cluster events when they occur. Every new release of PowerHA provides more verification checks. In PowerHA V7.2, there are both new default additional checks, and a new option for detailed verification checks.

The following new additional checks are the defaults:

- ▶ Verify that the `reserve_policy` setting on shared disks is *not* set to `single_path`.
- ▶ Verify that `/etc/filesystems` entries for shared file systems are consistent across nodes.

The new detailed verification checks, which run only when explicitly enabled, include the following actions:

- ▶ The physical volume identifier (PVID) checks between the logical volume manager (LVM) and object data manager (ODM) on various nodes.
- ▶ Use AIX Runtime Expert checks for LVM and network file system (NFS).
- ▶ Checks whether network errors exceed a predefined 5% threshold.
- ▶ GLVM buffer size.
- ▶ Security configuration, such as password rules.
- ▶ Kernel parameters, such as network, Virtual Memory Manager (VMM), and so on.

Using the new detailed verification checks might add a significant amount of time to the verification process. To enable it, run **smitty sysmirror**, select Custom Cluster Configuration, then Verify and Synchronize Cluster Configuration (Advanced), and then set the option of Detailed Checks to Yes, as shown in Figure 2-2. This must be set manually each time because it always defaults to No. This option is only available if cluster services are not running.

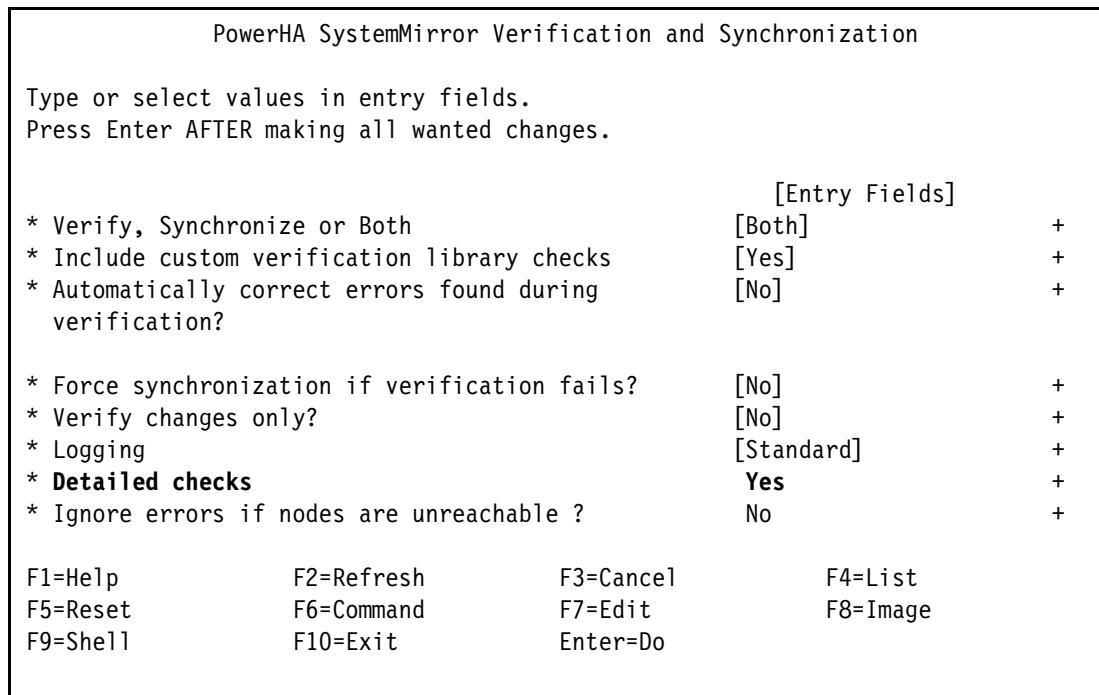


Figure 2-2 Enabling detail verification checking

2.1.4 Using Logical Volume Manager rootvg failure monitoring

AIX LVM recently added the capability to change a VG to be known as a *critical VG*. Though PowerHA has allowed critical VGs in the past, it applied only to non-operating system/data VGs. PowerHA V7.2 now also takes advantage of this function specifically for rootvg.

If the VG is set as a critical VG, any input/output (I/O) request failure starts the LVM metadata write operation to check the state of the disk before returning the I/O failure. If rootvg has the critical VG option set and if the system cannot access a quorum of rootvg disks or all rootvg disks if quorum is disabled, then the node is failed with a message sent to the console.

You can set and validate rootvg as a critical VG by running the commands that are shown in Figure 2-3. The command must run once because you use the **clcmd** CAA distributed command.

```
# clcmd chvg -r y rootvg
# clcmd lsvg rootvg |grep CRIT
DISK BLOCK SIZE: 512          CRITICAL VG: yes
DISK BLOCK SIZE: 512          CRITICAL VG: yes
```

Figure 2-3 Enabling rootvg as a critical volume group

Testing rootvg failure detection

In this environment, the rootvg is in Storwize V7000 logical unit numbers (LUNs) that are connected to the PowerHA nodes by virtual Fibre Channel (FC) adapters. Simulating a loss of any disk can often be accomplished in multiple ways, but often one of the following methods is used:

- ▶ From within the storage management, simply unmap the volumes from the host.
- ▶ Unmap the virtual FC adapter from the real adapter on the Virtual I/O Server (VIOS).
- ▶ Unzone the virtual worldwide port names (WWPNs) from the storage area network (SAN).

In this environment, we use the first option of unmapping from the storage side. The other two options usually affect all of the disks rather than only rootvg. However, usually that is fine too.

After the rootvg LUN is disconnected and detected, a kernel panic ensues. If the failure occurs on a PowerHA node that is hosting a resource group (RG), then an RG failover occurs as with any unplanned outage.

If you check the error report after restarting the system successfully, it has a kernel panic entry, as shown in Example 2-1.

Example 2-1 Kernel panic error report entry

```
-----  
LABEL: KERNEL_PANIC  
IDENTIFIER: 225E3B63  
  
Date/Time: Mon Jan 25 21:23:14 CST 2016  
Sequence Number: 140  
Machine Id: 00F92DB14C00  
Node Id: PHA72a  
Class: S  
Type: TEMP  
WPAR: Global  
Resource Name: PANIC
```

Description
SOFTWARE PROGRAM ABNORMALLY TERMINATED

Recommended Actions
PERFORM PROBLEM DETERMINATION PROCEDURES

Detail Data
ASSERT STRING

PANIC STRING
Critical VG Force off, halting.

The node must be restarted and cluster services resumed. As always, when a node rejoins the cluster, movement of RGs might be wanted, or happen automatically depending on the cluster configuration.

2.1.5 Live Partition Mobility automation

Performing a Live Partition Mobility (LPM) operation of a PowerHA node is supported. However, it is not without risk. Because of the unique nature of LPM, certain events, such as network loss, can be triggered during the operation. There have been some suggestions in the past, such as unmanage the node before performing LPM, but many users were unaware of them. As a result, the LPM automation integration feature was created.

Note: Previously, it was preferable to unmanage a node before performing LPM, but not many users were aware of this.

PowerHA scripts and checks during Live Partition Mobility

PowerHA provides scripts that are called during different phases of the LPM update notification mechanism. Here is an overview of the PowerHA operations that are performed at which phase:

- ▶ Check phase:
 - Verify that no other concurrent LPM is in progress in the cluster.
 - Verify that the cluster is in the stable state.
 - Verify network communications between cluster nodes.
- ▶ Pre-phase:
 - If set, or if IBM HyperSwap is used, stop cluster services in unmanaged mode.
 - On local node, and on peer node in two-node configuration:
 - Stop the Reliable Scalable Cluster Technology (RSCT) Dead Man Switch (DMS).
 - If HEARTBEAT_FREQUENCY_FOR_LPM is set, change the CAA node timeout.
 - If CAA deadman_mode at per-node level is a, set it to e.

Note: A deadman switch is an action that occurs when CAA detects that a node has become isolated in a multinode environment. This setting occurs when nodes are not communicating with each other through the network and the repository disk.

The AIX operating system can react differently depending on the deadman switch setting or the deadman_mode, which is tunable. The deadman switch mode can be set to either force a system shutdown or generate an Autonomic Health Advisor File System (AHAFS) event.

- Restrict SAN communications across nodes.
- ▶ Post phase:
 - Restart cluster services.
 - On local node, and on peer node in two-node configuration:
 - Restart the RSCT DMS.
 - Restore the CAA node timeout.
 - Restore the CAA deadman_mode.
 - Re-enable SAN communications across nodes.

The following new cluster heartbeat settings are associated with the auto handling of LPM:

► **Node Failure Detection Timeout during LPM**

If specified, this timeout value (in seconds) is used during an 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**

This specifies the action to be taken on the node during an LPM operation.

If unmanage is selected, the cluster services are stopped with the Unmanage Resource Groups option during the duration of the LPM operation. Otherwise, PowerHA SystemMirror continues to monitor the RGs and application availability.

As is common, these options can be set by using both SMIT and the **clmgr** command line. To change these options by using SMIT, run **smitty sysmirror** and select **Custom Cluster Configuration → Cluster Nodes and Networks → Manage the Cluster → Cluster Heartbeat Settings**, as shown in Figure 2-4.

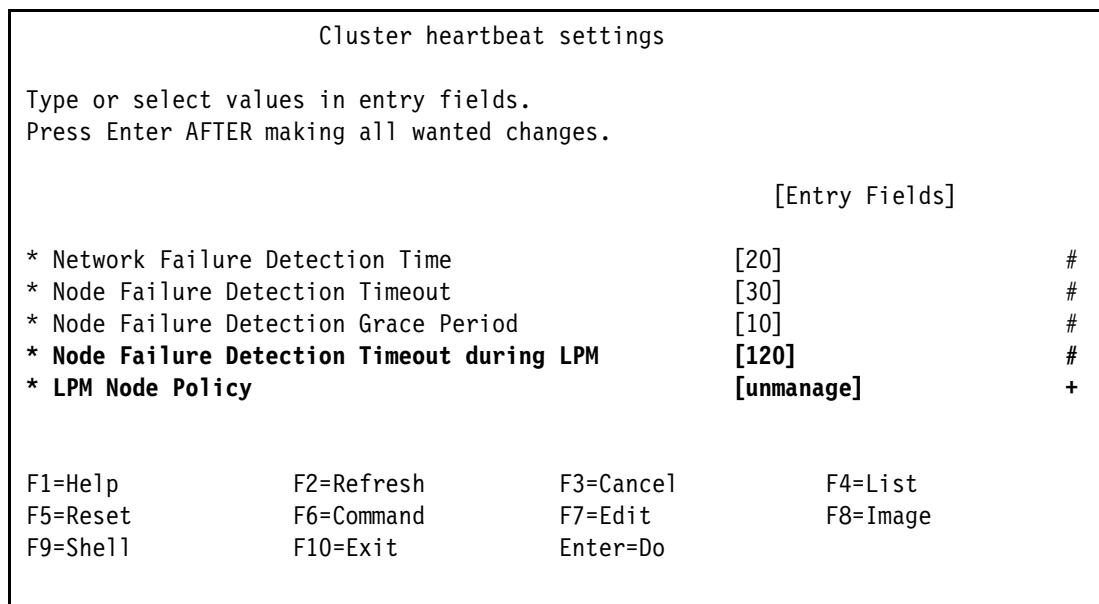


Figure 2-4 Enabling LPM integration

An example of using **clmgr** to check and change these settings is shown in Example 2-2.

Example 2-2 Using the clmgr command

```
[root@Jess] # clmgr query cluster |grep LPM
LPM_POLICY=""
HEARTBEAT_FREQUENCY_DURING_LPM="0"

[root@Jess] # clmgr modify cluster HEARTBEAT_FREQUENCY_DURING_LPM="120"
[root@Jess] # clmgr modify cluster LPM_POLICY=unmanage

[root@Jess] # clmgr query cluster |grep LPM
LPM_POLICY="120"
HEARTBEAT_FREQUENCY_DURING_LPM="unmanage"
```

Even with these new automated steps, there are still a few manual steps when using SAN communication:

- ▶ Before LPM

Verify that the `tme` attribute is set to yes on the target systems VIOS FC adapters.

- ▶ After LPM

Reestablish SAN communication between VIOS and the client LPAR through a virtual local area network (VLAN) 3358 adapter configuration.

No matter which method you chose to change these settings, the cluster must be synchronized for the change to take effect cluster-wide.

2.2 Cluster Aware AIX enhancements

In every new AIX level, CAA is also updated. The CAA version typically references the year in which it was released. For example, the AIX 7.2 CAA level is referenced as the 2015 version, also known as release 4. Table 2-1 shows the matching AIX and PowerHA levels to the CAA versions. This section continues with features that are new to CAA (2015/R4).

Table 2-1 IBM AIX and PowerHA levels to CAA versions

Internal version	External release	AIX level	PowerHA level
2011	R1	6.1.7/7.1.1	7.1.1
2012	R2	6.1.8/7.1.2	7.1.2
2013	R3	6.1.9/7.1.3	7.1.3
2015	R4	7.1.4/7.2.0	7.2
2016	R5	7.2.1	7.2.1

Note: The listed AIX and PowerHA levels are the preferred combinations to use all new features. However, these are not the only possible combinations.

2.2.1 Network failure detection tunable

PowerHA V7.1 had a fixed latency for network failure detection that was about 5 seconds. In PowerHA V7.2, the default is now 20 seconds. The tunable is named `network_fdt`.

Note: The `network_fdt tunable` is also available for PowerHA V7.1.3. To get it for PowerHA V7.1.3, you must open a PMR and request the “Tunable FDT interim fix bundle”.

The self-adjusting network heartbeat behavior (CAA), which was introduced with PowerHA V7.1.0, still exists and still is used. It has no impact to the network failure detection time.

For more information, see 4.1.5, “Network Failure Detection Time (FDT)” on page 80.

2.2.2 Built-in NETMON logic

NETMON logic was previously handled by RSCT. As it was difficult to keep both CAA and RSCT layers synchronized about the adapter state, NETMON logic was moved within the CAA layer.

The configuration file remains the same, namely `/usr/es/sbin/cluster/netmon.cf`. A `A !REQD` entry in the `netmon.cf` file indicates that special handling is needed that is different than the traditional `netmon` methods. For more information about `netmon.cf` file usage and formatting, see [IBM Knowledge Center](#).

2.2.3 Traffic stimulation for better interface failure detection

Multicast pings are sent to the *all hosts* multicast group just before marking an interface as down. This ping is distributed to the nodes within the subnet. Any node receiving this request replies (even the node is not a part of the cluster), and thus generates incoming traffic on the adapter. Multicast ping uses the address 224.0.0.1. All nodes register by default for this multicast group. Therefore, there is a good chance that some incoming traffic is generated by this method.

2.2.4 Monitoring /var usage

Starting with PowerHA V7.2.0 the `/var` file system is monitored by default. This monitoring is done by the `clconfd` subsystem. The following default values are used:

Threshold	75% (range 70 - 95)
Interval	15 min (range 5 - 30)

For more details, see 4.1.3, “Monitoring `/var` usage” on page 69.

2.2.5 New `lscluster` option `-g`

Starting with AIX V7.1 TL4 and AIX V7.2, there is an additional option for the CAA `lscluster` command.

The new option `-g` lists the interfaces that can potentially be used as a communication paths of CAA between the cluster nodes. For a more detailed description, see 4.1.4, “New `lscluster` option `-g`” on page 71.

2.2.6 CAA level added to the `lscluster -c` output

Starting with the AIX 7.2.1 the `lscluster -c` command also displays the CAA level. This command is useful if you need to know whether the new network failure detection tunable is supported by your installation by default. For more information, see 2.2.1, “Network failure detection tunable” on page 25 and 4.1.5, “Network Failure Detection Time (FDT)” on page 80.

This add-on was back level converted and is automatically included with AIX 7.1.4.2 or newer and with AIX 7.2.0.2 or newer.

2.3 Enhanced split-brain handling

Split-brain, also known as a partitioned cluster, refers to when all communications are lost between cluster nodes, yet the nodes are still running. PowerHA V7.2 supports new policies to quarantine a sick or dead active node. These policies help handle the cluster-split scenarios to ensure data protection during split scenarios. The following two new policies are supported:

- ▶ Disk fencing

Disk fencing uses the Small Computer System Interface (SCSI-3) Persistent Reservation mechanism to fence out the sick or dead node to block future writes from the sick node.

- ▶ Hardware Management Console (HMC)-based Active node shutdown

In the HMC-based Active node shutdown policy, standby node works with HMC to kill the previously active (sick) node, and only then starts the workload on the standby.

2.4 Resource Optimized High Availability failovers by using enterprise pools

PowerHA offers integrated support for dynamic LPAR (DLPAR), including using capacity on demand (CoD) resources since IBM HACMP 5.3. However, the type of CoD support was limited. Now, PowerHA V7.2 extends support to include Enterprise Pool CoD (EPCoD) and elastic CoD resources. Using these types of resources makes the solution less expensive to acquire and less expensive to own.

PowerHA SystemMirror 7.2.1 has the following enhancements compared to PowerHA SystemMirror 7.2.0:

- ▶ New ROHA tunable resource_allocation_order

You can use this to define the order in which hardware resources (CPU and memory) are allocated. Resources are released in reverse of the resource allocation order.

- ▶ New ROHA tunable ALWAYS_START_RG

You can use this tunable to do the failover (start the RG) even if there are not enough resources available (CPU or memory).

- ▶ Cross-HMC support

This supports the new Enterprise Pool capabilities.

PowerHA V7.2.0 has the following requirements:

- ▶ PowerHA SystemMirror 7.2, Standard Edition or Enterprise Edition

- ▶ One of the following AIX levels:

- AIX 6.1 TL09 SP5
- AIX 7.1 TL03 SP5
- AIX 7.1 TL4
- AIX 7.2 or later

- ▶ HMC requirement

- HMC V7.8 or later
- HMC must have a minimum of 2 GB of memory

- ▶ Hardware requirement for using Enterprise Pool CoD license
 - IBM POWER7+: 9117-MMD, 9179-MHD with FW780.10 or later
 - IBM POWER8®: 9119-MME, 9119-MHE with FW820 or later

Full details about using this integrated support can be found in Chapter 6, “Resource Optimized High Availability” on page 143.

2.5 Nondisruptive upgrades

PowerHA V7.2 enables nondisruptive cluster upgrades. It allows upgrades from PowerHA V7.1.3 to V7.2 without having to roll over the workload from one node to another as part of the migration. The key requirement is that the existing AIX/CAA levels must be either 6.1.9 or 7.1.3. More information about performing nondisruptive upgrades can be found in 5.2.5, “Nondisruptive upgrade from PowerHA V7.1.3” on page 124.

Tip: A demonstration of performing a nondisruptive upgrade can be found in this [YouTube video](#).

2.6 Geographic Logical Volume Manager wizard

PowerHA 6.1 introduced the first two-site GLVM configuration. However, it was limited to only synchronous implementations and still required some manual steps. PowerHA V7.2 introduces an enhanced GLVM wizard that involves fewer steps but also includes support for asynchronous implementations. More details can be found in Chapter 7, “Geographic Logical Volume Manager configuration assistant” on page 245.

2.7 New option for starting PowerHA by using clmgr

Starting with PowerHA V7.2.1, you can use an additional management option to start the cluster. The new argument for the option is named **delayed**:

```
clmgr online cluster manage=delayed
```

Note: This new option was backported to PowerHA V7.2 and V7.3.1.

At the time of writing, the only way to obtain the new option is to open a PMR and ask for an interim fix for the defect 100862, or ask for an interim fix for APAR IV90262.

Since PowerHA V7.1, the dependency “Start After” is available.

2.8 Graphical user interface

PowerHA V7.2.1 contains a new GUI. The focus for its design is:

- ▶ Quick and easy status revelation
- ▶ Easier way to view events
- ▶ Easier way to view logs

The new GUI has several features. The following list is just a brief overview. For a detailed description, see Chapter 9, “IBM PowerHA SystemMirror User Interface” on page 303.

- ▶ Visually display of relationships among resources.
- ▶ Systems with the highest severity problems are highly visible.
- ▶ Visualizes the health status for each resource.
- ▶ Formatted events are easy to scan.
- ▶ Visually distinguish critical, warning, and maintenance events.
- ▶ Organized by day and time.
- ▶ Can filter and search for specific types of events.
- ▶ You can see the progression of events by using the timeline.
- ▶ You can zoom in to see details or zoom out to see health over time.
- ▶ You can search for an event in the event log.
- ▶ If your system has internet access you can open a browser to the PowerHA IBM Knowledge Center.



Planning considerations

This chapter provides information to help you plan the implementation of IBM PowerHA SystemMirror.

This chapter covers the following topics:

- ▶ Introduction
- ▶ Cluster Aware AIX repository disk
- ▶ Cluster Aware AIX (CAA) tunables
- ▶ Important considerations for virtual input/output server
- ▶ Network considerations
- ▶ Network File System tie breaker

3.1 Introduction

There are many different ways to build a highly available environment. This chapter describes a small subset.

3.1.1 Mirrored architecture

In a mirrored architecture, you have identical or nearly identical physical components in each part of the data center. You can have this type of setup in a single room (although this is not recommended), in different rooms in the same building, or in different buildings. The distance between each part can be between few kilometers or several kilometers (or up to 50+ km, depending on the application latency requirements).

Figure 3-1 shows a high-level diagram of a cluster. In this example, there are two networks, two managed systems, two Virtual Input/Output Servers (VIOS) per managed system, and two storage subsystems. This example also uses the Logical Volume Manager (LVM) mirroring for maintaining a complete copy of data within each storage subsystem.

This example also has a logical unit number (LUN) for the Cluster Aware AIX (CAA) repository disk on each storage subsystem. For details about how to set up the CAA repository disk, see 3.2, “Cluster Aware AIX repository disk” on page 36.

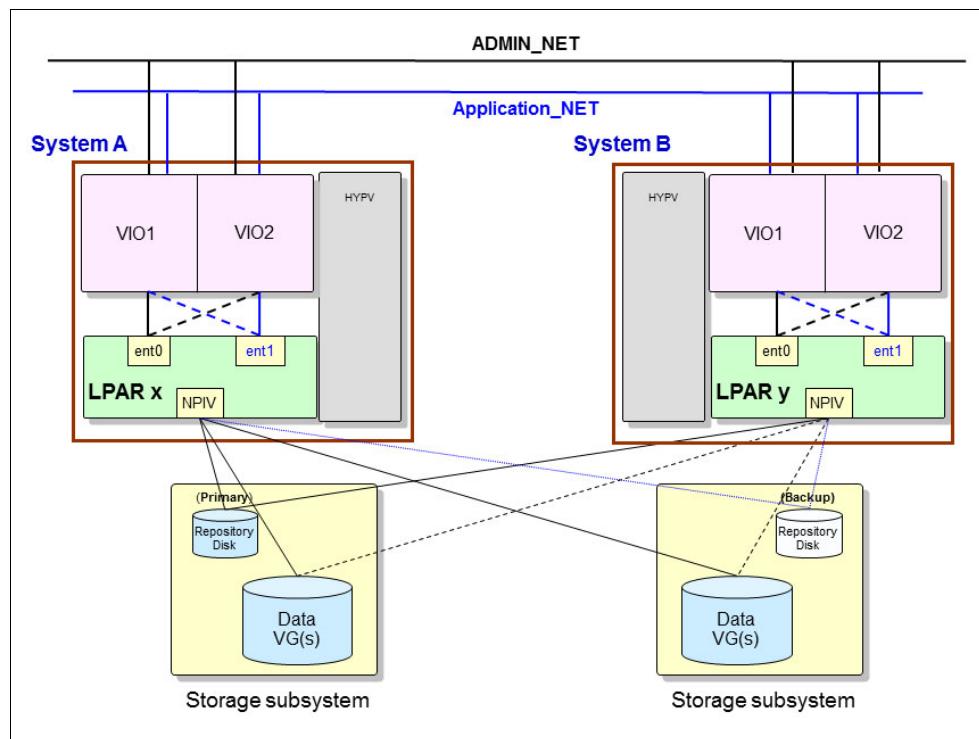


Figure 3-1 Cluster with multiple storage subsystems

3.1.2 Single storage architecture

In a single storage architecture, the storage is shared by both the primary and backup logical partition (LPAR). This solution can be used when there are lower availability requirements for the data, and is not uncommon when the LPARs are in the same location.

When it is possible to use a storage-based mirroring feature such as IBM SAN Volume Controller or a SAN Volume Controller stretched cluster, the layout look, from a physical point of view, identical or nearly identical to the mirrored architecture that is described in 3.1.1, “Mirrored architecture” on page 32. However, from an AIX and cluster point of view, it is a single storage architecture because it is aware of only a single set of LUNs. For more information about the layout in a SAN Volume Controller stretched cluster, see 3.1.3, “Stretched cluster” on page 33.

Figure 3-2 shows such a layout from a logical point of view.

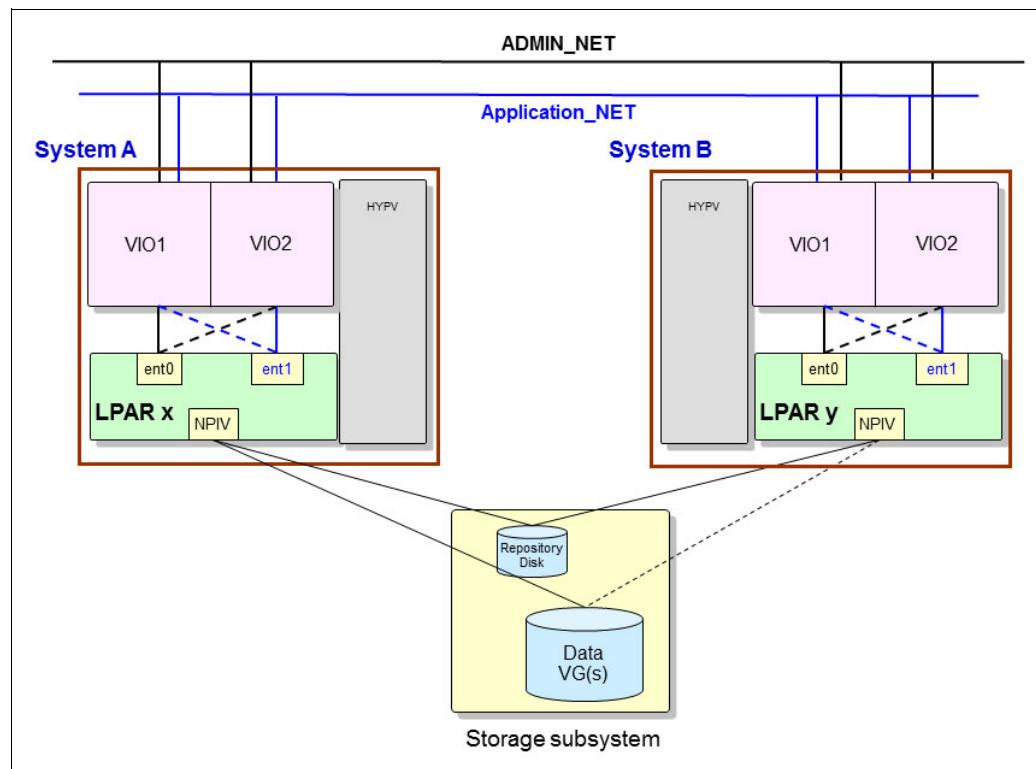


Figure 3-2 Cluster with single storage subsystem

3.1.3 Stretched cluster

A stretched cluster involves separating the cluster nodes into *sites*. A site can be in a different building within a campus or separated by a few kilometers in terms of distance. In this configuration, there is a storage area network (SAN) that spans the sites and storage can be presented across sites.

As with any multi-site cluster, Transmission Control Protocol/Internet Protocol (TCP/IP) communications are essential. Multiple links and routes are suggested such that a single network component or path failure can be incurred and communications between sites still be maintained.

Another main concern is having redundant storage and verifying that the data within the storage devices is synchronized across sites. The following section presents a method for synchronizing the shared data.

SAN Volume Controller in a stretched configuration

The SAN Volume Controller can be configured in a *stretched* configuration. In the stretched configuration, the SAN Volume Controller presents two storage devices that are separated by distance but look as though it is a single SAN Volume Controller device. The SAN Volume Controller itself keeps the data between the sites consistent through its disk mirroring technology.

The SAN Volume Controller in a stretched configuration allows the PowerHA cluster to provide continuous availability of the storage LUNs even if there is a single component failure anywhere in the storage environment. With this combination, the behavior of the cluster is similar in terms of function and failure scenarios in a local cluster (Figure 3-3).

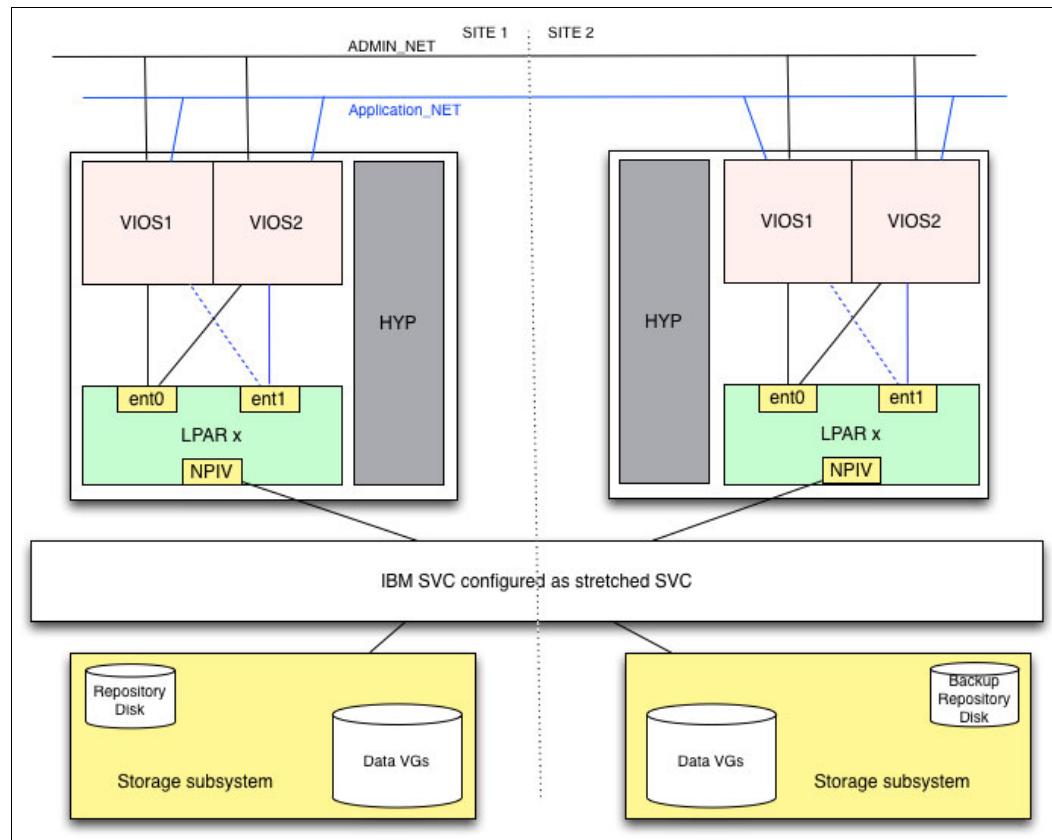


Figure 3-3 SAN Volume Controller stretched configuration

3.1.4 Linked cluster

A linked cluster is another type of cluster that involves multiple sites. In this case, there is no SAN across sites because the distance between sites is often too far or the expense is too great. In this configuration, the repository disk is mirrored across the Internet Protocol network. Each site has its own copy of the repository disk and PowerHA keeps those disks synchronized.

TCP/IP communications are essential, and multiple links and routes are suggested such that a single network component or path failure can be incurred and communications between sites still be maintained.

For more information about linked clusters see *IBM PowerHA SystemMirror 7.1.2 Enterprise Edition for AIX*, SG24-8106.

IBM supported storage that uses copy services

There are several IBM supported storage devices with copy services capabilities. For the following example, we use one of these devices, the SAN Volume Controller. SAN Volume Controller can replicate data across long distances with the SAN Volume Controller copy services functions. The data can be replicated in synchronous or asynchronous modes where synchronous provides the most up-to-date data redundancy.

For data replication in synchronous mode where both writes must complete before acknowledgment is sent to the application, the distance can greatly affect application performance. Synchronous mode is commonly used for 100 kilometers or less. Asynchronous modes are often used for distances over 100 km. However, these are common baseline recommendations.

If there is a failure that requires moving the workload to the remaining site, PowerHA interacts directly with the storage to switch the direction of the replication. PowerHA then makes the LUNs read/write capable and varies on the appropriate volume groups (VGs) to activate the application on the remaining site.

An example of this concept is shown in Figure 3-4.

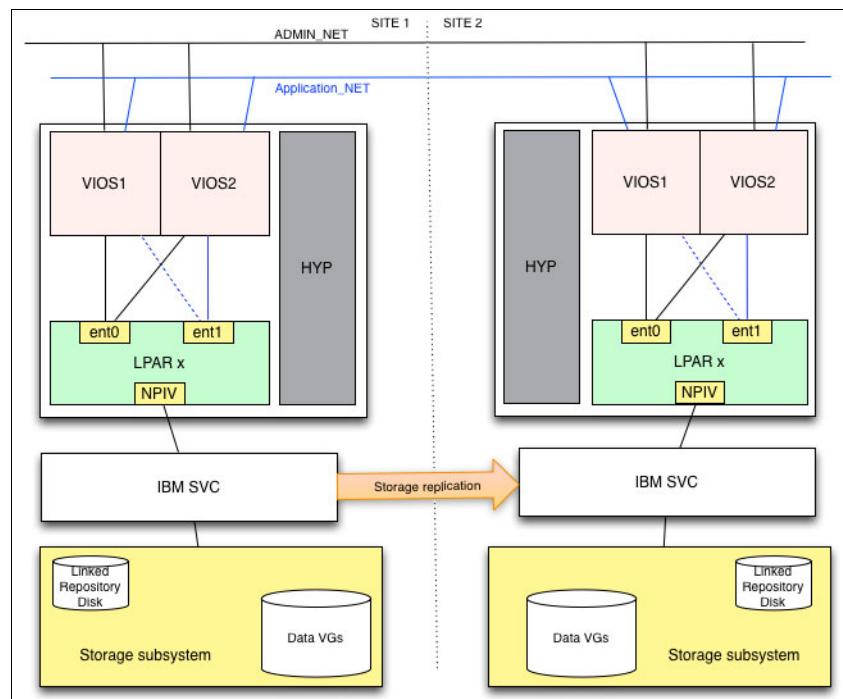


Figure 3-4 PowerHA and SAN Volume Controller storage replication

3.2 Cluster Aware AIX repository disk

CAA uses a shared disk to store its 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 is available to all nodes that are part of the cluster. This disk cannot be used for application storage or any other purpose.

The amount of configuration information that is stored on this repository disk directly depends on the number of cluster entities, such as shared disks, number of nodes, and number of adapters in the environment. You must ensure that you have enough space for the following components when you determine the size of a repository disk:

- ▶ Node-to-node communication
- ▶ CAA Cluster topology management
- ▶ All migration processes

The preferred size for the repository disk in a two-node cluster is 1 GB.

3.2.1 Preparing for a Cluster Aware AIX repository disk

A common way to protect the repository disk is to use storage-based mirroring or RAID. One example is the one that is described in 3.1.2, “Single storage architecture” on page 32. In this example, you must make sure that the LUN for the CAA repository disk is visible on all cluster nodes, and that there is a physical volume identifier (PVID) that is assigned to it.

If you have a multi-storage environment, such as the one that is described in 3.1.1, “Mirrored architecture” on page 32, then see 3.2.2, “Cluster Aware AIX with multiple storage devices” on page 36.

Important: The repository is *not* supported for mirroring by LVM.

3.2.2 Cluster Aware AIX with multiple storage devices

The description here is related to the architecture that is described in 3.1.1, “Mirrored architecture” on page 32. This example uses one backup CAA repository disk. The maximum number of backup disks that you can define is six.

If you plan to use one or more disks, which can potentially be used as backup disks for the CAA repository, it is a preferred practice to rename the disks, as described in “Renaming the hdisk” on page 38. However, this is not possible in all cases.

Important: Third-party MultiPath I/O (MPIO) management software, such as EMC PowerPath, uses disk mapping to manage multi-paths. These software programs typically have a disk definition at a higher level, and path-specific disks underneath. Also, these software programs typically use special naming conventions.

Renaming these types of disks by using the AIX `rendev` command can confuse the third-party MPIO software and create disk-related issues. For more information about any disk renaming tool that is available as part of the vendor’s software kit, see your vendor documentation.

The examples in this section mainly use **smitty sysmirror**. However, using the **clmgr** command line can be faster, but it can be hard to use by a novice. The examples use the **clmgr** command where it makes sense or where it is the only option.

Using the standard hdisk name

A current drawback of having multiple LUNs that can be used as repository disk is that they are not clearly identified as such by the **lspv** output. In this example, hdisk3 and hdisk4 are the LUNs that are prepared for the primary and backup CAA repository disks. Therefore, hdisk1 and hdisk2 are used for the application. Example 3-1 shows the output of the **lspv** command before starting the configuration.

Example 3-1 The lspv output before configuring Cluster Aware AIX

```
# lspv
hdisk0      00f71e6a059e7e1a          rootvg      active
hdisk1      00c3f55e34ff43cc          None        None
hdisk2      00c3f55e34ff433d          None        None
hdisk3      00f747c9b40ebfa5          None        None
hdisk4      00f747c9b476a148          None        None
hdisk5      00f71e6a059e701b          rootvg      active
#
```

After selecting hdisk3 as the CAA repository disk, synchronizing and creating the cluster, and creating the application VG, you get the output that is listed in Example 3-2. The commands that are used for this example are the following ones:

```
clmgr add cluster test_c1
clmgr sync cluster
```

As shown in Example 3-2, the problem is that the **lspv** command does not show that hdisk4 is reserved as the backup disk for the CAA repository.

Example 3-2 The lspv output after configuring Cluster Aware AIX

```
# lspv
hdisk0      00f71e6a059e7e1a          rootvg      active
hdisk1      00c3f55e34ff43cc          testvg     testvg
hdisk2      00c3f55e34ff433d          testvg     testvg
hdisk3      00f747c9b40ebfa5          caavg_private active
hdisk4      00f747c9b476a148          None        None
hdisk5      00f71e6a059e701b          rootvg      active
#
```

To see which disk is reserved as a backup disk, use the **clmgr -v query repository** command or the **odmget HACMPsircol** command. Example 3-3 shows the output of the **clmgr** command, and Example 3-4 on page 38 shows the output of the **odmget** command.

Example 3-3 The clmgr -v query repository output

```
# clmgr -v query repository
NAME="hdisk3"
NODE="c2n1"
PVID="00f747c9b40ebfa5"
UUID="12d1d9a1-916a-ceb2-235d-8c2277f53d06"
BACKUP="0"
TYPE="mpioosdisk"
DESCRIPTION="MPIO IBM 2076 FC Disk"
```

```
SIZE="1024"
AVAILABLE="512"
CONCURRENT="true"
ENHANCED_CONCURRENT_MODE="true"
STATUS="UP"

NAME="hdisk4"
NODE="c2n1"
PVID="00f747c9b476a148"
UUID="c961dda2-f5e6-58da-934e-7878cfbe199f"
BACKUP="1"
TYPE="mpioosdisk"
DESCRIPTION="MPIO IBM 2076 FC Disk"
SIZE="1024"
AVAILABLE="95808"
CONCURRENT="true"
ENHANCED_CONCURRENT_MODE="true"
STATUS="BACKUP"#
```

As you can see in the **c1mgr** output, you can directly see the hdisk name. The **odmget** command output (Example 3-4) lists the PVIDs.

Example 3-4 The odmget HACMPsircol output

```
# odmget HACMPsircol
HACMPsircol:
    name = "c2n1_cluster_sircol"
    id = 0
    uuid = "0"
    ip_address = ""
    repository = "00f747c9b40ebfa5"
    backup_repository = "00f747c9b476a148"#

```

Renaming the hdisk

To get around the issues that are mentioned in “Using the standard hdisk name” on page 37, rename the hdisks. The advantage is that it is much easier to see which disk is reserved as the CAA repository disk.

There are some points to consider:

- ▶ Generally, you can use any name, but if it gets too long you can experience some administration issues.
- ▶ The name must be unique.
- ▶ It is preferable not to have the string “disk” as part of the name. There might be some scripts or tools that can search for the string “disk”.
- ▶ You must manually rename the hdisks on all cluster nodes.

Important: Third-party MultiPath I/O (MPIO) management software, such as EMC PowerPath, uses disk mapping to manage multi-paths. These software programs typically have a disk definition at a higher level, and path-specific disks underneath. Also, these software programs typically use special naming conventions.

Renaming these types of disks by using the AIX `rendev` command can confuse the third-party MPIO software and create disk-related issues. For more information about any disk renaming tool that is available as part of the vendor's software kit, see your vendor documentation.

Using a long name

First, we test by using a longer and more descriptive name. Example 3-5 shows the output of the `lspv` command before we started.

Example 3-5 The lspv output before using rendev

```
# lspv
hdisk0      00f71e6a059e7e1a          rootvg      active
hdisk1      00c3f55e34ff43cc        None
hdisk2      00c3f55e34ff433d        None
hdisk3      00f747c9b40ebfa5        None
hdisk4      00f747c9b476a148        None
hdisk5      00f71e6a059e701b          rootvg      active
#
```

Initially we decide to use a longer name (`caa_reposX`). Example 3-6 shows what we did and what the `lspv` command output looks like afterward.

Important: Remember to do the same on all cluster nodes.

Example 3-6 The lspv output after using rendev (using a long name)

```
#rendev -l hdisk3 -n caa_repos0
#rendev -l hdisk4 -n caa_repos1
# lspv
hdisk0      00f71e6a059e7e1a          rootvg      active
hdisk1      00c3f55e34ff43cc        None
hdisk2      00c3f55e34ff433d        None
caa_repos0  00f747c9b40ebfa5        None
caa_repos1  00f747c9b476a148        None
hdisk5      00f71e6a059e701b          rootvg      active
#
```

Next, configure the cluster by using the SMIT. Using F4 to select the CAA repository disk returns the panel that is shown in Figure 3-5. As you can see, only the first part of the name is displayed. So, the only way to obtain which is the disk is to check for the PVID.

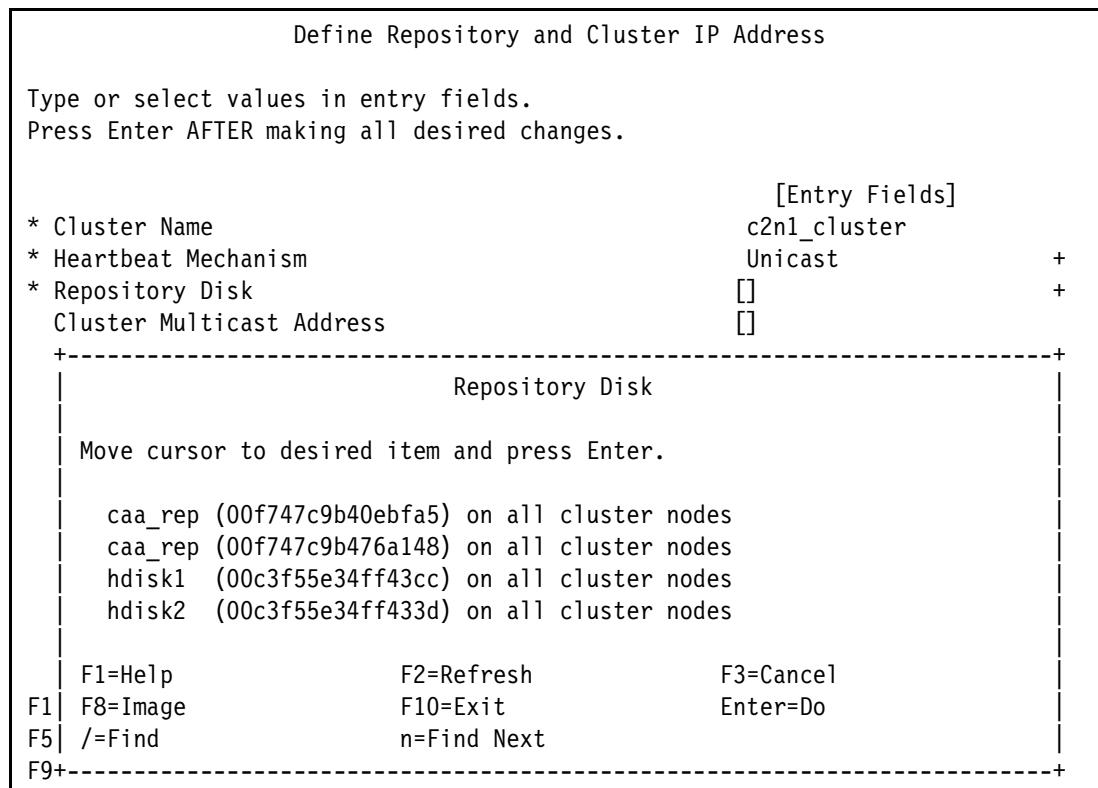


Figure 3-5 SMIT panel that uses long repository disk names

Using a short name

In this case, a short name means a name with a maximum of seven characters. We use the same starting point, as listed in Example 3-5 on page 39. This time, we decide to use a shorter name (caa_rX). Example 3-7 shows what we did and what the `lspv` command output looks like afterward.

Important: Remember to do the same on all cluster nodes.

Example 3-7 The `lspv` output after using rendev (using a short name)

```
#rendev -l hdisk3 -n caa_r0
#rendev -l hdisk4 -n caa_r1
# lspv
hdisk0      00f71e6a059e7e1a          rootvg      active
hdisk1      00c3f55e34ff43cc          None
hdisk2      00c3f55e34ff433d          None
caa_r0      00f747c9b40ebfa5        None
caa_r1      00f747c9b476a148        None
hdisk5      00f71e6a059e701b          rootvg      active
#
```

Now, we start configuring the cluster by using SMIT. Using F4 to select the CAA repository disk returns the panel that is shown in Figure 3-6. As you can see, the full name now is displayed.

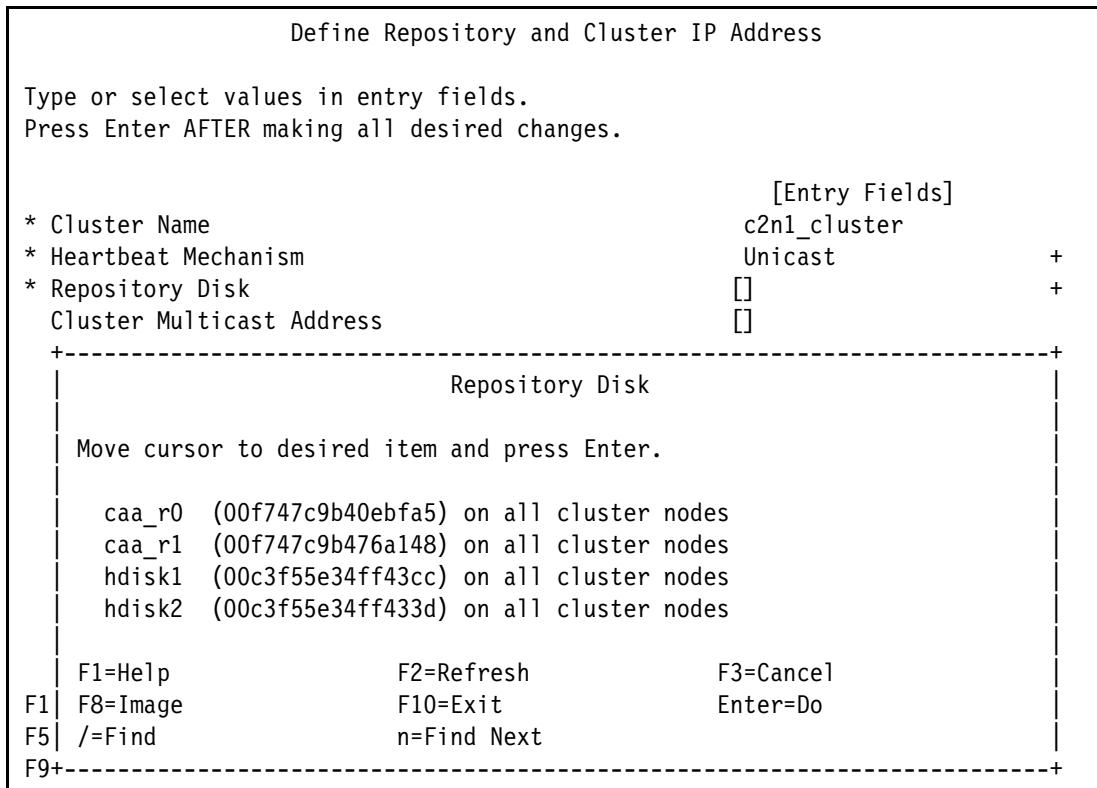


Figure 3-6 SMIT panel that uses short names

3.3 Cluster Aware AIX (CAA) tunables

This section describes some CAA tunables and what they are used for. Example 3-8 shows the list of the CAA tunables with IBM AIX V7.2.0.0 and IBM PowerHA V7.2.0. Newer versions can have more tunables, different defaults, or both.

Attention: Do not change any of these tunables without the explicit permission of IBM technical support.

In general, you must never modify these values because these values are modified and managed by PowerHA.

Example 3-8 List of Cluster Aware AIX tunables

```
# clctr -tune -a
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).communication_mode = u
    ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).config_timeout = 240
        ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).deadman_mode = a
            ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).link_timeout = 30000
                ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).local_merge_policy = m
                    ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).network_fdt = 20000
                        ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).no_if_traffic_monitor = 0
```

```
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).node_down_delay = 10000
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).node_timeout = 30000
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).packet_ttl = 32
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).remote_hb_factor = 1
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).repos_mode = e
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).site_merge_policy = p
#
```

3.3.1 CAA network monitoring

By default CAA monitors for incoming IP traffic and physical link status. In extremely rare situations mainly when you are using physical Ethernet adapters, there might be the need to disable the monitoring for incoming IP traffic. As mentioned before, this is done in PowerHA and not in CAA.

In general, do not change the monitoring values unless you are instructed by IBM.

Note: Starting with PowerHA V7.2, the *traffic stimulation* feature makes this flag obsolete.

For your information some details are listed in this section. To list and change this setting, use the **c1mgr** command. You must keep in mind that this change affects all IP networks.

Attention: Do not change this tunable without the explicit permission of IBM technical support.

- ▶ To list the current setting:

```
c1mgr -a MONITOR_INTERFACE query cluster
```

The default is MONITOR_INTERFACE=enable

- ▶ To disable it:

```
c1mgr -f modify cluster MONITOR_INTERFACE=disable
```

In PowerHA V7.1.3, this setting has been used to get around some netmon.cf related issues. In PowerHA V7.2.0 and later most of these issues do not exist any longer. Hence it is advised to change it back to the default (MONITOR_INTERFACE=enable) and test it.

- ▶ To enable it:

```
c1mgr -f modify cluster MONITOR_INTERFACE=enable
```

In the **c1ctrl -tune -a** command output, this is listed as no_if_traffic_monitor. The value 0 means enabled and the value 1 means disabled.

3.3.2 Network failure detection time

Starting with PowerHA V7.2.0, the network failure detection time can be defined. The default is 20 seconds. In the **c1ctrl** command output, it is listed as network_fdt.

To change this use **c1mgr** or **smit** commands.

3.4 Important considerations for virtual input/output server

This section lists some new features of AIX and Virtual I/O Server (VIOS) that help to increase overall availability, and are specially suggested to use for PowerHA environments.

3.4.1 Using poll_uplink

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 VIOS.
- ▶ The LPAR must be at AIX 7.1 TL3, or AIX 6.1 TL9 or later.
- ▶ The option **poll_uplink** must be set on the LPAR on the virtual entX interfaces.

The option **poll_uplink** can be defined directly on the virtual interface if you are using shared Ethernet adapter (SEA) failover or the Etherchannel device that points to the virtual interfaces. To enable **poll_uplink**, use the following command:

```
chdev -l entX -a poll_uplink=yes -P
```

Important: You must restart the LPAR to activate **poll_uplink**.

There are no additional changes to PowerHA and CAA needed. The information about the virtual link status is automatically detected by CAA. There is no need to change the MONITOR_INTERFACE setting. Details about MONITOR_INTERFACE are described in 3.3.1, “CAA network monitoring” on page 42.

Figure 3-7 shows an overview of how the option works. In production environments, you normally have at least two physical interfaces on the VIOS, and you can also use a dual-VIOS setup. In a multiple physical interface environment, the virtual link is reported as down only when all physical connections on the VIOS for this SEA are down.

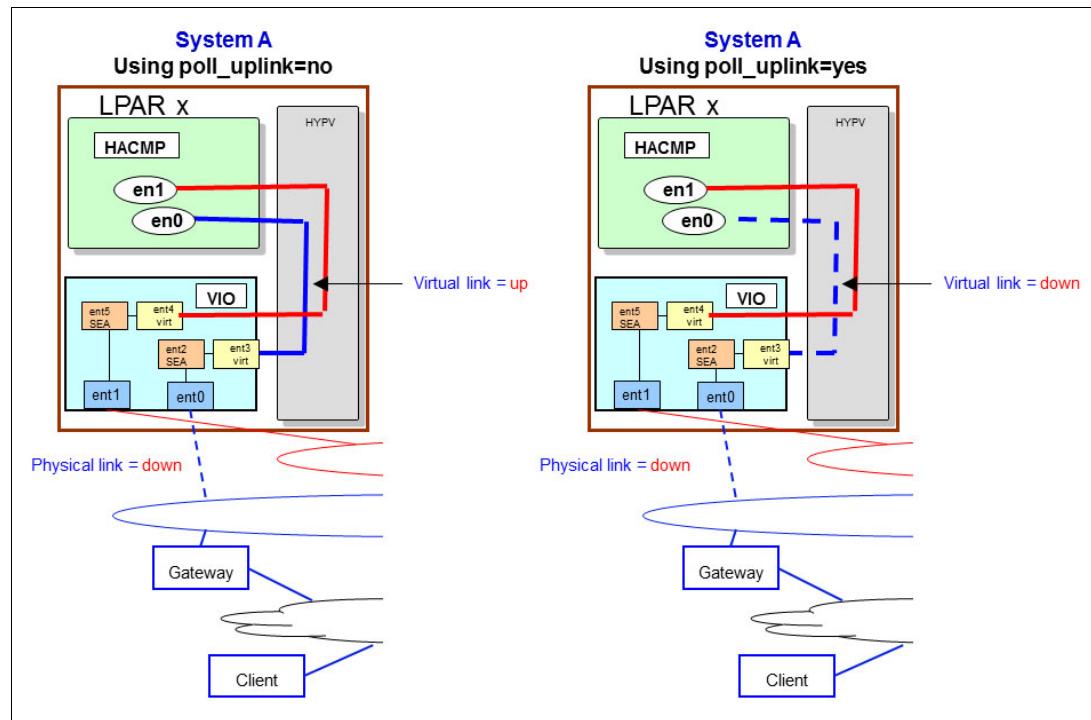


Figure 3-7 Using poll_uplink

The following settings are possible for **poll_uplink**:

- ▶ **poll_uplink** (yes, no)
- ▶ **poll_uplink_int** (100 milliseconds (ms) - 5000 ms)

To display the settings, use the **lsattr -El entX** command. Example 3-9 shows the default settings for **poll_uplink**.

Example 3-9 The lsattr details for poll_uplink

```
# lsdev -Cc Adapter | grep ^ent
ent0  Available      Virtual I/O Ethernet Adapter (1-lan)
ent1  Available      Virtual I/O Ethernet Adapter (1-lan)
# lsattr -El ent0 | grep "poll_up"
poll_uplink    no          Enable Uplink Polling           True
poll_uplink_int 1000       Time interval for Uplink Polling  True
#
```

If your LPAR is at least AIX 7.1 TL3 SP3, or AIX 6.1 TL9 SP3 or later, you can use the **entstat** command to check for the **poll_uplink** status and if it is enabled. Example 3-10 shows an excerpt of the **entstat** command output in an LPAR where **poll_uplink** is not enabled (set to no).

Example 3-10 Using poll_uplink=no

```
# entstat -d ent0
-----
ETHERNET STATISTICS (en0) :
Device Type: Virtual I/O Ethernet Adapter (1-1an)
...
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
...
#
```

Compared to Example 3-10, Example 3-11 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 3-11 Using poll_uplink=yes when physical link is up

```
# entstat -d ent0
-----
ETHERNET STATISTICS (en0) :
Device Type: Virtual I/O Ethernet Adapter (1-1an)
...
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
...
#
```

When all physical links on the VIOS are down, then the output that is listed in Example 3-12 is displayed. The text PHYS_LINK_UP no longer displays, and the Bridge Status changes from Up to Unknown.

Example 3-12 Using poll_uplink=yes when physical link is down

```
# entstat -d ent0
-----
ETHERNET STATISTICS (en0) :
Device Type: Virtual I/O Ethernet Adapter (1-1an)
...
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
...
#
```

3.4.2 Advantages for PowerHA when poll_uplink is used

In PowerHA V7, the network down detection is performed by CAA. CAA by default checks for IP traffic and for the link status of an interface. Therefore, using **poll_uplink** is advised for PowerHA LPARs, which helps the system to make a better decision when a given interface is up or down.

The network down failure detection is much faster if **poll_uplink** is used and the link is marked as down.

3.5 Network considerations

This section focuses on the network considerations from a PowerHA point of view only. From this point of view, it does not matter if you have virtual or physical network devices.

3.5.1 Dual-adapter networks

This type of network has historically been the most common since the inception of PowerHA. However, starting with virtualization, this type was replaced with single adapter network solutions. But, the “single” adapter is redundant by using Etherchannel and often combined with SEA.

In PowerHA V7.1, this solution can still be used, but it is not recommended. The cross-adapter checking logic is not implemented in PowerHA V7. The advantage of not having this feature is that PowerHA V7.1 and later versions do not require that the IP source route is enabled.

When using a dual-adapter network in PowerHA V7.1 or later, you must also use the netmon.cf file in a similar way as that for a single adapter layout. In this case, the netmon.cf file must have a path for all potential enX interfaces defined.

3.5.2 Single-adapter network

When we describe a single-adapter network, it is from a PowerHA point of view. In a highly available environment, you must always have redundant ways to access the network. This is commonly done today by using SEA failover or Etherchannel Link Aggregation or node initialization block (NIB). The Etherchannel NIB-based solution can be used in both scenarios, by using virtual adapters or physical adapters. The Etherchannel Link Aggregation-based solution can be used only if you have direct-attached adapters.

Note: With a *single adapter*, you use the SEA failover or the Etherchannel failover.

This setup eases the setup from a TCP/IP point of view, and it also reduces the content of the netmon.cf file. But netmon.cf must still be used.

3.5.3 The netmon.cf file

In PowerHA V7.2 the netmon.cf file is now used by CAA. Before PowerHA V7.2 it was used by RSCT. There are now (starting with PowerHA V7.2) different rules for the netmon.cf content as listed in Table 3-1.

Table 3-1 netmon.cf changes

PowerHA V7.1.x	PowerHA V7.2.x
RSCT based	CAA based
Up to 30 lines by interface	Up to 5 lines by interface - Uses last 5 lines if more than 5 lines are defined
Gets checked every 4 seconds	Gets checked every 10 minutes - To force a re-read use the command clusterconf
Runs continuously	Run only if CAA detects an outage

Independent from the PowerHA version used, if possible, you should have more than one address defined (by interface). It is not recommended to use the gateway address. Modern gateways start dropping ICMP packages if there is high workload. ICMP packages sent to an address behind the gateway are not affected by this behavior. Although the network team can decide to drop all ICMP packets addressed to the gateway.

3.6 Network File System tie breaker

This section describes the Network File System (NFS) tie breaker.

3.6.1 Introduction and concepts

The NFS tie-breaker function represents an extension of the previously introduced disk tie-breaker feature that relied on a Small Computer System Interface (SCSI) disk that is

accessible to all nodes in a PowerHA cluster. The differences between the protocols that are used for accessing the tie-breaker (SCSI disk or NFS-mounted file) favor the NFS-based solution for linked clusters.

Split-brain situation

A cluster split-brain event can occur when a group of nodes cannot communicate with the remaining nodes in a cluster. For example, in a two-site linked cluster, a split occurs if all communication links between the two sites fail. Depending on the communication network topology and the location of the interruption, a cluster split event splits the cluster into two (or more) partitions, each of them containing one or more cluster nodes. The resulting situation is commonly referred to as a *split-brain situation*.

In a split-brain situation, the two partitions have no knowledge of each other's status, each of them considering the other as being offline. As a consequence, each partition tries to bring online the other partition's resource groups (RGs), thus generating a high risk of data corruption on all shared disks. To prevent a split-brain situation, and subsequent potential data corruption, split and merge policies are available to be configured.

Tie breaker

The tie-breaker feature uses a tie-breaker resource to choose a surviving partition that continues to operate when a cluster split-brain event occurs. This feature prevents data corruption on the shared cluster disks. The tie breaker is identified either as a SCSI disk or an NFS-mounted file that must be accessible, under normal conditions, to all nodes in the cluster.

Split policy

When a split-brain situation occurs, each partition attempts to acquire the tie breaker by placing a lock on the tie-breaker disk or on the NFS file. The partition that first locks the SCSI disk or reserves the NFS file *wins*, and the other *loses*.

All nodes in the winning partition continue to process cluster events, and all nodes in the losing partition attempt to recover according to the defined split and merge action plan. This plan most often implies either the restart of the cluster nodes, or merely the restart of cluster services on those nodes.

Merge policy

There are situations in which, depending on the cluster split-brain policy, the cluster can have two partitions that run independent of each other. However, most often, it is a preferred practice to configure a merge policy that allows the partitions to operate together again after communications are restored between them.

In this second approach, when partitions that were part of the cluster are brought back online after the communication failure, they must be able to communicate with the partition that owns the tie-breaker disk or NFS file. If a partition that is brought back online cannot communicate with the tie-breaker disk or the NFS file, it does not join the cluster. The tie-breaker disk or NFS file is released when all nodes in the configuration rejoin the cluster.

The merge policy configuration, in this case an NFS-based tie breaker, must be of the same type as that for the split policy.

3.6.2 Test environment setup

The lab environment that we use to test the NFS tie-breaker function consists of a two-site linked cluster, each site having a single node with a common NFS-mounted resource, as shown in Figure 3-8.

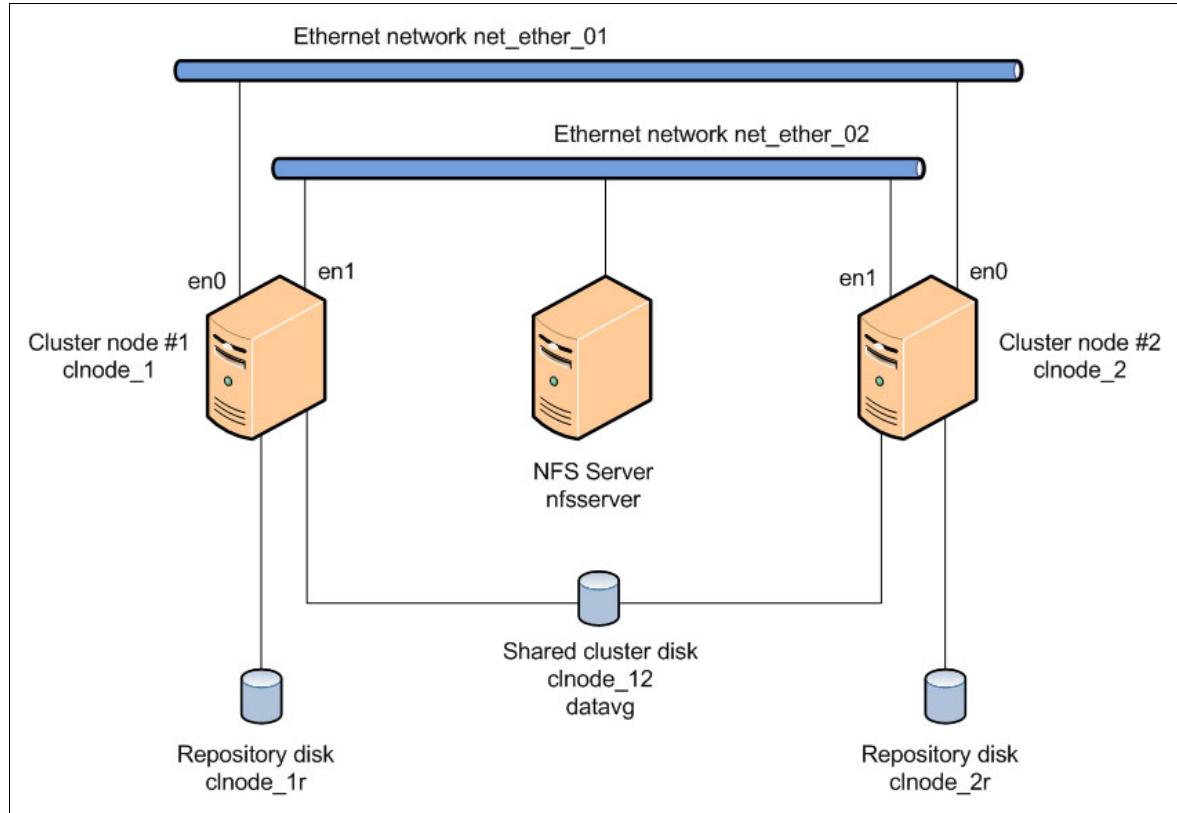


Figure 3-8 NFS tie-breaker test environment

Because the goal was to test the NFS tie-breaker function as a method for handling split-brain situations, the additional local nodes in a linked multisite cluster were considered irrelevant, and therefore not included in the test setup. Each node had its own cluster repository disk (`cnode_1r` and `cnode_2r`), and both nodes shared a common cluster disk (`cnode_12`, which is the one that must be protected from data corruption that is caused by a split-brain situation), as shown in Example 3-13.

Example 3-13 List of physical volumes on both cluster nodes

```
cnode_1:/# lspv
cnode_1r      00f6f5d0f8c9fbf4          caavg_private   active
cnode_12     00f6f5d0f8ca34ec          datavg         concurrent
hdisk0       00f6f5d09570f170          rootvg        active
cnode_1:/#
cnode_2:/# lspv
cnode_2r      00f6f5d0f8ceed1a          caavg_private   active
cnode_12     00f6f5d0f8ca34ec          datavg         concurrent
hdisk0       00f6f5d09570f31b          rootvg        active
cnode_2:/#
```

To allow greater flexibility for our test scenarios, we chose to use different network adapters for the production traffic and the connectivity to the shared NFS resource. The network setup of the two nodes is shown in Example 3-14.

Example 3-14 Network settings for both cluster nodes

```
clnode_1:/# netstat -in | egrep "Name|en"
Name  Mtu   Network      Address          Ipkts  Ierrs    Opkts  Oerrs  Coll
en0   1500  link#2     ee.af.e.90.ca.2  533916   0  566524   0   0
en0   1500  192.168.100 192.168.100.50  533916   0  566524   0   0
en0   1500  192.168.100 192.168.100.51  533916   0  566524   0   0
en1   1500  link#3     ee.af.e.90.ca.3  388778   0  457776   0   0
en1   1500  10          10.0.0.1        388778   0  457776   0   0
clnode_1:/#
clnode_2:/# netstat -in | egrep "Name|en"
Name  Mtu   Network      Address          Ipkts  Ierrs    Opkts  Oerrs  Coll
en0   1500  link#2     ee.af.7.e3.9a.2  391379   0  278953   0   0
en0   1500  192.168.100 192.168.100.52  391379   0  278953   0   0
en1   1500  link#3     ee.af.7.e3.9a.3  385787   0  350121   0   0
en1   1500  10          10.0.0.2        385787   0  350121   0   0
clnode_2:/#
```

During the setup of the cluster, the NFS communication network, with the en1 network adapters in Example 3-14, was discovered and automatically added to the cluster configuration as a heartbeat network, as net_ether_02. However, we manually removed it afterward to prevent interference with the NFS tie-breaker tests. Therefore, the cluster eventually had only one heartbeat network: net_ether_01.

The final cluster topology was reported, as shown in Example 3-15.

Example 3-15 Cluster topology information

```
clnode_1:/# cltopinfo
Cluster Name: nfs_tiebr_cluster
Cluster Type: Linked
Heartbeat Type: Unicast
Repository Disks:
    Site 1 (site1@clnode_1): clnode_1r
    Site 2 (site2@clnode_2): clnode_2r
Cluster Nodes:
    Site 1 (site1):
        clnode_1
    Site 2 (site2):
        clnode_2

There are 2 node(s) and 1 network(s) defined
NODE clnode_1:
    Network net_ether_01
        clst_svIP      192.168.100.50
        clnode_1       192.168.100.51
NODE clnode_2:
    Network net_ether_01
        clst_svIP      192.168.100.50
        clnode_2       192.168.100.52

Resource Group rg_IHS
    Startup Policy  Online On Home Node Only
    Fallover Policy Fallover To Next Priority Node In The List
    Fallback Policy Never Fallback
    Participating Nodes   clnode_1 clnode_2
    Service IP Label     clst_svIP
clnode_1:/#
```

At the end of our environment preparation, the cluster was active. The RG, IBM Hypertext Transfer Protocol (HTTP) Server that is installed on the clnode_12 cluster disk with the datavg VG was online, as shown in Example 3-16.

Example 3-16 Cluster nodes and resource groups status

```
clnode_1:/# clmgr -cv -a name,state,raw_state query node
# NAME:STATE:RAW_STATE
clnode_1:NORMAL:ST_STABLE
clnode_2:NORMAL:ST_STABLE

clnode_1:/#
clnode_1:/# clRGinfo
-----
Group Name          Group State   Node
-----
rg_IHS              ONLINE      clnode_1@site1
                      ONLINE SECONDARY clnode_2@site2

clnode_1:/#
```

3.6.3 NFS server and client configuration

An important prerequisite of the NFS tie-breaker function deployment is that the function does not work with the more common NFS version 3.

Important: The NFS tie-breaker function requires NFS version 4.

Our test environment used an NFS server that is configured on an AIX 7.1 TL3 SP5 LPAR. This, of course, is not a requirement for deploying an NFS version 4 server.

A number of services must be active to allow NFSv4 communication between clients and servers:

- ▶ On the NFS server:
 - biod
 - nfsd
 - nfsrgyd
 - portmap
 - rpc.lockd
 - rpc.mountd
 - rpc.statd
 - TCP
- ▶ On the NFS client (all cluster nodes):
 - biod
 - nfsd
 - rpc.mountd
 - rpc.statd
 - TCP

Most of the previous services are usually active by default, and particular attention is required for the setup of the **nfsrgyd** service. As mentioned previously, this daemon must be running on *both the server and the clients*. In our case, the two cluster nodes. This daemon provides a name conversion service for NFS servers and clients that use NFS v4.

Starting the **nfsrgyd** daemon requires that the local NFS domain is set. The local NFS domain is stored in the /etc/nfs/local_domain file and it can be set by using the **chnfsdom** command, as shown in Example 3-17.

Example 3-17 Setting the local NFS domain

```
nfsserver:/# chnfsdom nfs_local_domain
nfsserver:/# startsrc -g nfs
[...]
nfsserver:/# lssrc -g nfs
Subsystem      Group          PID      Status
[...]
nfsrgyd        nfs           7077944    active
[...]
nfsserver:#
```

In addition, for the server, you must specify the root node directory, what clients mount as /, and the public node directory with the command-line interface (CLI), by using the **chnfs** command, as shown in Example 3-18.

Example 3-18 Setting the root and public node directory

```
nfsserver:# chnfs -r /nfs_root -p /nfs_root  
nfsserver:#
```

Alternatively, root, the public node directory, and the local NFS domain can be set with SMIT. Use the **smit nfs** command, follow the path **Network File System (NFS) → Configure NFS on This System**, then select the corresponding option:

- ▶ Change Version 4 Server Root Node
- ▶ Change Version 4 Server Public Node
- ▶ **Configure NFS Local Domain → Change NFS Local Domain**

As a final step for the NFS configuration, create the NFS resource, also known as the NFS export. Example 3-19 shows the NFS resource that was created by using SMIT by running the **smit mknfs** command.

Example 3-19 Creating an NFS v4 export

Add a Directory to Exports List

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

[Entry Fields]		
* Pathname of directory to export	/[/nfs_root/nfs_tie_breaker]	/
[...]		
Public filesystem?	no	+
[...]		
Allow access by NFS versions	[4]	+
[...]		
* Security method 1	[sys,krb5p,krb5i,krb5,dh]	+
* Mode to export directory	read-write	+
[...]		

F1=Help	F2=Refresh	F3=Cancel	F4=List
F5=Reset	F6=Command	F7>Edit	F8=Image
F9=Shell1	F10=Exit	Enter=Do	

Test the NFS configuration by manually mounting the NFS export to the clients, as shown in Example 3-20. The date column was removed from the output for clarity.

Example 3-20 Mounting an NFS v4 export

```
clnode_1:/# mount -o vers=4 nfsserver:/nfs_tie_breaker /mnt  
clnode_1:/# mount | egrep "node|---|tie"  
node      mounted      mounted over   vfs    options  
-----  -----  
nfsserver  /nfs_tie_breaker  /mnt        nfs4  vers=4,fg,soft,retry=1,timeo=10  
clnode_1:/#  
clnode_1:/# umount /mnt  
clnode_1:/#
```

3.6.4 NFS tie-breaker configuration

The NFS tie-breaker function can be configured either with CLI commands or SMIT.

To configure the NFS tie breaker by using SMIT, complete the following steps:

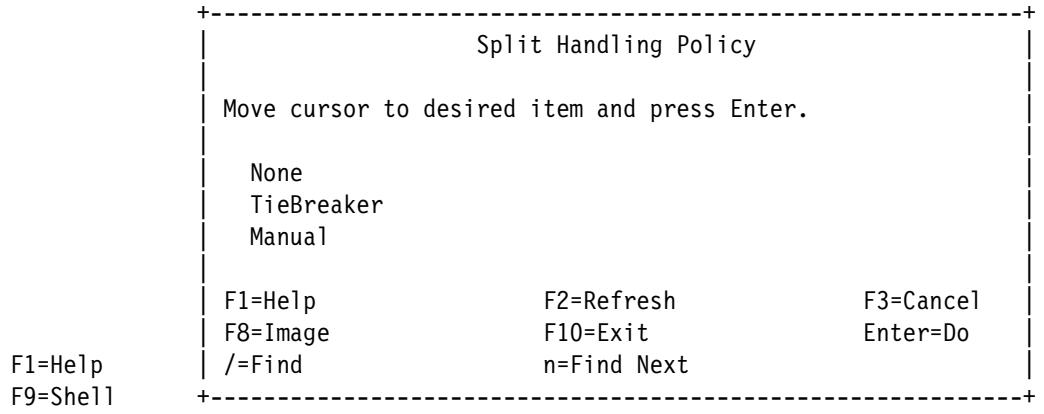
1. The SMIT menu that enables the configuration of NFS Tie Breaker split policy can be accessed by following the path **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy**.
2. Select Split Management Policy, as shown in Example 3-21.

Example 3-21 Configuring the split handling policy

Configure Cluster Split and Merge Policy

Move cursor to desired item and press Enter.

Split Management Policy
Merge Management Policy
Quarantine Policy



3. Select TieBreaker to open the menu where you choose the method to use for tie breaking, as shown in Example 3-22.

Example 3-22 Selecting the tie-breaker type

Configure Cluster Split and Merge Policy

Move cursor to desired item and press Enter.

Split Management Policy
Merge Management Policy
Quarantine Policy

Select TieBreaker Type		
Move cursor to desired item and press Enter.		
Disk	NFS	
F1=Help	F2=Refresh	F3=Cancel
F8=Image	F10=Exit	Enter=Do
/=Find	n=Find Next	
F1=Help		
F9=Shell		

4. After selecting NFS as the method for tie breaking, specify the NFS export server, directory, and the local mount point, as shown in Example 3-23.

Example 3-23 Configuring the NFS tie breaker for split handling policy by using SMIT

NFS TieBreaker Configuration

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

Split Handling Policy	[Entry Fields]
* NFS Export Server	NFS
* Local Mount Directory	[nfsserver_nfs]
* NFS Export Directory	[/nfs_tie_breaker]
	[/nfs_tie_breaker]

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

Split and merge policies must be of the same type, and the same rule applies for the tie-breaker type. Therefore, selecting the TieBreaker option for the Split Handling Policy field, and the NFS option for the TieBreaker type for that policy, implies also selecting those same options (TieBreaker and NFS) for the Merge Handling Policy:

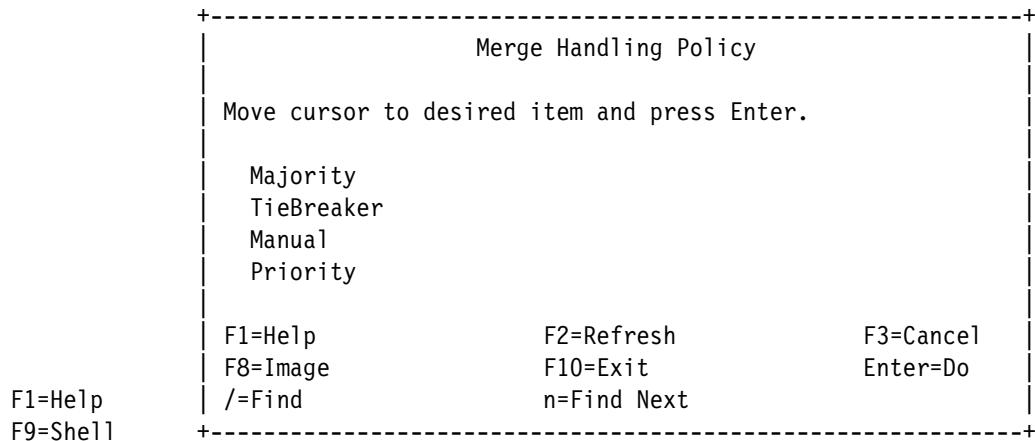
1. Configure the merge policy. From the same SMIT menu (**Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy**), select the Merge Management Policy option (Example 3-24).

Example 3-24 Configuring the merge handling policy

Configure Cluster Split and Merge Policy

Move cursor to desired item and press Enter.

Split Management Policy
Merge Management Policy
Quarantine Policy



2. Selecting the option of TieBreaker opens the menu that is shown in Example 3-25, where we again choose NFS as the method to use for tie breaking.

Example 3-25 Configuring NFS tie breaker for merge handling policy with SMIT

NFS TieBreaker Configuration

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

Merge Handling Policy	[Entry Fields]
* NFS Export Server	NFS
* Local Mount Directory	[nfsserver_nfs]
* NFS Export Directory	[/nfs_tie_breaker] [/nfs_tie_breaker]

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

Alternatively, both split and merge management policies can be configured by CLI by using the **clmgr modify cluster SPLIT_POLICY=tiebreaker MERGE_POLICY=tiebreaker** command followed by the **cl_sm** command, as shown in Example 3-26.

Example 3-26 Configuring the NFS tie breaker for the split and merge handling policy by using the CLI

```
clnode_1:/# /usr/es/sbin/cluster/utilities/cl_sm -s 'NFS' -k'nfsserver_nfs'
-g'/nfs_tie_breaker' -p'/nfs_tie_breaker'
The PowerHA SystemMirror split and merge policies have been updated.
Current policies are:
    Split Handling Policy :          NFS
    Merge Handling Policy :         NFS
NFS Export Server :
nfsserver_nfs
Local Mount Directory   :
/nfs_tie_breaker
NFS Export Directory   :
/nfs_tie_breaker
    Split and Merge Action Plan :      Restart
The configuration must be synchronized to make this change known across the
cluster.
clnode_1:/#
```

```
clnode_1:/# /usr/es/sbin/cluster/utilities/cl_sm -m 'NFS' -k'nfsserver_nfs'
-g'/nfs_tie_breaker' -p'/nfs_tie_breaker'
The PowerHA SystemMirror split and merge policies have been updated.
Current policies are:
    Split Handling Policy :          NFS
    Merge Handling Policy :         NFS
NFS Export Server :
nfsserver_nfs
Local Mount Directory   :
/nfs_tie_breaker
NFS Export Directory   :
/nfs_tie_breaker
    Split and Merge Action Plan :      Restart
The configuration must be synchronized to make this change known across the
cluster.
clnode_1:/#
```

At this point, a PowerHA cluster synchronization and restart and a CAA cluster restart are required. Complete the following steps:

1. Verify and synchronize the changes across the cluster either by using the SMIT menu (run the **smit sysmirror** command, then follow the path **Cluster Applications and Resources → Resource Groups → Verify and Synchronize Cluster Configuration**), or by the CLI by using the **clmgr sync cluster** command.
2. Stop cluster services for all nodes in the cluster by running the **clmgr stop cluster** command.
3. Stop the CAA daemon on all cluster nodes by running the **stopsrc -s clconfd** command.
4. Start the CAA daemon on all cluster nodes by running the **startsrc -s clconfd** command.

- Start cluster services for all nodes in the cluster by running the `clmgr start cluster` command.

Important: Verify all output messages that are generated by the synchronization and restart of the cluster because if an error occurred when activating the NFS tie-breaker policies, it might not necessarily produce an error on the overall result of a cluster synchronization action.

When all cluster nodes are synchronized and active, and the split and merge management policies are applied, the NFS resource is accessed by all nodes, as shown in Example 3-27 (the date column removed for clarity).

Example 3-27 Checking for the NFS export that is mounted on clients

node	mounted	mounted over	vfs	options
nfsserver_nfs	/nfs_tie_breaker	/nfs_tie_breaker	nfs4	vers=4,fg,soft,retry=1,timeo=10

node	mounted	mounted over	vfs	options
nfsserver_nfs	/nfs_tie_breaker	/nfs_tie_breaker	nfs4	vers=4,fg,soft,retry=1,timeo=10

3.6.5 NFS tie-breaker tests

A common method to simulate network connectivity loss is to use the `ifconfig` command to bring network interfaces down. Its effect is not persistent across restarts, so the NFS tie-breaker induced restart has the expected *recovery* effect. The test scenarios that we use and the actual results that we got are presented in the following sections.

Loss of network communication to the NFS server

Because using an NFS server resource is a secondary communication means, the primary one being the heartbeat network, the loss of communication between the cluster nodes and the NFS server did not have any visible results other than the expected log entries.

Loss of production/heartbeat network communication on standby node

The loss of the production heartbeat network communication on the standby node triggered no response because no RGs were online on that node at the time the simulated event occurred.

Loss of production heartbeat network communication on active node

The loss of the production heartbeat network communication on the active node triggered the expected failover action. This occurred because the network service IP and the underlying network (as resources essential to the RG that was online until the simulated event) were no longer available.

This action can be seen on both nodes' logs, as shown in the cluster.mmddyyyy logs in Example 3-28, for the disconnected node (the one that releases the RG).

Example 3-28 The cluster.mmddyyyy log for the node releasing the resource group

```
Nov 13 14:42:13 EVENT START: network_down clnode_1 net_ether_01
Nov 13 14:42:13 EVENT COMPLETED: network_down clnode_1 net_ether_01 0
Nov 13 14:42:13 EVENT START: network_down_complete clnode_1 net_ether_01
Nov 13 14:42:13 EVENT COMPLETED: network_down_complete clnode_1 net_ether_01 0
Nov 13 14:42:20 EVENT START: resource_state_change clnode_1
Nov 13 14:42:20 EVENT COMPLETED: resource_state_change clnode_1 0
Nov 13 14:42:20 EVENT START: rg_move_release clnode_1 1
Nov 13 14:42:20 EVENT START: rg_move clnode_1 1 RELEASE
Nov 13 14:42:20 EVENT START: stop_server app_IHS
Nov 13 14:42:20 EVENT COMPLETED: stop_server app_IHS 0
Nov 13 14:42:21 EVENT START: release_service_addr
Nov 13 14:42:22 EVENT COMPLETED: release_service_addr 0
Nov 13 14:42:25 EVENT COMPLETED: rg_move clnode_1 1 RELEASE 0
Nov 13 14:42:25 EVENT COMPLETED: rg_move_release clnode_1 1 0
Nov 13 14:42:27 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:27 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:30 EVENT START: network_up clnode_1 net_ether_01
Nov 13 14:42:30 EVENT COMPLETED: network_up clnode_1 net_ether_01 0
Nov 13 14:42:31 EVENT START: network_up_complete clnode_1 net_ether_01
Nov 13 14:42:31 EVENT COMPLETED: network_up_complete clnode_1 net_ether_01 0
Nov 13 14:42:33 EVENT START: rg_move_release clnode_1 1
Nov 13 14:42:33 EVENT START: rg_move clnode_1 1 RELEASE
Nov 13 14:42:33 EVENT COMPLETED: rg_move clnode_1 1 RELEASE 0
Nov 13 14:42:33 EVENT COMPLETED: rg_move_release clnode_1 1 0
Nov 13 14:42:35 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:36 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:38 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:39 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:39 EVENT START: rg_move_acquire clnode_1 1
Nov 13 14:42:39 EVENT START: rg_move clnode_1 1 ACQUIRE
Nov 13 14:42:39 EVENT COMPLETED: rg_move clnode_1 1 ACQUIRE 0
Nov 13 14:42:39 EVENT COMPLETED: rg_move_acquire clnode_1 1 0
Nov 13 14:42:41 EVENT START: rg_move_complete clnode_1 1
Nov 13 14:42:42 EVENT COMPLETED: rg_move_complete clnode_1 1 0
Nov 13 14:42:46 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:47 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:47 EVENT START: rg_move_acquire clnode_1 1
Nov 13 14:42:47 EVENT START: rg_move clnode_1 1 ACQUIRE
Nov 13 14:42:47 EVENT COMPLETED: rg_move clnode_1 1 ACQUIRE 0
Nov 13 14:42:47 EVENT COMPLETED: rg_move_acquire clnode_1 1 0
Nov 13 14:42:49 EVENT START: rg_move_complete clnode_1 1
Nov 13 14:42:53 EVENT COMPLETED: rg_move_complete clnode_1 1 0
Nov 13 14:42:55 EVENT START: resource_state_change_complete clnode_1
Nov 13 14:42:55 EVENT COMPLETED: resource_state_change_complete clnode_1 0
```

This action is also shown in Example 3-29 for the other node (the one that acquires the RG).

Example 3-29 The cluster.mmddyyyy log for the node acquiring the resource group

```
Nov 13 14:42:13 EVENT START: network_down clnode_1 net_ether_01
Nov 13 14:42:13 EVENT COMPLETED: network_down clnode_1 net_ether_01 0
Nov 13 14:42:14 EVENT START: network_down_complete clnode_1 net_ether_01
Nov 13 14:42:14 EVENT COMPLETED: network_down_complete clnode_1 net_ether_01 0
Nov 13 14:42:20 EVENT START: resource_state_change clnode_1
Nov 13 14:42:20 EVENT COMPLETED: resource_state_change clnode_1 0
Nov 13 14:42:20 EVENT START: rg_move_release clnode_1 1
Nov 13 14:42:20 EVENT START: rg_move clnode_1 1 RELEASE
Nov 13 14:42:20 EVENT COMPLETED: rg_move clnode_1 1 RELEASE 0
Nov 13 14:42:20 EVENT COMPLETED: rg_move_release clnode_1 1 0
Nov 13 14:42:27 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:29 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:31 EVENT START: network_up clnode_1 net_ether_01
Nov 13 14:42:31 EVENT COMPLETED: network_up clnode_1 net_ether_01 0
Nov 13 14:42:31 EVENT START: network_up_complete clnode_1 net_ether_01
Nov 13 14:42:31 EVENT COMPLETED: network_up_complete clnode_1 net_ether_01 0
Nov 13 14:42:33 EVENT START: rg_move_release clnode_1 1
Nov 13 14:42:33 EVENT START: rg_move clnode_1 1 RELEASE
Nov 13 14:42:34 EVENT COMPLETED: rg_move clnode_1 1 RELEASE 0
Nov 13 14:42:34 EVENT COMPLETED: rg_move_release clnode_1 1 0
Nov 13 14:42:36 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:36 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:39 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:39 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:39 EVENT START: rg_move_acquire clnode_1 1
Nov 13 14:42:39 EVENT START: rg_move clnode_1 1 ACQUIRE
Nov 13 14:42:39 EVENT COMPLETED: rg_move clnode_1 1 ACQUIRE 0
Nov 13 14:42:39 EVENT COMPLETED: rg_move_acquire clnode_1 1 0
Nov 13 14:42:42 EVENT START: rg_move_complete clnode_1 1
Nov 13 14:42:45 EVENT COMPLETED: rg_move_complete clnode_1 1 0
Nov 13 14:42:47 EVENT START: rg_move_fence clnode_1 1
Nov 13 14:42:47 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 14:42:47 EVENT START: rg_move_acquire clnode_1 1
Nov 13 14:42:47 EVENT START: rg_move clnode_1 1 ACQUIRE
Nov 13 14:42:49 EVENT START: acquire_takeover_addr
Nov 13 14:42:50 EVENT COMPLETED: acquire_takeover_addr 0
Nov 13 14:42:50 EVENT COMPLETED: rg_move clnode_1 1 ACQUIRE 0
Nov 13 14:42:50 EVENT COMPLETED: rg_move_acquire clnode_1 1 0
Nov 13 14:42:50 EVENT START: rg_move_complete clnode_1 1
Nov 13 14:42:50 EVENT START: start_server app_IHS
Nov 13 14:42:51 EVENT COMPLETED: start_server app_IHS 0
Nov 13 14:42:52 EVENT COMPLETED: rg_move_complete clnode_1 1 0
Nov 13 14:42:55 EVENT START: resource_state_change_complete clnode_1
Nov 13 14:42:55 EVENT COMPLETED: resource_state_change_complete clnode_1 0
```

Either log includes split_merge_prompt, site_down, or node_down events.

Loss of all network communication on standby node

The loss of all network communications from both the production heartbeat and connectivity to NFS server on the standby node triggers a restart of that node. This is in accordance with the split and merge action plan that was defined earlier.

As a starting point, both nodes were operational and the RG was online on node c1node_1 (Example 3-30).

Example 3-30 The cluster nodes and resource group status before the simulated network down event

```
c1node_1:/# clmgr -cva name,state,raw_state query node
# NAME:STATE:RAW_STATE
c1node_1:NORMAL:ST_STABLE
c1node_2:NORMAL:ST_STABLE
c1node_1:/#
```

```
c1node_1:/# clRGinfo
```

Group Name	Group State	Node
rg_IHS	ONLINE	c1node_1@site1
	ONLINE SECONDARY	c1node_2@site2

```
c1node_1:/#
```

Complete the following steps:

1. Temporarily bring down the network interfaces on the standby node c1node_2, in a terminal console opened by using the Hardware Management Console (HMC), as shown in Example 3-31.

Example 3-31 Simulating a network down event

```
c1node_2:/# ifconfig en0 down; ifconfig en1 down
c1node_2:/#
```

2. Within about a minute of the previous step, as a response to the split-brain situation, the node c1node_2 (with no communication to the NFS server) restarted itself. This can be seen on the virtual terminal console opened (by using the HMC) on that node, and is also reflected by the status of the cluster nodes (Example 3-32).

Example 3-32 Cluster nodes status immediately after a simulated network down event

```
c1node_1:/# clmgr -cva name,state,raw_state query node
# NAME:STATE:RAW_STATE
c1node_1:NORMAL:ST_STABLE
c1node_2:UNKNOWN:UNKNOWN
c1node_1:/#
```

3. After a restart, the node c1node_2 was functional, but with cluster services stopped (Example 3-33).

Example 3-33 Cluster nodes and resource group status after node restart

```
c1node_1:/# clmgr -cva name,state,raw_state query node
# NAME:STATE:RAW_STATE
c1node_1:NORMAL:ST_STABLE
c1node_2:OFFLINE:ST_INIT
c1node_1:/#
```

```
c1node_2:/# clRGinfo
```

Group Name	Group State	Node
rg_IHS	ONLINE	c1node_1@site1
	OFFLINE	c1node_2@site2

c1node_2:/#

- Manually start the services on the c1node_2 node (Example 3-34).

Example 3-34 Starting cluster services on the recently rebooted node

```
c1node_2:/# clmgr start node
[...]
c1node_2: Completed execution of /usr/es/sbin/cluster/etc/rc.cluster
c1node_2: with parameters: -boot -N -A -b -P cl_rc_cluster.
c1node_2: Exit status = 0
c1node_2:/#
```

- You are now back to the point before the simulated network loss event, with both nodes operational and the RG online on node c1node_1 (Example 3-35).

Example 3-35 Cluster nodes and resource group status after cluster services start

```
c1node_2:/# clmgr -cva name,state,raw_state query node
# NAME:STATE:RAW_STATE
c1node_1:NORMAL:ST_STABLE
c1node_2:NORMAL:ST_STABLE
c1node_2:/#
```

```
c1node_2:/# clRGinfo
```

Group Name	Group State	Node
rg_IHS	ONLINE	c1node_1@site1
	ONLINE SECONDARY	c1node_2@site2

c1node_2:/#

Loss of all network communication on the active node

The loss of all network communications of the production heartbeat and connectivity to NFS server on the active node, the node with the RG online, triggers the restart of that node. At the same time, the RG is independently brought online on the other node.

The test was performed exactly like the one on the standby node, as described in “Loss of all network communication on standby node” on page 60, and the process was similar. The only notable difference was that the previously active node, now disconnected, restarted. The other node, previously the standby node, was now bringing the RG online, thus ensuring service availability.

3.6.6 Log entries for monitoring and debugging

As expected, the usual system and cluster log files contain information that is related to the NFS tie-breaker events and actions. However, the particular content of these logs varies between the nodes as each node’s role differs.

Error report (errpt)

The surviving node includes log entries that are presented in chronological order with older entries first, as shown in Example 3-36.

Example 3-36 Error report events on the surviving node

LABEL: CONFIGRM_SITE_SPLIT

Description

ConfigRM received Site Split event notification

LABEL: CONFIGRM_PENDINGQUO

Description

The operational quorum state of the active peer domain has changed to PENDING_QUORUM. This state usually indicates that exactly half of the nodes that are defined in the peer domain are online. In this state cluster resources cannot be recovered although none will be stopped explicitly.

LABEL: LVM_GS_RLEAVE

Description

Remote node Concurrent Volume Group failure detected

LABEL: CONFIGRM_HASQUORUM_

Description

The operational quorum state of the active peer domain has changed to HAS_QUORUM. In this state, cluster resources may be recovered and controlled as needed by management applications.

The disconnected or restarted node includes log entries that are presented in chronological order with the older entries listed first, as shown in Example 3-37.

Example 3-37 Error report events on the restarted node

LABEL: CONFIGRM_SITE_SPLIT

Description

ConfigRM received Site Split event notification

LABEL: CONFIGRM_PENDINGQUO

Description

The operational quorum state of the active peer domain has changed to PENDING_QUORUM. This state usually indicates that exactly half of the nodes that are defined in the peer domain are online. In this state cluster resources cannot be recovered although none will be stopped explicitly.

LABEL: LVM_GS_RLEAVE

Description

Remote node Concurrent Volume Group failure detected

LABEL: CONFIGRM_NOQUORUM_E

Description

The operational quorum state of the active peer domain has changed to NO_QUORUM.

This indicates that recovery of cluster resources can no longer occur and that the node may be rebooted or halted in order to ensure that critical resources are released so that they can be recovered by another subdomain that may have operational quorum.

LABEL: CONFIGRM_REBOOTOS_E

Description

The operating system is being rebooted to ensure that critical resources are stopped so that another subdomain that has operational quorum may recover these resources without causing corruption or conflict.

LABEL: REBOOT_ID

Description

SYSTEM SHUTDOWN BY USER

LABEL: CONFIGRM_HASQUORUM_

Description

The operational quorum state of the active peer domain has changed to HAS_QUORUM. In this state, cluster resources may be recovered and controlled as needed by management applications.

LABEL: CONFIGRM_ONLINE_ST

Description

The node is online in the domain indicated in the detail data.

The restarted node's log includes information that is relative to the surviving node's log, and information about the restart event.

The cluster.mmddyyyy log file

For each split-brain situation encountered, the content of the cluster.mmddyyyy log file was similar on the two nodes. The surviving node's log entries are presented in Example 3-38.

Example 3-38 The cluster.mmddyyyy log entries on the surviving node

```
Nov 13 13:40:03 EVENT START: split_merge_prompt split
Nov 13 13:40:07 EVENT COMPLETED: split_merge_prompt split 0
Nov 13 13:40:07 EVENT START: site_down site2
Nov 13 13:40:09 EVENT START: site_down_remote site2
Nov 13 13:40:09 EVENT COMPLETED: site_down_remote site2 0
Nov 13 13:40:09 EVENT COMPLETED: site_down site2 0
Nov 13 13:40:09 EVENT START: node_down clnode_2
Nov 13 13:40:09 EVENT COMPLETED: node_down clnode_2 0
Nov 13 13:40:11 EVENT START: rg_move_release clnode_1 1
Nov 13 13:40:11 EVENT START: rg_move clnode_1 1 RELEASE
Nov 13 13:40:11 EVENT COMPLETED: rg_move clnode_1 1 RELEASE 0
Nov 13 13:40:11 EVENT COMPLETED: rg_move_release clnode_1 1 0
Nov 13 13:40:11 EVENT START: rg_move_fence clnode_1 1
Nov 13 13:40:12 EVENT COMPLETED: rg_move_fence clnode_1 1 0
Nov 13 13:40:14 EVENT START: node_down_complete clnode_2
Nov 13 13:40:14 EVENT COMPLETED: node_down_complete clnode_2 0
```

The log entries for the same event, but this time on the disconnected or restarted node, are shown in Example 3-39.

Example 3-39 The cluster.mmddyyyy log entries on the restarted node

```
Nov 13 13:40:03 EVENT START: split_merge_prompt split
Nov 13 13:40:03 EVENT COMPLETED: split_merge_prompt split 0
Nov 13 13:40:12 EVENT START: site_down site1
Nov 13 13:40:13 EVENT START: site_down_remote site1
Nov 13 13:40:13 EVENT COMPLETED: site_down_remote site1 0
Nov 13 13:40:13 EVENT COMPLETED: site_down site1 0
Nov 13 13:40:13 EVENT START: node_down clnode_1
Nov 13 13:40:13 EVENT COMPLETED: node_down clnode_1 0
Nov 13 13:40:15 EVENT START: network_down clnode_2 net_ether_01
Nov 13 13:40:15 EVENT COMPLETED: network_down clnode_2 net_ether_01 0
Nov 13 13:40:15 EVENT START: network_down_complete clnode_2 net_ether_01
Nov 13 13:40:15 EVENT COMPLETED: network_down_complete clnode_2 net_ether_01 0
Nov 13 13:40:18 EVENT START: rg_move_release clnode_2 1
Nov 13 13:40:18 EVENT START: rg_move clnode_2 1 RELEASE
Nov 13 13:40:18 EVENT COMPLETED: rg_move clnode_2 1 RELEASE 0
Nov 13 13:40:18 EVENT COMPLETED: rg_move_release clnode_2 1 0
Nov 13 13:40:18 EVENT START: rg_move_fence clnode_2 1
Nov 13 13:40:19 EVENT COMPLETED: rg_move_fence clnode_2 1 0
Nov 13 13:40:21 EVENT START: node_down_complete clnode_1
Nov 13 13:40:21 EVENT COMPLETED: node_down_complete clnode_1 0
```

This log also includes the information about the network_down event.

The cluster.log file

The cluster.log file includes much of the information in the cluster.mmddyyyy log file. The notable exception is that this one cluster.log also included information about the quorum status losing and regaining quorum. For the disconnected or restarted node only, the cluster.log file has information about the restart event, as shown in Example 3-40.

Example 3-40 The cluster.log entries on the restarted node

```
Nov 13 13:40:03 clnode_2 [...] EVENT START: split_merge_prompt split
Nov 13 13:40:03 clnode_2 [...] CONFIGRM_SITE_SPLIT_ST ConfigRM received Site Split event
notification
Nov 13 13:40:03 clnode_2 [...] EVENT COMPLETED: split_merge_prompt split 0
Nov 13 13:40:09 clnode_2 [...] CONFIGRM_PENDINGQUORUM_ER The operational quorum state of
the active peer domain has changed to PENDING_QUORUM. This state usually indicates that
exactly half of the nodes that are defined in the peer domain are online. In this state
cluster resources cannot be recovered although none will be stopped explicitly.
Nov 13 13:40:12 clnode_2 [...] EVENT START: site_down site1
Nov 13 13:40:13 clnode_2 [...] EVENT START: site_down_remote site1
Nov 13 13:40:13 clnode_2 [...] EVENT COMPLETED: site_down_remote site1 0
Nov 13 13:40:13 clnode_2 [...] EVENT COMPLETED: site_down site1 0
Nov 13 13:40:13 clnode_2 [...] EVENT START: node_down clnode_1
Nov 13 13:40:13 clnode_2 [...] EVENT COMPLETED: node_down clnode_1 0
Nov 13 13:40:15 clnode_2 [...] EVENT START: network_down clnode_2 net_ether_01
Nov 13 13:40:15 clnode_2 [...] EVENT COMPLETED: network_down clnode_2 net_ether_01 0
Nov 13 13:40:15 clnode_2 [...] EVENT START: network_down_complete clnode_2 net_ether_01
Nov 13 13:40:16 clnode_2 [...] EVENT COMPLETED: network_down_complete clnode_2 net_ether_01
0
Nov 13 13:40:18 clnode_2 [...] EVENT START: rg_move_release clnode_2 1
Nov 13 13:40:18 clnode_2 [...] EVENT START: rg_move clnode_2 1 RELEASE
Nov 13 13:40:18 clnode_2 [...] EVENT COMPLETED: rg_move clnode_2 1 RELEASE 0
```

```
Nov 13 13:40:18 clnode_2 [...] EVENT COMPLETED: rg_move_release clnode_2 1 0
Nov 13 13:40:18 clnode_2 [...] EVENT START: rg_move_fence clnode_2 1
Nov 13 13:40:19 clnode_2 [...] EVENT COMPLETED: rg_move_fence clnode_2 1 0
Nov 13 13:40:21 clnode_2 [...] EVENT START: node_down_complete clnode_1
Nov 13 13:40:21 clnode_2 [...] EVENT COMPLETED: node_down_complete clnode_1 0
Nov 13 13:40:29 clnode_2 [...] CONFIGRM_NOQUORUM_ER The operational quorum state of the
active peer domain has changed to NO_QUORUM. This indicates that recovery of cluster
resources can no longer occur and that the node may be rebooted or halted in order to
ensure that critical resources are released so that they can be recovered by another
subdomain that may have operational quorum.
Nov 13 13:40:29 clnode_2 [...] CONFIGRM_REBOOTOS_ER The operating system is being rebooted
to ensure that critical resources are stopped so that another subdomain that has
operational quorum may recover these resources without causing corruption or conflict.
[...]
Nov 13 13:41:32 clnode_2 [...] RMCD_INFO_0_ST The daemon is started.
Nov 13 13:41:33 clnode_2 [...] CONFIGRM_STARTED_ST IBM.ConfigRM daemon has started.
Nov 13 13:42:03 clnode_2 [...] GS_START_ST Group Services daemon started DIAGNOSTIC
EXPLANATION HAGS daemon started by SRC. Log file is
/var/ct/1Z4w8kYNeHvP2dxgyEaCe2/log/cttags/trace.
Nov 13 13:42:36 clnode_2 [...] CONFIGRM_HASQUORUM_ST The operational quorum state of the
active peer domain has changed to HAS_QUORUM. In this state, cluster resources may be
recovered and controlled as needed by management applications.
Nov 13 13:42:36 clnode_2 [...] CONFIGRM_ONLINE_ST The node is online in the domain
indicated in the detail data. Peer Domain Name nfs_tiebr_cluster
Nov 13 13:42:38 clnode_2 [...] STORAGERM_STARTED_ST IBM.StorageRM daemon has started.
```



What is new with IBM Cluster Aware AIX and Reliable Scalable Clustering Technology

This chapter provides details about what is new with Cluster Aware AIX (CAA) and with Reliable Scalable Clustering Technology (RSCT).

This chapter covers the following topics:

- ▶ Cluster Aware AIX
- ▶ Automatic repository update for the repository disk
- ▶ Reliable Scalable Cluster Technology overview
- ▶ PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX

4.1 Cluster Aware AIX

This section describes some of the new CAA features.

4.1.1 Cluster Aware AIX tunables

This section mentions how CAA tunables behave. Usually you do not have to changes these CAA tunables. Changes to CAA tunables can be made by using **clmgr** or **smit**. Example 4-1 shows the list of the CAA tunables with IBM AIX V7.2.0.0 and IBM PowerHA V7.2.0. Newer versions can have more tunables, different defaults, or both.

Attention: Do not change any of these tunables without the explicit permission of IBM technical support.

In general, you must never modify these values because these values are modified and managed by PowerHA.

Example 4-1 List of Cluster Aware AIX tunables

```
# clctrl -tune -a
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).communication_mode = u
    ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).config_timeout = 240
        ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).deadman_mode = a
            ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).link_timeout = 30000
                ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).local_merge_policy = m
                    ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).network_fdt = 20000
ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).no_if_traffic_monitor = 0
    ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).node_down_delay = 10000
        ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).node_timeout = 30000
            ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).packet_ttl = 32
                ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).remote_hb_factor = 1
                    ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).repos_mode = e
                        ha72cluster(71a0d83c-e467-11e5-8022-4217e0ce7b02).site_merge_policy = p
#
#
```

4.1.2 What is new in Cluster Aware AIX: Overview

The following new features are included in CAA:

- ▶ Automatic Repository Update (ARU)
Also known as Automatic Repository Replacement (ARR). For more information, see 4.2, “Automatic repository update for the repository disk” on page 82.
- ▶ Monitor /var usage
For more information, see 4.1.3, “Monitoring /var usage” on page 69.
- ▶ New **-g** option for the **lsccluster** command
For more information, see 4.1.4, “New lsccluster option -g” on page 71.
- ▶ Interface Failure Detection:
 - Tuning for Interface Failure Detection
 - Send multicast packets to generate incoming traffic
 - Implementation of network monitor (NETMON) within CAA

For more information, see 4.1.5, “Network Failure Detection Time (FDT)” on page 80.

- ▶ Functional enhancements
 - Reduce dependency of CAA node name on host name
 - Roll back on **mkcluster** failure or partial success
- ▶ Reliability, availability, and serviceability (RAS) enhancements
 - Message improvements
 - Several `syslog.caa` serviceability improvements
 - Enhanced Dead Man Switch (DMS) error logging

4.1.3 Monitoring /var usage

Starting with PowerHA V7.2, the `/var` file system is monitored by default. This monitoring is done by the `clconfd` subsystem. The following default values are used:

Threshold	75% (range 70 - 95)
Interval	15 min (range 5 - 30)

To change the default values, use the **chssys** command. The **-t** option is used to specify the threshold in % and the **-i** option is used to specify the interval:

```
chssys -s clconfd -a "-t 80 -i 10"
```

To check what values are currently used, you have two options: You can use the **ps -ef | grep clconfd** or the **odmget -q "subsysname='clconfd'" SRCsubsys** command. Example 4-2 shows the output of the two commands with default values. When using the **odmget** command, the **cmdargs** line has no arguments that are listed. The same happens if **ps -ef** is used because there are no arguments that are displayed after **clconfd**.

Example 4-2 Check clconfd (when default values are used)

```
# ps -ef | grep clconfd
    root 3713096  3604778  0 17:50:30      -  0:00 /usr/sbin/clconfd
#
# odmget -q "subsysname='clconfd'" SRCsubsys
SRCsubsys:

    subsysname = "clconfd"
    synonym = ""
    cmdargs =
    path = "/usr/sbin/clconfd"
    uid = 0
    auditid = 0
    standin = "/dev/null"
    standout = "/dev/null"
    standerr = "/dev/null"
    action = 1
    multi = 0
    contact = 2
    svrkey = 0
    svrmtype = 0
    priority = 20
    signorm = 2
    sigforce = 9
    display = 1
    waittime = 20
```

```
grpname = "caa"
```

Example 4-3 shows what happens when you change the default values, and what the output of **odmget** and **ps -ef** looks like after that change.

Important: You need to stop and start the subsystem to activate your changes.

Example 4-3 Change monitoring for /var

```
# chssys -s clconfd -a "-t 80 -i 10"
0513-077 Subsystem has been changed
#
# stopsrc -s clconfd
0513-044 The clconfd Subsystem was requested to stop.
#
# startsrc -s clconfd
0513-059 The clconfd Subsystem has been started. Subsystem PID is 13173096.
# ps -ef | grep clconfd
    root 13173096  3604778  0 17:50:30      -  0:00 /usr/sbin/clconfd -t 80 -i 10
#
# odmget -q "subsysname='clconfd'" SRCsubsys
```

SRCsubsys:

```
    subsysname = "clconfd"
    synonym = ""
    cmdargs = "-t 80 -i 10"
    path = "/usr/sbin/clconfd"
    uid = 0
    auditid = 0
    standin = "/dev/null"
    standout = "/dev/null"
    standerr = "/dev/null"
    action = 1
    multi = 0
    contact = 2
    svrkey = 0
    svrmtype = 0
    priority = 20
    signorm = 2
    sigforce = 9
    display = 1
    waittime = 20
    grpname = "caa"
```

If the threshold is exceeded, then you get an entry in the error log. Example 4-4 shows what such an error entry can look like.

Example 4-4 Error message of /var monitoring

```
LABEL:          CL_VAR_FULL
IDENTIFIER:     E5899EEB
```

```
Date/Time:      Fri Nov 13 17:47:15 2015
Sequence Number: 1551
Machine Id:     00F747C94C00
```

```

Node Id: esp-c2n1
Class: S
Type: PERM
WPAR: Global
Resource Name: CAA (for RSCT)

Description
/var filesystem is running low on space

Probable Causes
Unknown

Failure Causes
Unknown

Recommended Actions
RSCT could malfunction if /var gets full
Increase the filesystem size or delete unwanted files

Detail Data
Percent full
81
Percent threshold
80

```

4.1.4 New `lscuster -g`

Starting with AIX 7.1 TL4 and AIX 7.2, there is an additional option for the CAA `lscuster` command. The new option `-g` lists the used communication paths of CAA.

Note: At the time of writing, this option was not available in AIX versions earlier than AIX 7.1.4.

The `lscuster -i` command lists all of the seen communication paths by CAA but it does not show if all of them can potentially be used for heartbeating. This is particularly the case if you use a network that is set to private, or if you removed a network from the PowerHA configuration.

Using all interfaces

When using the standard way to configure a cluster, all configured networks in AIX are added to the PowerHA and CAA configuration. In our test cluster, we configured two IP interfaces in AIX. Example 4-5 shows the two networks in our PowerHA configuration, all set to public.

Example 4-5 The `clisif` command with all interfaces on public

```

> clisif
Adapter          Type      Network   Net Type  Attribute  Node       IP Address
Hardware Address Interface Name  Global Name     Netmask        Alias for HB Prefix Length

n1adm            boot      adm_net   ether    public    powerha-c2n1 10.17.1.100
en1              boot      255.255.255.0          24
powerha-c2n1    boot      service_net ether    public    powerha-c2n1 172.16.150.121
en0              boot      255.255.0.0           16

```

```

c2svc      service  service_net ether  public   powerha-c2n1 172.16.150.125
255.255.0.0          16
n2adm      boot    adm_net  ether    public   powerha-c2n2 10.17.1.110
en1          255.255.255.0          24
powerha-c2n2     boot    service_net ether  public   powerha-c2n2 172.16.150.122
en0          255.255.0.0          16
c2svc      service  service_net ether  public   powerha-c2n2 172.16.150.125
255.255.0.0          16
#

```

In this case, the **lscluster -i** output looks like what is shown in Example 4-6.

Example 4-6 The lscluster -i command (all interfaces on public)

```

> lscluster -i
Network/Storage Interface Query

Cluster Name: ha72cluster
Cluster UUID: 63d12f4e-e61b-11e5-8016-4217e0ce7b02
Number of nodes reporting = 2
Number of nodes stale = 0
Number of nodes expected = 2

Node powerha-c2n1.munich.de.ibm.com
Node UUID = 63b68a36-e61b-11e5-8016-4217e0ce7b02
Number of interfaces discovered = 3
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E0:CE:7B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.121 broadcast 172.16.255.255 netmask
255.255.0.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, en1
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E0:CE:7B:05
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 10.17.1.100 broadcast 10.17.1.255 netmask
255.255.255.0

```

```

Number of cluster multicast addresses configured on interface = 1
IPv4 MULTICAST ADDRESS: 228.16.150.121
Interface number 3, dpcm
  IFNET type = 0 (none)
  NDD type = 305 (NDD_PINGCOMM)
  Smoothed RTT across interface = 750
  Mean deviation in network RTT across interface = 1500
  Probe interval for interface = 22500 ms
  IFNET flags for interface = 0x00000000
  NDD flags for interface = 0x00000009
  Interface state = UP RESTRICTED AIX_CONTROLLED

Node powerha-c2n2.munich.de.ibm.com
Node UUID = 63b68a86-e61b-11e5-8016-4217e0ce7b02
Number of interfaces discovered = 3
  Interface number 1, en0
    IFNET type = 6 (IFT_ETHER)
    NDD type = 7 (NDD_IS088023)
    MAC address length = 6
    MAC address = 42:17:E4:E6:1B:02
    Smoothed RTT across interface = 0
    Mean deviation in network RTT across interface = 0
    Probe interval for interface = 990 ms
    IFNET flags for interface = 0x1E084863
    NDD flags for interface = 0x0021081B
    Interface state = UP
    Number of regular addresses configured on interface = 1
    IPv4 ADDRESS: 172.16.150.122 broadcast 172.16.255.255 netmask
255.255.0.0
  Number of cluster multicast addresses configured on interface = 1
  IPv4 MULTICAST ADDRESS: 228.16.150.121
  Interface number 2, en1
    IFNET type = 6 (IFT_ETHER)
    NDD type = 7 (NDD_IS088023)
    MAC address length = 6
    MAC address = 42:17:E4:E6:1B:05
    Smoothed RTT across interface = 0
    Mean deviation in network RTT across interface = 0
    Probe interval for interface = 990 ms
    IFNET flags for interface = 0x1E084863
    NDD flags for interface = 0x0021081B
    Interface state = UP
    Number of regular addresses configured on interface = 1
    IPv4 ADDRESS: 10.17.1.110 broadcast 10.17.1.255 netmask
255.255.255.0
  Number of cluster multicast addresses configured on interface = 1
  IPv4 MULTICAST ADDRESS: 228.16.150.121
  Interface number 3, dpcm
    IFNET type = 0 (none)
    NDD type = 305 (NDD_PINGCOMM)
    Smoothed RTT across interface = 750
    Mean deviation in network RTT across interface = 1500
    Probe interval for interface = 22500 ms
    IFNET flags for interface = 0x00000000
    NDD flags for interface = 0x00000009

```

```
Interface state = UP RESTRICTED AIX_CONTROLLED
root@powerha-c2n1:/>
```

Example 4-7 shows the output of the **lscluster -g** command. When you compare the output of the **lscluster -g** command with the **lscluster -i** command, you should not find any differences. There are no differences because all of the networks are allowed to potentially be used for heartbeat in this example.

Example 4-7 The lscluster -g command output in relation to the clsif output

```
# > lscluster -g
Network/Storage Interface Query

Cluster Name: ha72cluster
Cluster UUID: 63d12f4e-e61b-11e5-8016-4217e0ce7b02
Number of nodes reporting = 2
Number of nodes stale = 0
Number of nodes expected = 2

Node powerha-c2n1.munich.de.ibm.com
Node UUID = 63b68a36-e61b-11e5-8016-4217e0ce7b02
Number of interfaces discovered = 3
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E0:CE:7B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.121 broadcast 172.16.255.255 netmask 255.255.0.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, en1
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E0:CE:7B:05
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 10.17.1.100 broadcast 10.17.1.255 netmask 255.255.255.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 3, dpcom
        IFNET type = 0 (none)
        NDD type = 305 (NDD_PINGCOMM)
        Smoothed RTT across interface = 750
        Mean deviation in network RTT across interface = 1500
        Probe interval for interface = 22500 ms
        IFNET flags for interface = 0x00000000
        NDD flags for interface = 0x00000009
```

```

        Interface state = UP RESTRICTED AIX_CONTROLLED

Node powerha-c2n2.munich.de.ibm.com
Node UUID = 63b68a86-e61b-11e5-8016-4217e0ce7b02
Number of interfaces discovered = 3
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E4:E6:1B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.122 broadcast 172.16.255.255 netmask 255.255.0.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, en1
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E4:E6:1B:05
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 10.17.1.110 broadcast 10.17.1.255 netmask 255.255.255.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 3, dpcom
        IFNET type = 0 (none)
        NDD type = 305 (NDD_PINGCOMM)
        Smoothed RTT across interface = 750
        Mean deviation in network RTT across interface = 1500
        Probe interval for interface = 22500 ms
        IFNET flags for interface = 0x00000000
        NDD flags for interface = 0x00000009
        Interface state = UP RESTRICTED AIX_CONTROLLED
root@powerha-c2n1:/>

```

One network set to private

The following examples in this section describe the **lscluster** command output when you decide to change one or more networks to private. Example 4-8 shows the starting point for this example. In our testing environment, we changed one network to private.

Note: Private networks cannot be used for any services. When you want to use a service IP address, the network must be public.

Example 4-8 The clslif command (private)

```
# clslif
Adapter          Type      Network   Net Type  Attribute  Node     IP
Address         Hardware Address Interface Name Global Name    Netmask
Alias for HB Prefix Length

n1adm           service   adm_net   ether    private    powerha-c2n1
10.17.1.100
24
powerha-c2n1   boot      service_net ether    public     powerha-c2n1
172.16.150.121
16
c2svc          service   service_net ether    public     powerha-c2n1
172.16.150.125
16
n2adm          service   adm_net   ether    private    powerha-c2n2
10.17.1.110
24
powerha-c2n2   boot      service_net ether    public     powerha-c2n2
172.16.150.122
16
c2svc          service   service_net ether    public     powerha-c2n2
172.16.150.125
16
#
```

Because we did not change the architecture of our cluster, the output of the **lscluster -i** command is still the same, as shown in Example 4-6 on page 72.

Remember: You must synchronize your cluster before the change to private is visible in CAA.

Example 4-9 shows the **lscluster -g** command output after the synchronization. If you now compare the output of the **lscluster -g** command with the **lscluster -i** command or with the **lscluster -g** output from the previous example, you see that the entries about en1 (in our example) do not appear any longer. The list of networks potentially allowed to be used for heartbeat is shorter.

Example 4-9 The lscluster -g command (one private network)

```
# lscluster -g
Network/Storage Interface Query

Cluster Name: ha72cluster
Cluster UUID: 55430510-e6a7-11e5-8035-4217e0ce7b02
```

```

Number of nodes reporting = 2
Number of nodes stale = 0
Number of nodes expected = 2

Node powerha-c2n1.munich.de.ibm.com
Node UUID = 55284db0-e6a7-11e5-8035-4217e0ce7b02
Number of interfaces discovered = 2
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E0:CE:7B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.121 broadcast 172.16.255.255 netmask
255.255.0.0
    Number of cluster multicast addresses configured on interface = 1
    IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, dpcm
        IFNET type = 0 (none)
        NDD type = 305 (NDD_PINGCOMM)
        Smoothed RTT across interface = 750
        Mean deviation in network RTT across interface = 1500
        Probe interval for interface = 22500 ms
        IFNET flags for interface = 0x00000000
        NDD flags for interface = 0x00000009
        Interface state = UP RESTRICTED AIX_CONTROLLED

Node powerha-c2n2.munich.de.ibm.com
Node UUID = 55284df6-e6a7-11e5-8035-4217e0ce7b02
Number of interfaces discovered = 2
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E4:E6:1B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.122 broadcast 172.16.255.255 netmask
255.255.0.0
    Number of cluster multicast addresses configured on interface = 1
    IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, dpcm
        IFNET type = 0 (none)
        NDD type = 305 (NDD_PINGCOMM)

```

```

Smoothed RTT across interface = 750
Mean deviation in network RTT across interface = 1500
Probe interval for interface = 22500 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = UP RESTRICTED AIX_CONTROLLED
#

```

Removing networks from PowerHA

The examples in this section describe the **lscuster** command output when you remove one or more networks from the list of known networks in PowerHA. Example 4-10 shows the starting point for this example. In our test environment, we removed the `adm_net` network.

Example 4-10 The clisif command (removed network)

```

# clisif
Adapter          Type      Network   Net Type  Attribute  Node     IP
Address         Hardware Address Interface Name Global Name    Netmask
Alias for HB Prefix Length

powerha-c2n1      boot      service_net ether    public    powerha-c2n1
172.16.150.121
16
c2svc           service    service_net ether    public    powerha-c2n1
172.16.150.125
16
powerha-c2n2      boot      service_net ether    public    powerha-c2n2
172.16.150.122
16
c2svc           service    service_net ether    public    powerha-c2n2
172.16.150.125
16
#

```

Because we did not change the architecture of our cluster, the output of the **lscuster -i** command is still the same as listed in Example 4-6 on page 72.

You must synchronize your cluster before the change to private is visible in CAA.

Example 4-11 shows the **lscuster -g** output after the synchronization. If you now compare the output of the **lscuster -g** command with the previous **lscuster -i** command, or with the **lscuster -g** output in “Using all interfaces” on page 71, you see that the entries about `en1` (in our example) do not appear.

When you compare the content of Example 4-11 with the content of Example 4-9 on page 76 in “One network set to private” on page 76, you see that the output of the **lscuster -g** commands is identical.

Example 4-11 The lscuster -g command output (removed network)

```

# lscuster -g
Network/Storage Interface Query

Cluster Name: ha72cluster
Cluster UUID: 63d12f4e-e61b-11e5-8016-4217e0ce7b02
Number of nodes reporting = 2

```

```

Number of nodes stale = 0
Number of nodes expected = 2

Node powerha-c2n1.munich.de.ibm.com
Node UUID = 63b68a36-e61b-11e5-8016-4217e0ce7b02
Number of interfaces discovered = 2
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E0:CE:7B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.121 broadcast 172.16.255.255 netmask
255.255.0.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, dpcm
        IFNET type = 0 (none)
        NDD type = 305 (NDD_PINGCOMM)
        Smoothed RTT across interface = 750
        Mean deviation in network RTT across interface = 1500
        Probe interval for interface = 22500 ms
        IFNET flags for interface = 0x00000000
        NDD flags for interface = 0x00000009
        Interface state = UP RESTRICTED AIX_CONTROLLED

Node powerha-c2n2.munich.de.ibm.com
Node UUID = 63b68a86-e61b-11e5-8016-4217e0ce7b02
Number of interfaces discovered = 2
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = 42:17:E4:E6:1B:02
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.150.122 broadcast 172.16.255.255 netmask
255.255.0.0
        Number of cluster multicast addresses configured on interface = 1
        IPv4 MULTICAST ADDRESS: 228.16.150.121
    Interface number 2, dpcm
        IFNET type = 0 (none)
        NDD type = 305 (NDD_PINGCOMM)
        Smoothed RTT across interface = 750

```

```
Mean deviation in network RTT across interface = 1500
Probe interval for interface = 22500 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = UP RESTRICTED AIX_CONTROLED
root@powerha-c2n1:/>
#
```

4.1.5 Network Failure Detection Time (FDT)

PowerHA V7.1 had a fixed latency for network failure detection that was about 5 seconds. In PowerHA V7.2, the default is now 20 seconds. The tunable is named `network_fdt` in CAA. To change it use `c1mgr` or `smit`.

Note: The `network_fdt` tunable is also available in PowerHA V7.1.3. To get it for your PowerHA V7.1.3 version, you must open a PMR and request the *Tunable FDT interim fix bundle*.

The self-adjusting network heartbeating behavior (CAA) that was introduced with PowerHA V7.1.0 is still there and is used. It has no impact in the network failure detection time.

The `network_fdt` tunable can be set to zero to maintain the default behavior. In newer versions it may be enforced to be at least 5 seconds. The minimum delta between the `network_fdt` tunable and the `node_timeout` tunable must be 10 second.

The default recognition time for a network problem is not affected by this tunable. It is 0 for hard failures and 5 seconds for soft failures (since PowerHA V7.1.0). CAA continues to check the network, but it waits until the end of the defined timeout to create a network down event.

For PowerHA nodes, when the effective level of CAA is 4, also known as the 2015 release, CAA automatically sets the `network_fdt` to 20 seconds and the `node_timeout` to 30 seconds.

To check for the effective CAA level, use the `lsc1uster -c` command. The last two lines of the `lsc1uster -c` output list the local CAA level and the effective CAA level. In normal situations, these two show the same level. In case of an operating system update, it can temporarily show different levels. Example 4-12 shows the numbers that you get when you are on CAA level 4.

Example 4-12 The lsc1uster -c command to check for CAA level

```
# lsc1uster -c
Cluster Name: ha72cluster
Cluster UUID: 55430510-e6a7-11e5-8035-4217e0ce7b02
Number of nodes in cluster = 2
    Cluster ID for node powerha-c2n1: 1
    Primary IP address for node powerha-c2n1: 172.16.150.121
    Cluster ID for node powerha-c2n2: 2
    Primary IP address for node powerha-c2n2: 172.16.150.122
Number of disks in cluster = 1
    Disk = hdisk2 UUID = 7c83d44b-9ac1-4ed7-ce3f-4e950f7ac9c6 cluster_major =
0 cluster_minor = 1
Multicast for site LOCAL: IPv4 228.16.150.121 IPv6 ff05::e410:9679
Communication Mode: unicast
Local node maximum capabilities: CAA_NETMON, AUTO_REPO_REPLACE, HNAME_CHG,
UNICAST, IPV6, SITE
```

```
Effective cluster-wide capabilities: CAA_NETMON, AUTO_REPOS_REPLACE, HNAME_CHG,  
UNICAST, IPV6, SITE  
Local node max level: 40000  
Effective cluster level: 40000
```

Note: Depending on the installed AIX Service Pack and fixes, the CAA level might not be displayed.

In this case, the only way to know whether the CAA level is 4 is to check whether AUTO_REPOS_REPLACE is listed for the effective cluster-wide capabilities in the output of **lscluster -c** command.

For example, use the following command:

```
# lscluster -c | grep "Effective cluster-wide capabilities"  
Effective cluster-wide capabilities: CAA_NETMON, AUTO_REPOS_REPLACE, HNAME_CHG,  
UNICAST, IPV6, SITE  
#
```

Example 4-13 shows how to both check and change the CAA network tunable attribute for PowerHA V7.1.3 by using the CAA native **clctrl** command. Keep in mind this is for PowerHA V7.1.3 FDT iFix only. In this case the values are listed in milliseconds.

Example 4-13 Using clctrl to change the CAA network tunable

```
# clctrl -tune -o network_fdt  
HA72a_cluster(641d80c2-bd87-11e5-8005-96d75a7c7f02).network_fdt = 20000  
  
# clctrl -tune -o network_fdt=10000  
1 tunable updated on cluster PHA72a_cluster  
  
# clctrl -tune -o network_fdt  
PHA72a_cluster(641d80c2-bd87-11e5-8005-96d75a7c7f02).network_fdt = 10000
```

Attention: Do not use the **clctrl** command to change the network_fdt tunable.
Remember to use **clmgr** or **smit**.

Example 4-13 is only for old versions where **clmgr** or **smit** could not be used.

For PowerHA V7.2 and newer version of the FDT iFix bundle, the correct way to change is by using **clmgr** or **smit**. Example 4-14 shows the SMIT screen where the network_fdt and the node_timeout tunable can be changed. The value needs to be specified in seconds. Network Failure Detection Time is the wording in smit for FDT. The path to it in smit is smit sysmirror → Custom Cluster Configuration → Cluster Nodes and Networks → Manage the Cluster → Cluster heartbeat settings. Or use the fast path smit cm_chng_tunables.

Important: Only use **clmgr** or **smit** to change the network_fdt and/or the node_timeout tunable.

Example 4-14 Cluster heartbeat settings

Cluster heartbeat settings

Type or select values in entry fields.

Press Enter AFTER making all desired 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	[0]	#	
* LPM Node Policy	[manage]	+	

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

The **c1mgr** command to change the FDT setting is the following:

```
c1mgr modify cluster NETWORK_FAILURE_DETECTION_TIME=10
```

Remember when you increase the FDT value you also have to increase value for the Node Failure Detection Timeout. The minimum delta between these two values must be 10 seconds. So if we would like to change the FDT to 40 seconds and like to keep the minimum delta the command would look like this:

```
c1mgr modify cluster NETWORK_FAILURE_DETECTION_TIME=40 HEARTBEAT_FREQUENCY=50
```

4.2 Automatic repository update for the repository disk

This section describes the new PowerHA ARU feature for the PowerHA repository disk.

4.2.1 Introduction to the Automatic Repository Update

Starting with PowerHA V7.0.0, PowerHA uses a shared disk, which is called the *PowerHA repository disk*, for various purposes. The availability of this repository disk is critical to the operation of PowerHA clustering and its nodes. The initial implementation of the repository disk, at PowerHA V7.0.0, did not allow for the operation of PowerHA cluster services if the repository disk failed, making that a single point of failure (SPOF).

With later versions of PowerHA, features were added to make the cluster more resilient if there is a PowerHA repository disk failure. The ability to survive a repository disk failure, in addition to the ability to manually replace a repository disk without an outage, increased the resiliency of PowerHA. With PowerHA V7.2.0, a new feature to increase the resiliency further was introduced, and this is called ARU.

If there is an active repository disk failure, the purpose of ARU is to automate the replacement of a PowerHA repository disk without intervention from a system administrator and without affecting the active cluster services. All that is needed is to point PowerHA to the backup repository disks to use if there is an active repository disk failure.

If a repository disk fails, PowerHA detects the failure of the active repository disk. At that point, it verifies that the active repository disk is not usable. If the active repository disk is unusable, it attempts to switch to the backup repository disk. If it is successful, then the backup repository disk becomes the active repository disk.

4.2.2 Requirements for Automatic Repository Update

ARU has the following requirements:

- ▶ AIX 7.1.4 or AIX 7.2.0.
- ▶ PowerHA V7.2.0.
- ▶ The storage that is used for the backup repository disk has the same requirements as the primary repository disk.

For more information about the PowerHA repository disk requirements, see [IBM Knowledge Center](#).

4.2.3 Configuring Automatic Repository Update

The configuration of ARU is automatic when you configure a backup repository disk for PowerHA. Essentially, all you must do is configure a backup repository disk.

This section shows an example of ARU in a 2-site, 2-node cluster. The cluster configuration is similar to Figure 4-1.

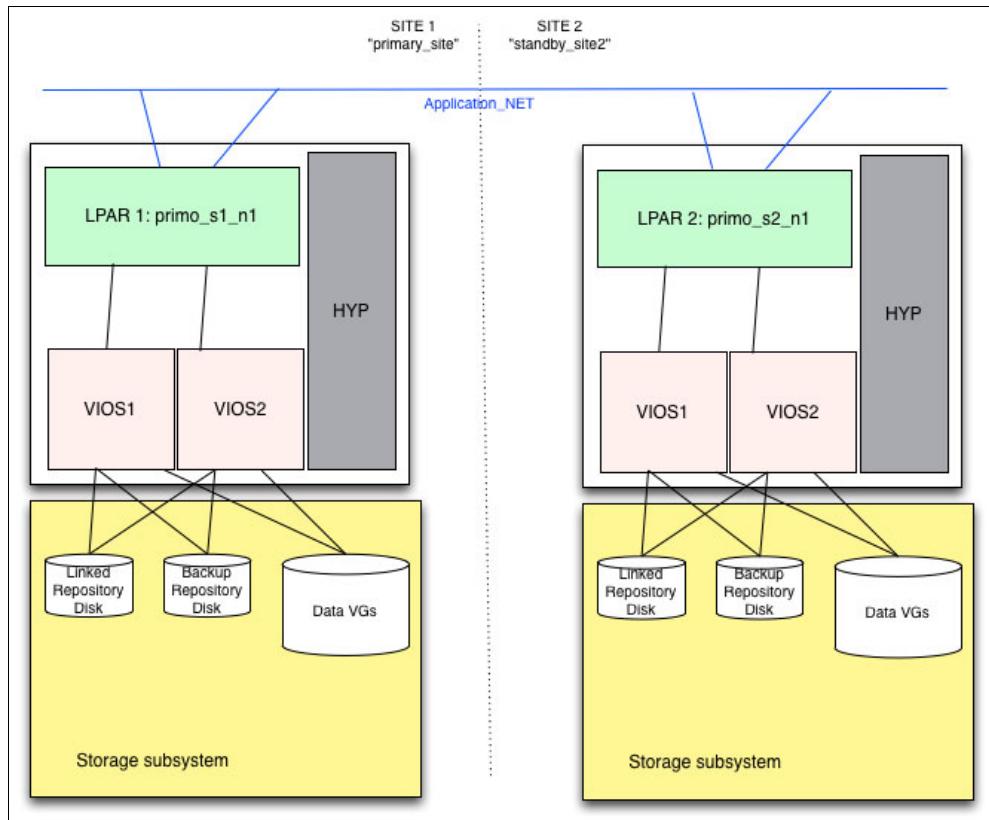


Figure 4-1 Storage example for PowerHA ARU showing linked and backup repository disks

For the purposes of this example, we configure a backup repository disk for each site of this 2-site cluster.

Configuring a backup repository disk

The following process details how to configure a backup repository disk. For our example, we perform this process for each site in our cluster.

1. Using SMIT, run **smitty sysmirror** and select **Cluster Nodes and Networks → Cluster Nodes and Networks → Add a Repository Disk**. You are prompted for a site due to the fact that our example is a 2-site cluster, and then given a selection of possible repository disks. The panels that are shown in the following sections provide more details.

When you select **Add a Repository Disk**, you are prompted to select a site, as shown in Example 4-15.

Example 4-15 Selecting Add a Repository Disk in a multi-site cluster

Manage Repository Disks

Move cursor to desired item and press Enter.

Add a Repository Disk

Remove a Repository Disk
Show Repository Disks

Verify and Synchronize Cluster Configuration

Select a Site		
Move cursor to desired item and press Enter.		
primary_site1 standby_site2		
F1=Help F1 / =Find F9+-----	F2=Refresh F8=Image n=Find Next	F3=Cancel Enter=Do

2. After selecting **primary_site1**, Example 4-16 shows the repository disk menu.

Example 4-16 Add a Repository Disk panel

Add a Repository Disk

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

Site Name		[Entry Fields]
* Repository Disk		primary_site1 []
F1=Help F5=Reset F9=Shell	F2=Refresh F6=Command F10=Exit	F3=Cancel F7>Edit Enter=Do
F4>List F8=Image		

3. Next, press F4 on the Repository Disk field, and you are shown the repository disk selection list, as shown in Example 4-17.

Example 4-17 Backup repository disk selection

Add a Repository Disk

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

Site Name * Repository Disk	[Entry Fields] primary_site1 [] +			
<hr/>				
Repository Disk				
Move cursor to desired item and press F7. ONE OR MORE items can be selected. Press Enter AFTER making all selections.				
hdisk3 (00f61ab295112078) on all nodes at site primary_site1 hdisk4 (00f61ab2a61d5bc6) on all nodes at site primary_site1 hdisk5 (00f61ab2a61d5c7e) on all nodes at site primary_site1				
<table border="0"><tr><td style="width: 33%;">F1=Help F7=Select F5 Enter=Do F9+</td><td style="width: 33%;">F2=Refresh F8=Image /Find</td><td style="width: 33%;">F3=Cancel F10=Exit n=Find Next</td></tr></table>		F1=Help F7=Select F5 Enter=Do F9+	F2=Refresh F8=Image /Find	F3=Cancel F10=Exit n=Find Next
F1=Help F7=Select F5 Enter=Do F9+	F2=Refresh F8=Image /Find	F3=Cancel F10=Exit n=Find Next		

4. After selecting the appropriate disk, the choice is shown in Example 4-18.

Example 4-18 Add a Repository Disk preview panel

Add a Repository Disk

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

Site Name * Repository Disk	[Entry Fields] primary_site1 [(00f61ab295112078)] +
<hr/>	
F1=Help F2=Refresh F3=Cancel F4=List F5=Reset F6=Command F7>Edit F8=Image F9=Shell F10=Exit Enter=Do	

5. Next, after pressing the Enter key to make the changes, the confirmation panel appears, as shown in Example 4-19.

Example 4-19 Backup repository disk addition confirmation panel

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

Successfully added one or more backup repository disks.
To view the complete configuration of repository disks use:
"clmgr query repository" or "clmgr view report repository"

F1=Help
F8=Image

F2=Refresh
F9=Shell

F3=Cancel
F10=Exit

F6=Command
/=Find

4.2.4 Automatic Repository Update operations

PowerHA ARU operations are automatic when a backup repository disk is configured.

Successful Automatic Repository Update operation

ARU operations are automatic when a backup repository disk is defined. Our scenario has a 2-site cluster and a backup repository disk per site.

To induce a failure of the primary repository disk, we log in to the Virtual I/O Server (VIOS) servers that present storage to the cluster LPARs and deallocate the disk LUN that corresponds to the primary repository disk on one site of our cluster. This disables the primary repository disk, and PowerHA ARU detects the failure and automatically activates the backup repository disk as the active repository disk.

This section presents the following examples that are during this process:

1. Before disabling the primary repository disk, we look at the **lspv** command output and note that the active repository disk is hdisk1, as shown in Example 4-20.

Example 4-20 Output of the lspv command in an example cluster

hdisk0	00f6f5d09570f647	rootvg	active
hdisk1	00f6f5d0ba49cdcc	caavg_private	active
hdisk2	00f6f5d0a621e9ff	None	
hdisk3	00f61ab2a61d5c7e	None	
hdisk4	00f61ab2a61d5d81	testvg01	concurrent
hdisk5	00f61ab2a61d5e5b	testvg01	concurrent
hdisk6	00f61ab2a61d5f32	testvg01	concurrent

2. We then proceed to log in to the VIOS servers that present the repository disk to this logical partition (LPAR) and de-allocate that logical unit (LUN) so that the cluster LPAR no longer has access to that disk. This causes the primary repository disk to fail.

- PowerHA ARU detects the failure and activates the backup repository disk as the active repository disk. You can verify this behavior in the `syslog.caa` log file. This log file logs the ARU activities and shows the detection of the primary repository disk failure and the activation of the backup repository disk. See Example 4-21.

Example 4-21 The /var/adm/ras/syslog.caa file showing repository disk failure and recovery

```

Nov 12 09:13:29 primo_s2_n1 caa:info cluster[14025022]: caa_config.c
run_list      1377   1      = = END REPLACE_REPO Op = = POST Stage = =
Nov 12 09:13:30 primo_s2_n1 caa:err|error cluster[14025022]: cluster_utils.c
cluster_repository_read 5792   1      Could not open cluster repository
device /dev/rhdisk1: 5
Nov 12 09:13:30 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_kern_repos_check  11769   1      Could not read the respository.
Nov 12 09:13:30 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_run_log_method  11862   1      START '/usr/sbin/importvg -y
caavg_private_t -0 hdisk1'
Nov 12 09:13:32 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_run_log_method  11893   1      FINISH return = 1
Nov 12 09:13:32 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_run_log_method  11862   1      START '/usr/sbin/reducevg -df
caavg_private_t hdisk1'
Nov 12 09:13:32 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_run_log_method  11893   1      FINISH return = 1
Nov 12 09:13:33 primo_s2_n1 caa:err|error cluster[14025022]: cluster_utils.c
cluster_repository_read 5792   1      Could not open cluster repository
device /dev/rhdisk1: 5
Nov 12 09:13:33 primo_s2_n1 caa:info cluster[14025022]: cl_chrepos.c
destroy_old_repository 344   1      Failed to read repository data.
Nov 12 09:13:34 primo_s2_n1 caa:err|error cluster[14025022]: cluster_utils.c
cluster_repository_write 5024   1      return = -1, Could not open
cluster repository device /dev/rhdisk1: I/O error
Nov 12 09:13:34 primo_s2_n1 caa:info cluster[14025022]: cl_chrepos.c
destroy_old_repository 350   1      Failed to write repository data.
Nov 12 09:13:34 primo_s2_n1 caa:warn|warning cluster[14025022]: cl_chrepos.c
destroy_old_repository 358   1      Unable to destroy repository disk
hdisk1. Manual intervention is required to clear the disk of cluster
identifiers.
Nov 12 09:13:34 primo_s2_n1 caa:info cluster[14025022]: cl_chrepos.c
automatic_repository_update 2242   1      Replaced hdisk1 with hdisk2
Nov 12 09:13:34 primo_s2_n1 caa:info cluster[14025022]: cl_chrepos.c
automatic_repository_update 2255   1      FINISH rc = 0
Nov 12 09:13:34 primo_s2_n1 caa:info cluster[14025022]: caa_protocols.c
recv_protocol_slave 1542   1      Returning from Automatic Repository
replacement rc = 0

```

- As an extra verification, the AIX error log has an entry showing that a successful repository disk replacement occurred, as shown in Example 4-22.

Example 4-22 AIX error log showing successful repository disk replacement message

LABEL:	CL_ARU_PASSED
IDENTIFIER:	92EE81A5

Date/Time:	Thu Nov 12 09:13:34 2015
Sequence Number:	1344
Machine Id:	00F6F5D04C00

```

Node Id:      primo_s2_n1
Class:       H
Type:        INFO
WPAR:        Global
Resource Name: CAA ARU
Resource Class: NONE
Resource Type: NONE
Location:

Description
Automatic Repository Update succeeded.

Probable Causes
Primary repository disk was replaced.

Failure Causes
A hardware problem prevented local node from accessing primary repository disk.

Recommended Actions
Primary repository disk was replaced using backup repository disk.

Detail Data
Primary Disk Info
hdisk1 6c1b76e1-3e0a-ff3c-3c43-cb6c3881c3bf
Replacement Disk Info
hdisk2 5890b139-e987-1451-211e-24ba89e7d1df

```

It is safe to remove the failed repository disk and replace it. The replacement disk can become the new backup repository disk by following the steps in “Configuring a backup repository disk” on page 84.

Possible ARU failure situations

Some activities can affect the operation of ARU. Specifically, any administrative activity that uses the backup repository disk can affect ARU. If a volume group (VG) was previously created on a backup repository disk and this disk was not *cleaned up*, then ARU cannot operate properly.

In our sample scenario, we complete the following steps:

1. Configure a backup repository disk that previously had an AIX volume group (VG).
2. Export the AIX VG so that the disk did not display a VG by using the AIX command **1spv**. However, we did not delete that VG from the disk, so the disk itself still had that information.
3. For our example, we ran the AIX command **1spv**. Our backup repository disk is hdisk2. The disk shows a PVID but no VG, as shown in Example 4-23.

Example 4-23 Output of lsvp command in an example cluster showing hdisk2

hdisk0	00f6f5d09570f647	rootvg	active
hdisk1	00f6f5d0ba49cdcc	caavg_private	active
hdisk2	00f6f5d0a621e9ff	None	
hdisk3	00f61ab2a61d5c7e	None	
hdisk4	00f61ab2a61d5d81	testvg01	concurrent
hdisk5	00f61ab2a61d5e5b	testvg01	concurrent
hdisk6	00f61ab2a61d5f32	testvg01	concurrent

- We disconnect the primary repository disk from the LPAR by going to the VIOS and de-allocating the disk LUN from the cluster LPAR. This made the primary repository disk *fail* immediately.

ARU attempts to perform the following actions:

- Checks the primary repository disk that is not accessible.
- Switches to the backup repository disk (but this action fails).

- ARU leaves an error message in the AIX error report, as shown in Example 4-24.

Example 4-24 Output of the AIX errpt command showing failed repository disk replacement

```
LABEL:          CL_ARU_FAILED
IDENTIFIER:     F63D60A2

Date/Time:      Wed Nov 11 17:15:17 2015
Sequence Number: 1263
Machine Id:    00F6F5D04C00
Node Id:       primo_s2_n1
Class:         H
Type:          INFO
WPAR:          Global
Resource Name: CAA ARU
Resource Class: NONE
Resource Type:  NONE
Location:

Description
Automatic Repository Update failed.

Probable Causes
Unknown.

Failure Causes
Unknown.

Recommended Actions
Try manual replacement of cluster repository disk.

Detail Data
Primary Disk Info
hdisk1 6c1b76e1-3e0a-ff3c-3c43-cb6c3881c3bf
```

- In addition, we note that ARU verifies the primary repository disk and fails, as shown in the CAA log /var/adm/ras/syslog.caa in Example 4-25.

Example 4-25 Selected messages from the /var/adm/ras/syslog.caa log file

```
Nov 12 09:13:20 primo_s2_n1 caa:info unix: *base_kernext_services.c  aha_thread_queue
614      The AHAFS event is EVENT_TYPE=REP_DOWN DISK_NAME=hdisk1 NODE_NUMBER=2
NODE_ID=0xD9DDB48A889411E580106E8DDDB7B3702 SITE_NUMBER=2
SITE_ID=0xD9DE2028889411E580106E8DDDB7B3702 CLUSTER_ID=0xD34E8658889411E580026E8DD
Nov 12 09:13:20 primo_s2_n1 caa:info unix: caa_sock.c  caa_kclient_tcp 231
entering caa_kclient_tcp ....
Nov 12 09:13:20 primo_s2_n1 caa:info unix: *base_kernext_services.c  aha_thread_queue
614      The AHAFS event is EVENT_TYPE=VG_DOWN DISK_NAME=hdisk1 VG_NAME=caavg_private
NODE_NUMBER=2 NODE_ID=0xD9DDB48A889411E580106E8DDDB7B3702 SITE_NUMBER=2
SITE_ID=0xD9DE2028889411E580106E8DDDB7B3702 CLUSTER_ID=0xD34E8
```

```
Nov 12 09:13:20 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_run_log_method      11862  1      START '/usr/lib/cluster/caa_syslog'
Nov 12 09:13:20 primo_s2_n1 caa:info unix: kcluster_event.c      find_event_disk 742
Find disk called for hdisk4
Nov 12 09:13:20 primo_s2_n1 caa:info unix: kcluster_event.c
ahafs_Disk_State_register      1504      diskState set opqId = 0xF1000A0150301A00
Nov 12 09:13:20 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_run_log_method      11893  1      FINISH return = 0
Nov 12 09:13:20 primo_s2_n1 caa:info cluster[14025022]: caa_message.c
inherit_socket_inetd      930      1      IPv6=:ffff:127.0.0.1
Nov 12 09:13:20 primo_s2_n1 caa:info cluster[14025022]: cluster_utils.c
cl_kern_repos_check      11769  1      Could not read the repository.
Nov 12 09:13:20 primo_s2_n1 caa:info cluster[14025022]: caa_message.c  cl_recv_req
172      1      recv successful, sock = 0, recv rc = 32, msgbytes = 32
Nov 12 09:13:20 primo_s2_n1 caa:info cluster[14025022]: caa_protocols.c
recv_protocol_slave      1518      1      Automatic Repository Replacement request being
processed.
```

- ARU attempts to activate the backup repository disk, but it fails due to the fact that an AIX VG previously existed in this disk, as shown in Example 4-26.

Example 4-26 Messages from the /var/adm/ras/syslog.caa log file showing an ARU failure

Recovering from a failed ARU event

In “Possible ARU failure situations” on page 88, we used an example about what can prevent a successful repository disk replacement by using ARU. To recover from that failed event, we manually switch the repository disks by using the PowerHA SMIT panels.

Complete the following steps:

1. Using SMIT, run **smitty sysmirror** and select **Problem Determination Tools → Replace the Primary Repository Disk**. In our sample cluster, we have multiple sites, so we select a site, as shown in Example 4-27.

Example 4-27 Site selection prompt after selecting “Replace the Primary Repository Disk”

Problem Determination Tools

Move cursor to desired item and press Enter.

[MORE...1]

[View Current State](#)

PowerHA SystemMirror Log Viewing and Management

```

Recover From PowerHA SystemMirror Script Failure
Recover Resource Group From SCSI Persistent Reserve Error
Restore PowerHA SystemMirror Configuration Database from Active Configuration
Release Locks Set By Dynamic Reconfiguration
Cluster Test Tool
+-----+
|                               Select a Site
| Move cursor to desired item and press Enter.
| primary_site1
| standby_site2
|M F1=Help          F2=Refresh        F3=Cancel
|M F8=Image          F10=Exit          Enter=Do
F1  /=Find           n=Find Next
F9+-----+

```

2. In our example, we select **standby_site2** and a panel opens with an option to select the replacement repository disk, as shown in Example 4-28.

Example 4-28 Prompt to select a new repository disk

Select a new repository disk

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

Site Name * Repository Disk	[Entry Fields] standby_site2 <input type="checkbox"/>	+
--------------------------------	---	-------

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

3. Pressing the F4 key shows the available backup repository disks, as shown in Example 4-29.

Example 4-29 SMIT menu prompting for replacement repository disk

Select a new repository disk

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

Site Name * Repository Disk	[Entry Fields] standby_site2 <input type="checkbox"/>	+
--------------------------------	---	-------

+-----+ Repository Disk Move cursor to desired item and press Enter.
--

	00f6f5d0ba49cdcc		
F1	F1=Help F8=Image F5 /=Find F9+-----+	F2=Refresh F10=Exit n=Find Next	F3=Cancel Enter=Do

4. Selecting the backup repository disk opens the SMIT panel showing the selected disk, as shown in Example 4-30.

Example 4-30 SMIT panel showing the selected repository disk

Select a new repository disk

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

Site Name	[Entry Fields]
* Repository Disk	standby_site2
	[00f6f5d0ba49cdcc]
	+-----+

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

5. Last, pressing the Enter key runs the repository disk replacement. After the repository disk is replaced, the panel that is shown in Example 4-31 opens.

Example 4-31 SMIT panel showing a successful repository disk replacement

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

chrepos: Successfully modified repository disk or disks.

New repository "hdisk1" (00f6f5d0ba49cdcc) is now active.
The configuration must be synchronized to make this change known across the cluster.

F1=Help F8=Image n=Find Next	F2=Refresh F9=Shell	F3=Cancel F10=Exit	F6=Command /=Find
------------------------------------	------------------------	-----------------------	----------------------

Now, it is safe to remove the failed repository disk and replace it. The replacement disk can become the new backup repository disk by following the steps that are described in “Configuring a backup repository disk” on page 84.

4.3 New manage option to start PowerHA

Starting with PowerHA V7.2.1, you can use an additional management option to start the cluster. The new argument for the option is named **delayed**:

clmgr online cluster manage=delayed

When this option is used, all cluster managers are initialized and then all RGs are started. Under the covers the system does a manual cluster start first and then activates all resource groups.

This option is very helpful if you have start dependencies defined and these dependencies are across more than one node.

Figure 4-2 shows what can happen if RG dependencies are defined and the cluster gets started by using automatic. The left side shows what we intend to achieve and the right side shows what can happen. This example defines three RGs with a start after dependency. All nodes have the same startup policy but different home nodes.

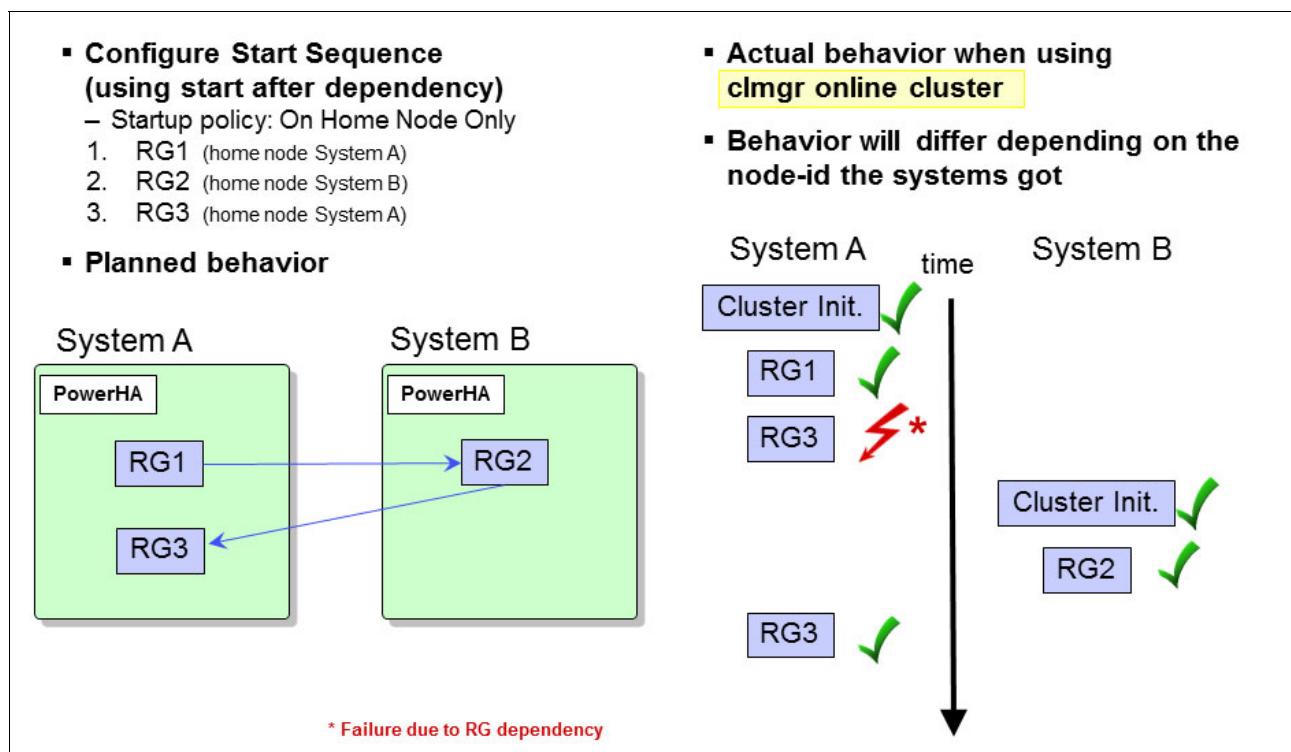


Figure 4-2 Normal cluster start with RG dependencies

Figure 4-3 on page 94 shows how the cluster start happens when the new delayed option is used. Compared to the example in Figure 4-2 nothing changes from a configuration point of view. The only difference is that the cluster has started with the `manage=delayed` option.

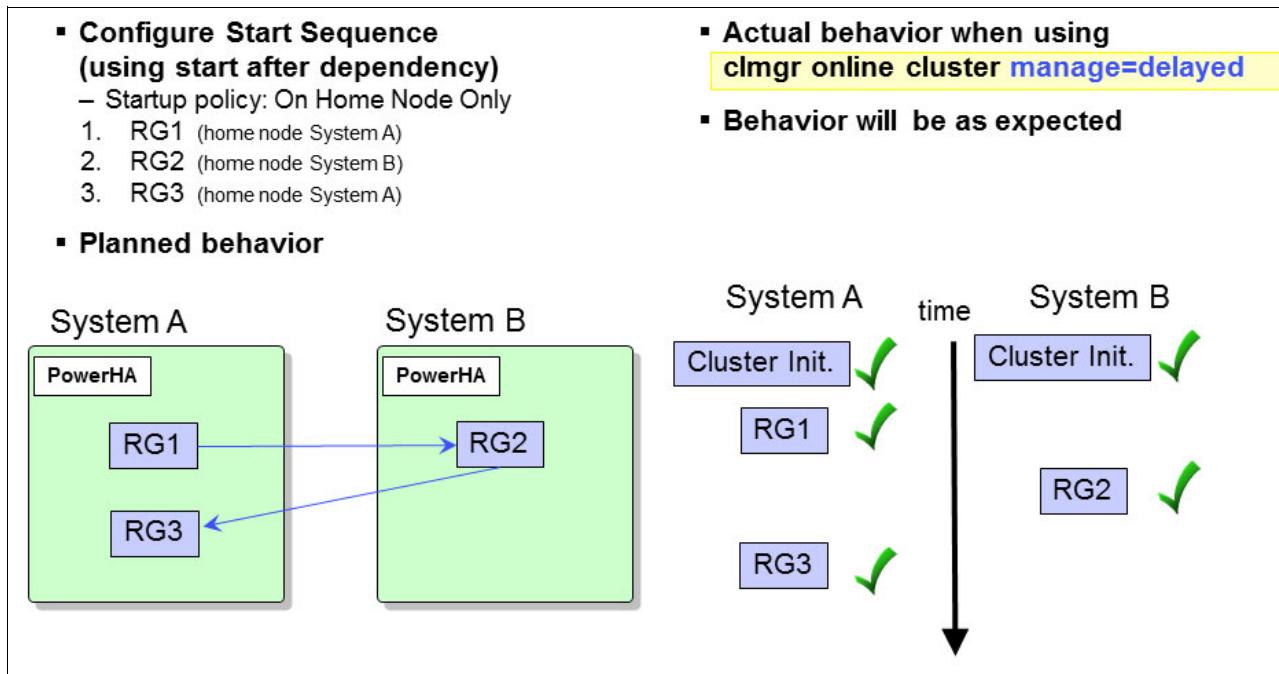


Figure 4-3 Cluster start using manage option delayed

4.4 Reliable Scalable Cluster Technology overview

This section provides an overview of Reliable Scalable Cluster Technology (RSCT), its components, and the communication path between these components. This section also describes what parts of it are used by PowerHA. The items that are described here are not new but are needed for a basic understanding of the PowerHA underlying infrastructure.

4.4.1 What Reliable Scalable Cluster Technology is

RSCT is a set of software components that provide a comprehensive clustering environment for AIX, Linux, Solaris, and Microsoft Windows operating systems. RSCT is the infrastructure that is used by various IBM products to provide clusters with improved system availability, scalability, and ease of use.

4.4.2 Reliable Scalable Cluster Technology components

This section describes the RSCT components and how they communicate with each other.

Reliable Scalable Cluster Technology components overview

For a more detailed description of the RSCT components, see *IBM RSCT for AIX: Guide and Reference*, SA22-7889.

The main RSCT components are explained in this section:

- ▶ Resource Monitoring and Control (RMC) subsystem

This is the scalable and reliable backbone of RSCT. RMC runs on a single machine or on each node (operating system image) of a cluster, and provides a common abstraction for the resources of the individual system or the cluster of nodes. You can use RMC for a

single system monitoring, or for monitoring nodes in a cluster. However, in a cluster, RMC provides global access to subsystems and resources throughout the cluster, thus providing a single monitoring and management infrastructure for clusters.

- ▶ RSCT core resource managers (RMs)

A *resource manager* is a software layer between a resource (a hardware or software entity that provides services to some other component) and RMC. An RM maps programmatic abstractions in RMC into the actual calls and commands of a resource.

- ▶ RSCT cluster security services

This RSCT component provides the security infrastructure that enables RSCT components to authenticate the identity of other parties.

- ▶ Group Services subsystem

This RSCT component provides cross-node/process coordination on some cluster configurations.

- ▶ Topology Services subsystem

This RSCT component provides node and network failure detection on some cluster configurations.

Communication between RSCT components

The RMC subsystem and RSCT core RMs are today the only ones that use the RSCT cluster security services. Since the availability of PowerHA V7, RSCT Group Services are able to use Topology Services or CAA. Figure 4-4 shows the RSCT components and their relationships.

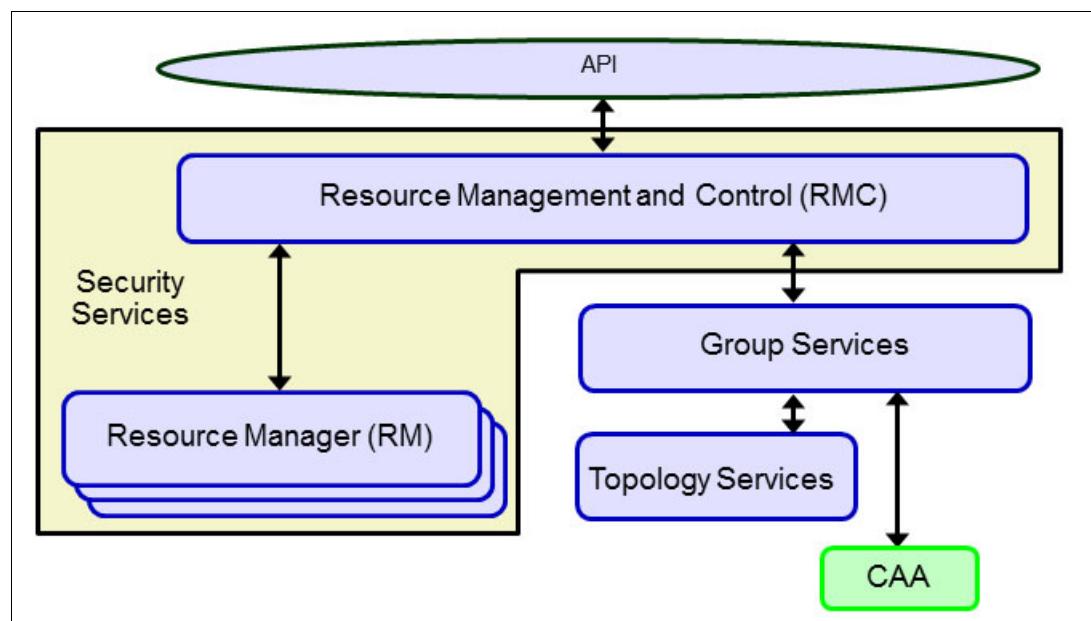


Figure 4-4 RSCT components

The RMC application programming interface (API) is the only interface that can be used by applications to exchange data with the RSCT components. RMC manages the RMs and receives data from them. Group Services is a client of RMC. Depending on whether PowerHA V7 is installed, it connects to CAA. Otherwise, it connects to the RSCT Topology Services.

RSCT domains

An RSCT management domain is a set of nodes with resources that can be managed and monitored from one of the nodes, which is designated as the *management control point* (MCP). All other nodes are considered to be managed nodes. Topology Services and Group Services are not used in a management domain. Example 4-5 shows the high-level architecture of an RSCT management domain.

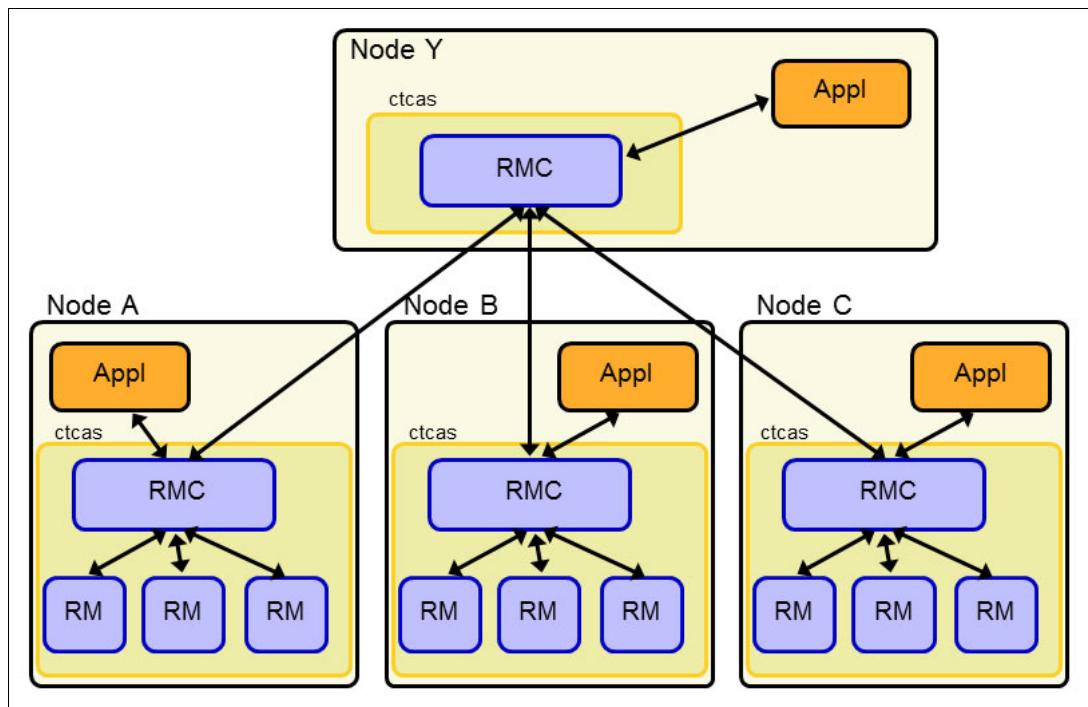


Figure 4-5 RSCT-managed domain (architecture)

An RSCT *peer domain* is a set of nodes that have a consistent knowledge of the existence of each other, and of the resources shared among them. On each node within the peer domain, RMC depends on a core set of cluster services, which include Topology Services, Group Services, and cluster security services. Figure 4-6 shows the high-level architecture of an RSCT peer domain.

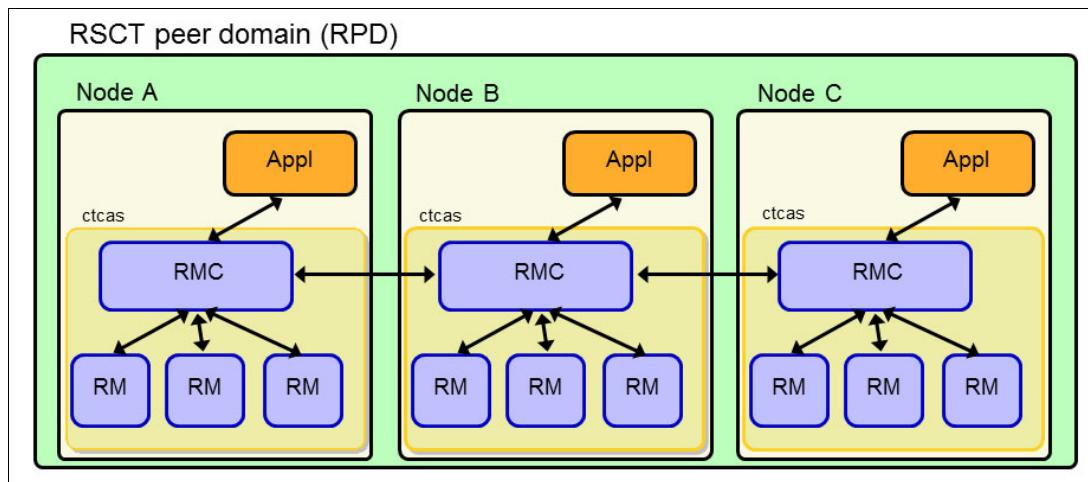


Figure 4-6 RSCT peer domain (architecture)

Group Services are used in peer domains. If PowerHA V7 is installed, Topology Services are not used, and CAA is used instead. Otherwise, Topology Services are used too.

Combination of management and peer domains

You can have a combination of both types of domains (management domain and peer domains).

Figure 4-7 shows the high-level architecture for how an RSCT-managed domain and RSCT peer domains can be combined. In this example, Node Y is an RSCT management server. You have three nodes as managed nodes (Node A, Node B, and Node C). Node B and Node C are part of an RSCT peer domain.

You can have multiple peer domains within a managed domain. A node can be part of a managed domain and a peer domain. A given node can belong to only a single peer domain, as shown in Figure 4-7.

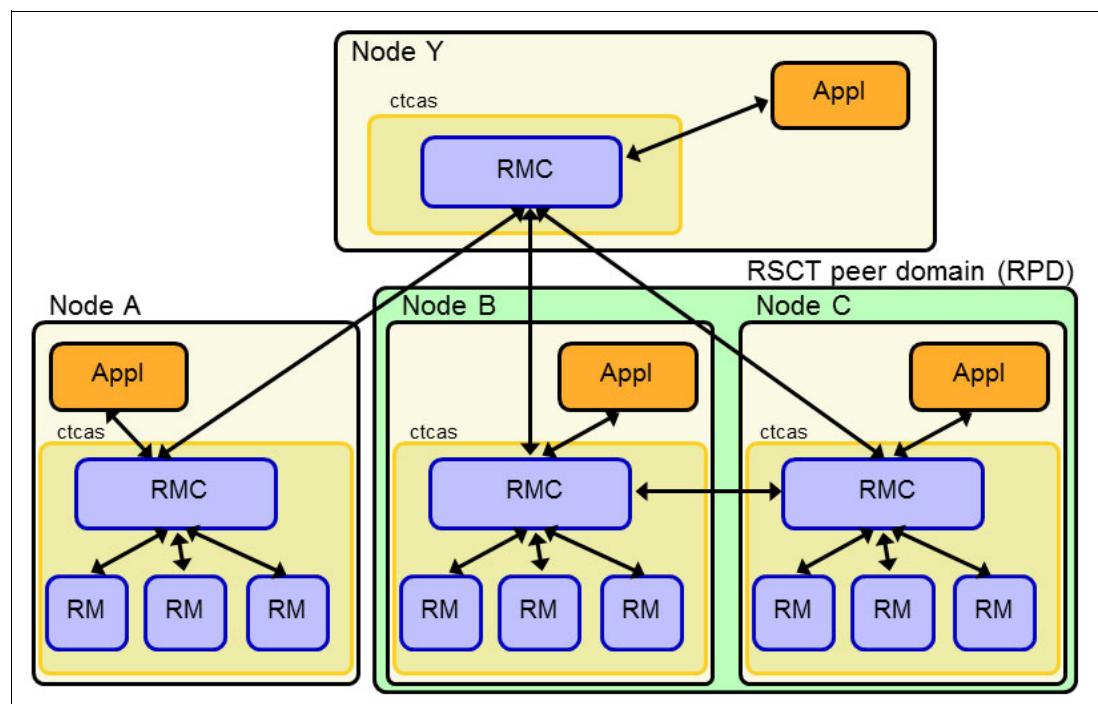


Figure 4-7 Management and peer domain (architecture)

Important: A node can belong to only one RSCT peer domain.

Example of a management and a peer domain

The example here is simplified. It just shows one Hardware Management Console (HMC) that is managing three LPARS, where two of them are used for a 2-node PowerHA cluster.

In a Power Systems environment, the HMC is always the management server in the RSCT management domain. The LPARs are clients to this server from an RSCT point of view. For example, this management domain is used to do dynamic LPAR (DLPAR) operations on the different LPARs.

Figure 4-8 shows this simplified setup.

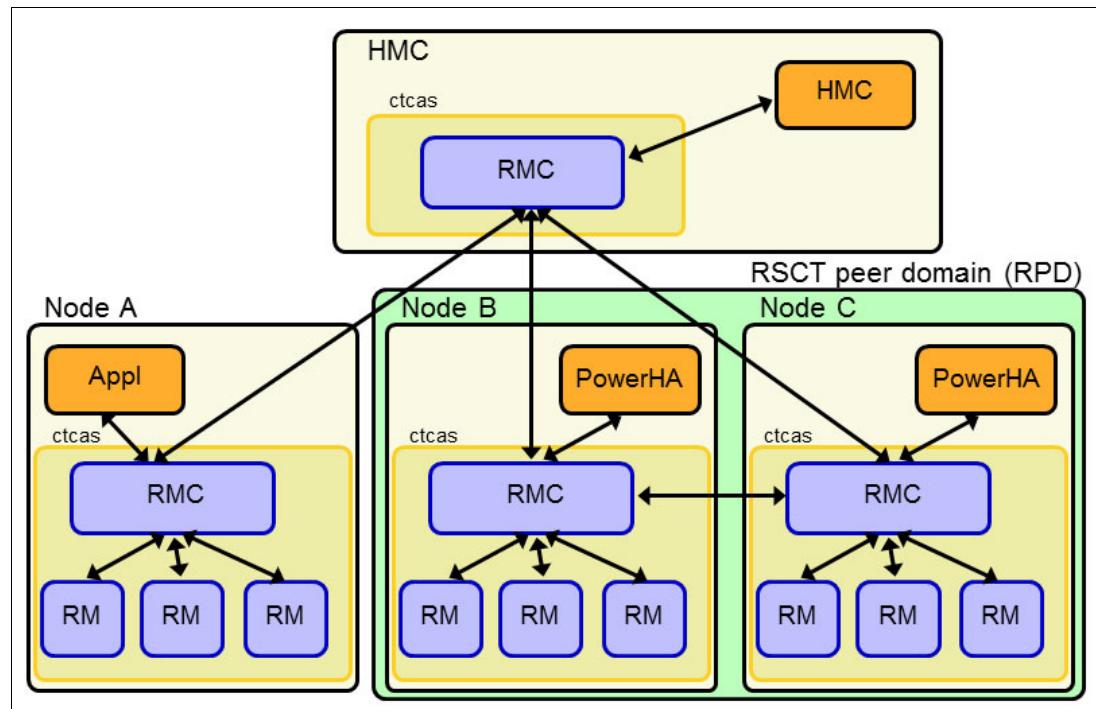


Figure 4-8 Example management and peer domain

RSCT peer domain on Cluster Aware AIX

When RSCT operates on nodes in a CAA cluster, a peer domain is created that is equivalent to the CAA cluster. This RSCT peer domain presents largely the same set of function to users and software as other peer domains not based on CAA. Consider a peer domain, which is operating without CAA, and autonomously manages and monitors the configuration and liveness of the nodes and interfaces that it comprises.

The peer domain that represents a CAA cluster acquires configuration information and liveness results from CAA. It introduces some differences in the mechanics of peer domain operations, but few in the view of the peer domain that is available to the users.

Only one CAA cluster can be defined on a set of nodes. Therefore, if a CAA cluster is defined, the peer domain that represents it is the only peer domain that can exist, and it exists and is online for the life of the CAA cluster.

Figure 4-9 illustrates the relationship that is described in this section.

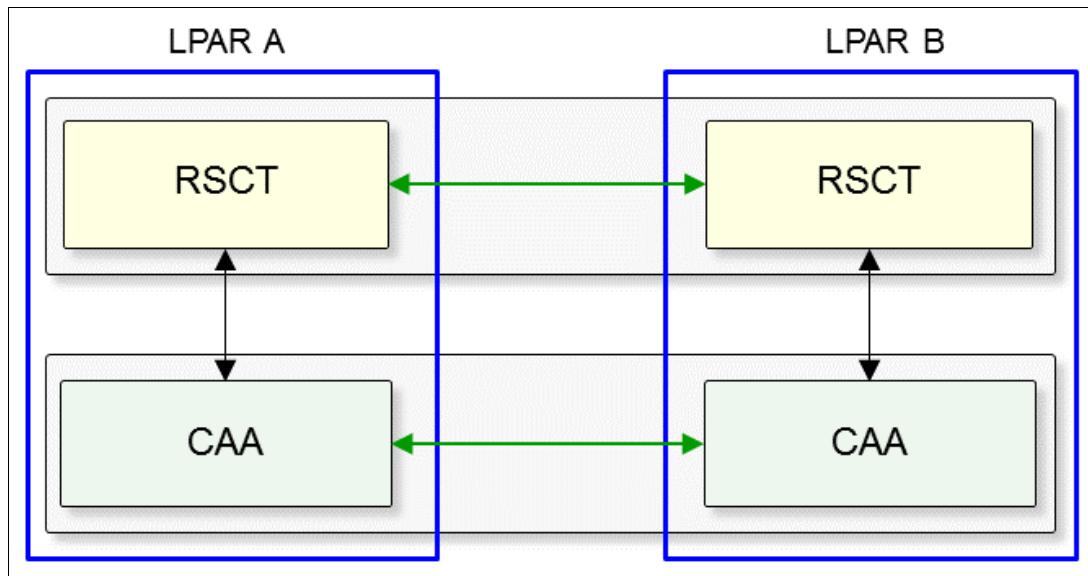


Figure 4-9 RSCT peer domain and CAA

When your cluster is configured and synchronized, you can check the RSCT peer domain by using the **1srpdomain** command. To list the nodes in this peer domain, you can use the **1srpnode** command. Example 4-32 shows a sample output of these commands.

Example 4-32 Listing RSCT peer domain information

```
# 1srpdomain
Name          OpState RSCTActiveVersion MixedVersions TSPort GSPort
c2n1_cluster  Online  3.1.5.0           Yes        12347  12348
# 1srpnode
1srpnode
Name          OpState RSCTVersion
c2n2.munich.de.ibm.com  Online  3.2.1.0
c2n1.munich.de.ibm.com  Online  3.2.1.0
#
```

The RSCTActiveVersion number of the **1srpdomain** output can show a back-level version number. This is the lowest RSCT version that is required by a new joining node. In a PowerHA environment, there is no need to modify this number.

The value of yes for MixedVersions means that you have at least one node with a higher version than the displayed RSCT version. The **1srpnode** command lists the used RSCT version by node.

Updating the RSCT peer domain version

If you like, you can upgrade the RSCT version of the RSCT peer domain, which is reported by the **1srpdomain** command. To do this, use the command that is listed in Example 4-33.

Example 4-33 Updating the RSCT peer domain

```
# export CT_MANAGEMENT_SCOPE=2; runact -c IBM.PeerDomain \
CompleteMigration Options=0
#
```

To be clear, doing such an update does not give you any advantages in a PowerHA environment. In fact, if you delete the cluster and then re-create it manually, or by using an existing snapshot of the RSCT peer domain version, you are back to the original version, which was 3.1.5.0 in our example.

Checking for Cluster Aware AIX

To do a quick check on the CAA cluster, you can use the **lscluster -c** command or use the **lscluster -m** command. Example 4-34 shows an example output of these two commands. For most situations, when you get an output of the **lscluster** command, CAA is running. To be on the safe side, use the **lscluster -m** command.

Example 4-34 shows that in our case CAA is running on the local node where we used the **lscluster** command. But, on the remote node CAA was stopped.

To stop CAA, we use the **clmgr off node powerha-c2n2 STOP_CAA=yes** command.

Example 4-34 The lscluster -c and lscluster -m commands

```
# lscluster -c
Cluster Name: c2n1_cluster
Cluster UUID: d19995ae-8246-11e5-806f-fa37c4c10c20
Number of nodes in cluster = 2
    Cluster ID for node c2n1.munich.de.ibm.com: 1
    Primary IP address for node c2n1.munich.de.ibm.com: 172.16.150.121
    Cluster ID for node c2n2.munich.de.ibm.com: 2
    Primary IP address for node c2n2.munich.de.ibm.com: 172.16.150.122
Number of disks in cluster = 1
    Disk = caa_r0 UUID = 12d1d9a1-916a-ceb2-235d-8c2277f53d06 cluster_major =
0 cluster_minor = 1
Multicast for site LOCAL: IPv4 228.16.150.121 IPv6 ff05::e410:9679
Communication Mode: unicast
Local node maximum capabilities: AUTO_REPOS_REPLACE, HNAME_CHG, UNICAST, IPV6,
SITE
Effective cluster-wide capabilities: AUTO_REPOS_REPLACE, HNAME_CHG, UNICAST, IPV6,
SITE
#
# lscluster -m | egrep "Node name|State of node"
    Node name: powerha-c2n1.munich.de.ibm.com
    State of node: DOWN
    Node name: powerha-c2n2.munich.de.ibm.com
    State of node: UP NODE_LOCAL
#
```

Peer domain on CAA linked clusters

Starting with PowerHA V7.1.2, linked clusters can be used. An RSCT peer domain that operates on linked clusters encompasses all nodes at each site. The nodes that comprise each site cluster are all members of the same peer domain.

Figure 4-10 shows how this looks from an architecture point of view.

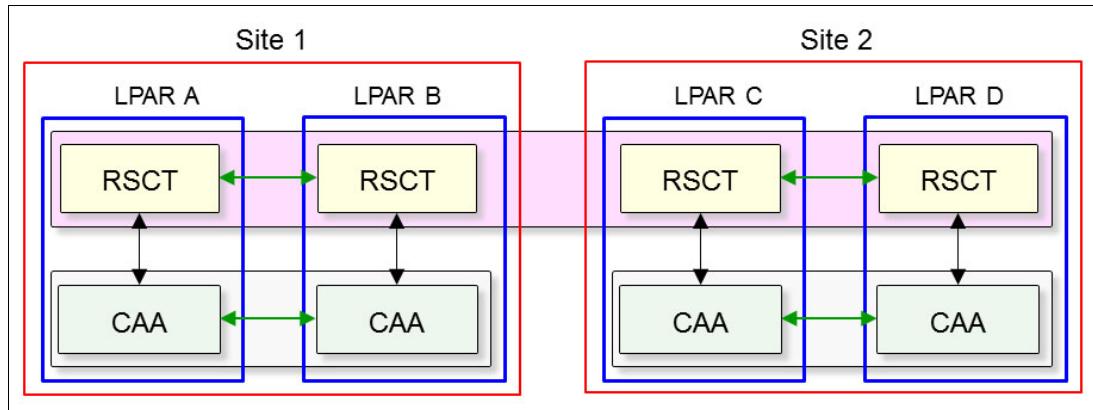


Figure 4-10 RSCT peer domain and CAA linked cluster

Example 4-35 shows what the RSCT looks like in our 2-node cluster.

Example 4-35 Output of the lsrpdomain command

```

# lsrpdomain
Name          OpState RSCTActiveVersion MixedVersions TSPort GSPort
primo_s1_n1_cluster Online  3.1.5.0      Yes           12347  12348
# lsrpnode
Name          OpState RSCTVersion
primo_s2_n1  Online  3.2.1.0
primo_s1_n1  Online  3.2.1.0
#

```

Because we define each of our nodes to a different site, the **lscluster -c** command lists only one node. Example 4-36 shows an example output from node 1.

Example 4-36 Output of the lscluster command (node 1)

```

# lscluster -c
Cluster Name: primo_s1_n1_cluster
Cluster UUID: d34e8658-8894-11e5-8002-6e8ddb7b3702
Number of nodes in cluster = 2
    Cluster ID for node primo_s1_n1: 1
    Primary IP address for node primo_s1_n1: 192.168.100.20
    Cluster ID for node primo_s2_n1: 2
    Primary IP address for node primo_s2_n1: 192.168.100.21
Number of disks in cluster = 4
    Disk = hdisk2 UUID = 2f1b2492-46ca-eb3b-faf9-87fa7d8274f7 cluster_major =
0 cluster_minor = 1
    Disk = UUID = 6c1b76e1-3e0a-ff3c-3c43-cb6c3881c3bf cluster_major = 0
cluster_minor = 2
    Disk = hdisk3 UUID = 20d93b0c-97e8-85ee-8b71-b880ccf848b7 cluster_major =
0 cluster_minor = 3
    Disk = UUID = 5890b139-e987-1451-211e-24ba89e7d1df cluster_major = 0
cluster_minor = 4
Multicast for site primary_site1: IPv4 228.168.100.20 IPv6 ff05::e4a8:6414
Multicast for site standby_site2: IPv4 228.168.100.21 IPv6 ff05::e4a8:6415
Communication Mode: unicast

```

```
Local node maximum capabilities: CAA_NETMON, AUTO_REPO_REPLACE, HNAME_CHG,  
UNICAST, IPV6, SITE  
Effective cluster-wide capabilities: CAA_NETMON, AUTO_REPO_REPLACE, HNAME_CHG,  
UNICAST, IPV6, SITE  
#
```

Example 4-37 shows the output from node 2.

Example 4-37 Output of the lscluster command (node 2)

```
# lscluster -c  
Cluster Name: primo_s1_n1_cluster  
Cluster UUID: d34e8658-8894-11e5-8002-6e8ddb7b3702  
Number of nodes in cluster = 2  
    Cluster ID for node primo_s1_n1: 1  
    Primary IP address for node primo_s1_n1: 192.168.100.20  
    Cluster ID for node primo_s2_n1: 2  
    Primary IP address for node primo_s2_n1: 192.168.100.21  
Number of disks in cluster = 4  
    Disk = UUID = 2f1b2492-46ca-eb3b-faf9-87fa7d8274f7 cluster_major = 0  
cluster_minor = 1  
    Disk = UUID = 20d93b0c-97e8-85ee-8b71-b880ccf848b7 cluster_major = 0  
cluster_minor = 3  
    Disk = hdisk2 UUID = 5890b139-e987-1451-211e-24ba89e7d1df cluster_major =  
0 cluster_minor = 4  
    Disk = hdisk1 UUID = 6c1b76e1-3e0a-ff3c-3c43-cb6c3881c3bf cluster_major =  
0 cluster_minor = 2  
Multicast for site standby_site2: IPv4 228.168.100.21 IPv6 ff05::e4a8:6415  
Multicast for site primary_site1: IPv4 228.168.100.20 IPv6 ff05::e4a8:6414  
Communication Mode: unicast  
Local node maximum capabilities: CAA_NETMON, AUTO_REPO_REPLACE, HNAME_CHG,  
UNICAST, IPV6, SITE  
Effective cluster-wide capabilities: CAA_NETMON, AUTO_REPO_REPLACE, HNAME_CHG,  
UNICAST, IPV6, SITE  
#
```

4.5 PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX

Starting with PowerHA V7.1, instead of the RSCT Topology Service, the CAA component is used in a PowerHA V7 setup. Figure 4-11 shows the connections between PowerHA V7, RSCT, and CAA (mainly the connection from PowerHA to RSCT Group services, and from there to CAA and back, are used). The potential communication to RMC is rarely used.

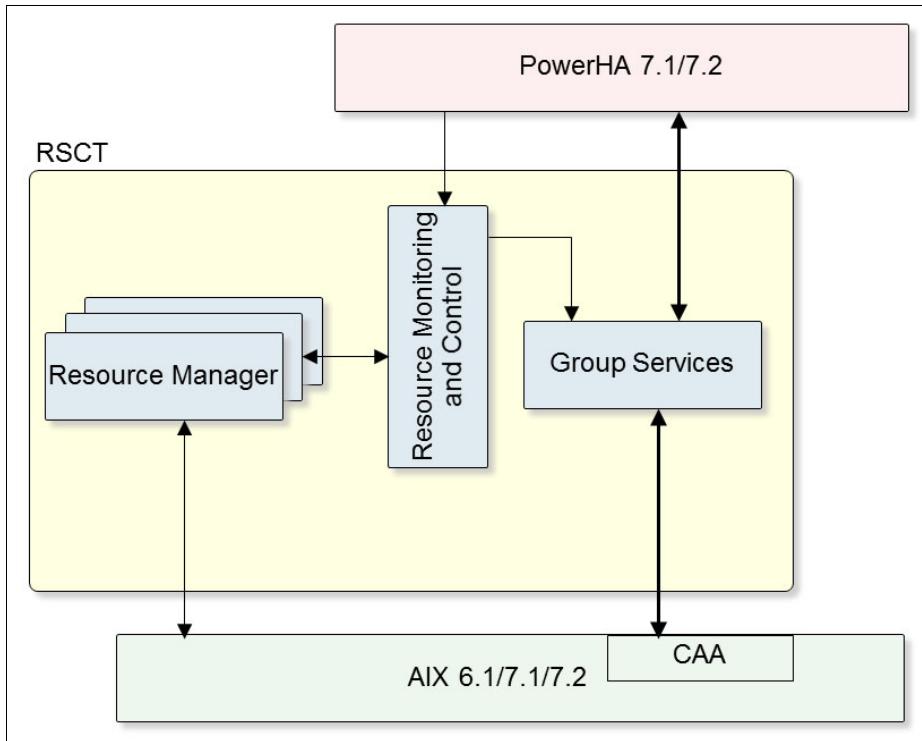


Figure 4-11 PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX overview

4.5.1 Configuring PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX

There is no need to configure RSCT or CAA. You just need to configure or migrate PowerHA. To set it up, use the `smitty sysmirror` panels or the `clmgr` command, as shown in Figure 4-12. The different migration processes operate in a similar way.

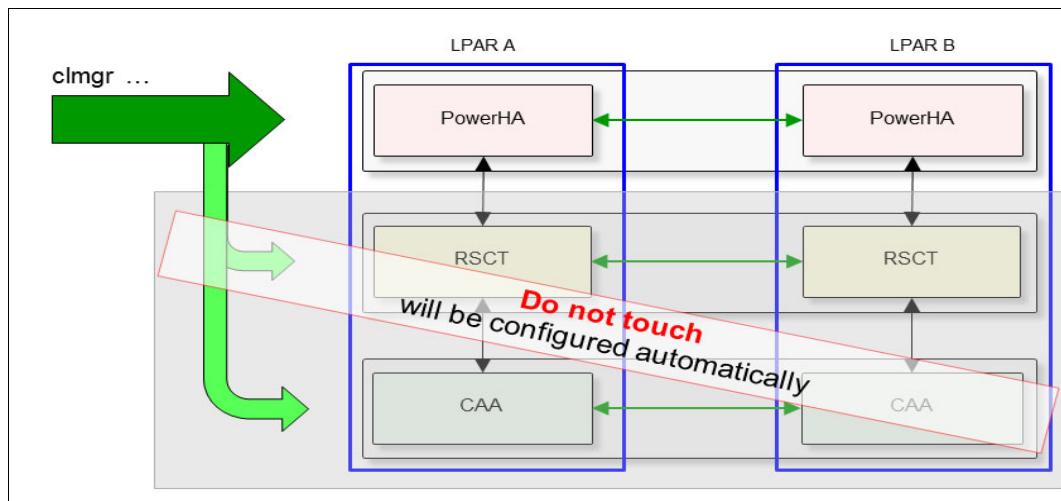


Figure 4-12 Set up PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX

4.5.2 Relationship between PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX

This section describes, from a high-level point of view, the relationship between PowerHA, RSCT, and CAA. The intention of this section is to give you a general understanding of what is running in the background. The examples in this section are based on a 2-node cluster.

In traditional situations, there is no need to use CAA or RSCT commands because they are all managed by PowerHA.

All PowerHA components are up

In a cluster where the state of PowerHA is up on all nodes, you also have all of the RSCT and CAA services running, as shown in Figure 4-13.

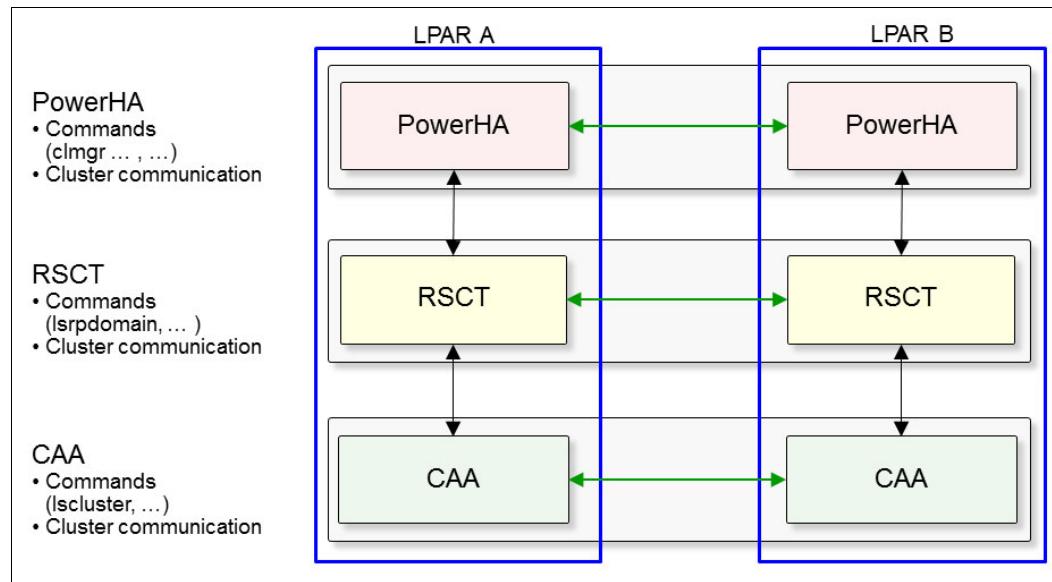


Figure 4-13 All cluster services are up

To check whether the services are up, you can use different commands. In the following examples, we use the **c1mgr**, **c1RGinfo**, **1srpdomain**, and **1scluster** commands. Example 4-38 shows the output of the **c1mgr** and **c1RGinfo** PowerHA commands.

Example 4-38 Checking PowerHA when all is up

```
# c1mgr -a state query cluster
STATE="STABLE"
# c1RGinfo
-----
Group Name          Group State    Node
-----
Test_RG            ONLINE        CL1_N1
                  OFFLINE       CL1_N2
#
```

To check whether RSCT is running, use the **1srpdomain** command. Example 4-39 shows the output of the command.

Example 4-39 Checking for RSCT when all components are running

```
# 1srpdomain
Name          OpState RSCTActiveVersion MixedVersions TSPort GSPort
CL1_N1_cluster  Online  3.1.5.0      Yes           12347  12348
#
```

To check whether CAA is running correctly, we use the `lsccluster` command. You must specify an option when using the `lsccluster` command. We use the option `-m` in Example 4-40. In most cases, any other valid option can be used as well. However, to be sure, use the `-m` option.

In most cases, the general behavior is that when you get a valid output, CAA is running. Otherwise, you get an error message informing you that the cluster services are not active.

Example 4-40 Checking for CAA when all is up

```
# lsccluster -m | egrep "Node name|State of node"
  Node name: powerha-c2n1
  State of node: UP
  Node name: powerha-c2n2
  State of node: UP NODE_LOCAL
#
```

One node that is stopped with Unmanage

In a cluster where one node is stopped with an Unmanage state, all of the underlying components (RSCT and CAA) must stay running. Figure 4-14 illustrates what happens when LPAR A is stopped with an Unmanage state.

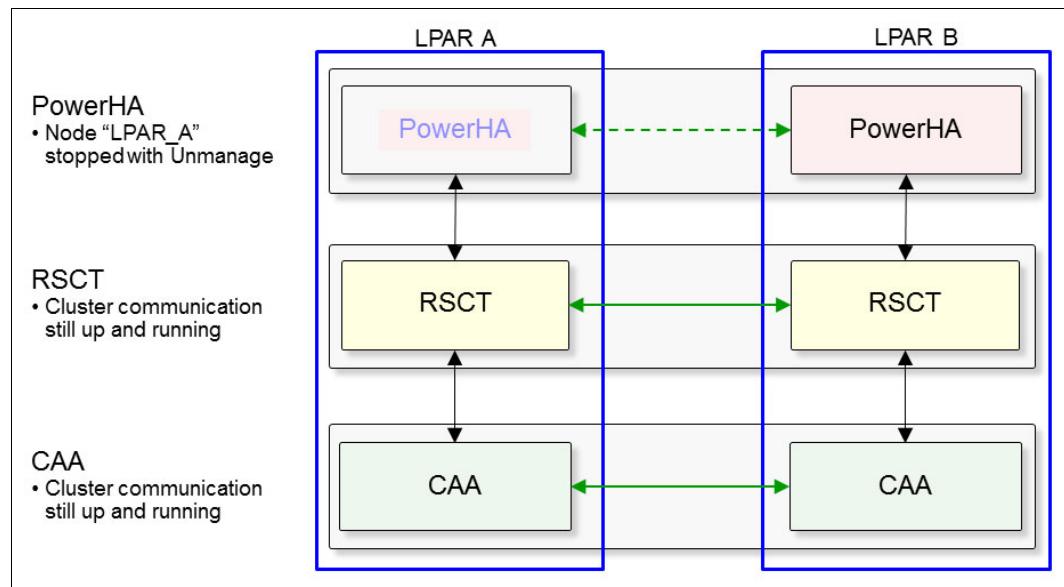


Figure 4-14 One node where all RGs are unmanaged

The following examples use the same commands as in “All PowerHA components are up” on page 105 to check the status of the different components. Example 4-41 shows the output of the `clmgr` and `c1RGinfo` PowerHA commands.

Example 4-41 Checking PowerHA: One node in state Unmanaged

```
# clmgr -a state query cluster
STATE="WARNING"
# c1RGinfo
-----
Group Name          Group State    Node
-----
Test_RG            UNMANAGED     CL1_N1
```

UNMANAGED	CL1_N2
#	

As expected, the output of the **1srpdomain** RSCT command shows that RSCT is still online (see Example 4-42).

Example 4-42 Checking RSCT: One node in state unmAnaged

Name	OpState	RSCTActiveVersion	MixedVersions	TSPort	GSPort
CL1_N1_cluster	Online	3.1.5.0	Yes	12347	12348
#					

Also, as expected, checking for CAA shows that it is running, as shown in Example 4-43.

Example 4-43 Checking CAA: One node in state Unmanaged

Node name:	powerha-c2n1
State of node:	UP
Node name:	powerha-c2n2
State of node:	UP NODE_LOCAL
#	

PowerHA stopped on all nodes

When you stop PowerHA on all cluster nodes, then you get the situation that is shown in Figure 4-15. In this case, PowerHA is stopped on all cluster nodes but RSCT and CAA are still running. You have the same situation after a system restart of all your cluster nodes (assuming that you do not use the automatic startup of PowerHA).

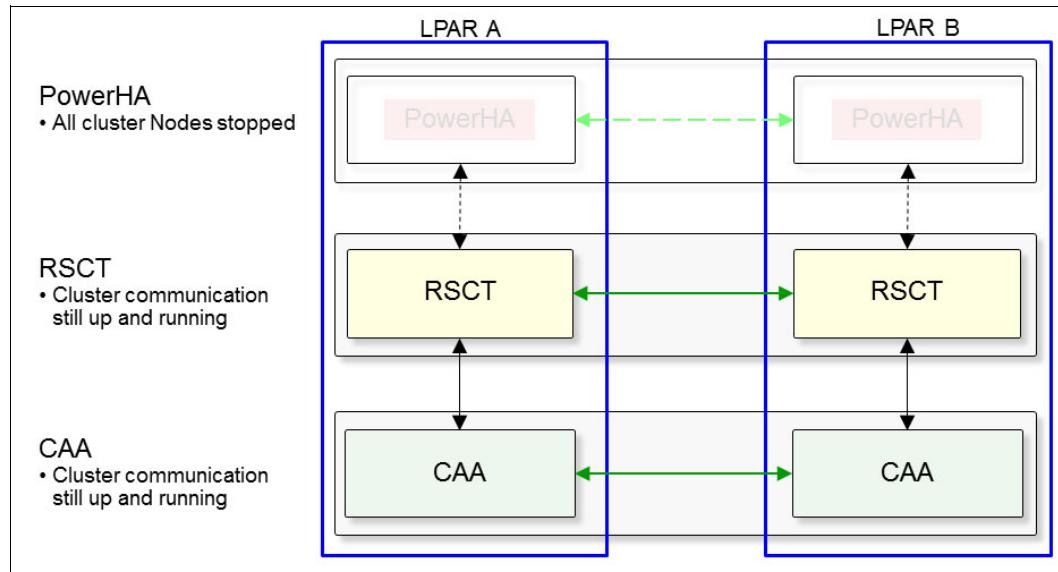


Figure 4-15 PowerHA stopped on all cluster nodes

Again, we use the commands used in “All PowerHA components are up” on page 105 to check the status of the different components. Example 4-44 shows the output of the PowerHA commands **c1mgr** and **c1RGinfo**.

As expected, the **c1mgr** command shows that PowerHA is offline, and **c1RGinfo** returns an error message.

Example 4-44 Checking PowerHA: PowerHA stopped on all cluster nodes

```
# c1mgr -a state query cluster
STATE="OFFLINE"
# c1RGinfo
Cluster IPC error: The cluster manager on node CL1_N1 is in ST_INIT or
NOT_CONFIGURED state and cannot process the IPC request.
#
```

The output of the RSCT **1srpdomain** command shows that RSCT is still online (Example 4-45).

Example 4-45 Checking RSCT: PowerHA stopped on all cluster nodes

```
# 1srpdomain
Name          OpState RSCTActiveVersion MixedVersions TSPort GSPort
CL1_N1_cluster    Online   3.1.5.0        Yes      12347  12348
#
```

The check for CAA shows that it is running, as shown in Example 4-46.

When RSCT is running, CAA must be up as well. This statement is only true for a PowerHA cluster.

Example 4-46 Checking CAA: PowerHA stopped on all cluster nodes

```
# lscluster -m | egrep "Node name|State of node"
Node name: powerha-c2n1
State of node: UP
Node name: powerha-c2n2
State of node: UP NODE_LOCAL
#
```

All cluster components are stopped

By default, CAA and RSCT are automatically started as part of an operating system restart (if the system is configured by PowerHA).

There are situations when you need to stop all three cluster components, for example, when you must change the RSCT or CAA software, as shown in Figure 4-16 on page 109.

For example, to stop all cluster components, use `c1mgr off cluster STOP_CAA=yes`. For more information about starting and stopping CAA, see 4.5.3, “How to start and stop CAA and RSCT” on page 110.

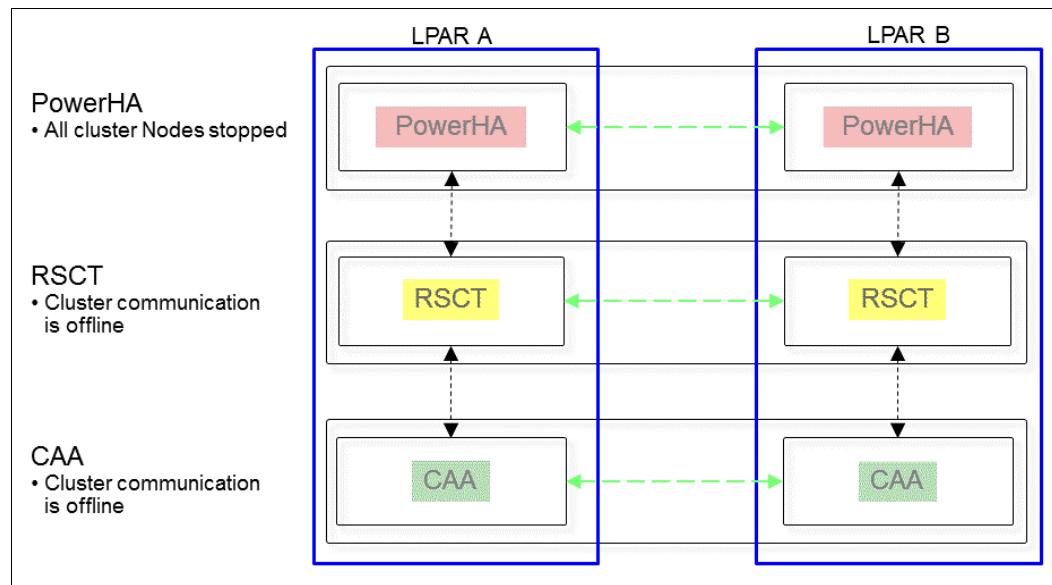


Figure 4-16 All cluster services stopped

Example 4-47 shows the status of the cluster with all services stopped. As in the previous examples, we use the `c1mgr` and `c1RGinfo` commands.

Example 4-47 Checking PowerHA: All cluster services stopped

```
# c1mgr -a state query cluster
STATE="OFFLINE"
root@CL1_N1:/home/root# c1RGinfo
Cluster IPC error: The cluster manager on node CL1_N1 is in ST_INIT or
NOT_CONFIGURED state and cannot process the IPC request.
#
```

The `1srpdomain` command shows that the RSCT cluster is offline, as shown in Example 4-48.

Example 4-48 Checking RSCT: All cluster services stopped

```
# 1srpdomain
Name          OpState RSCTActiveVersion MixedVersions TSPort GSPort
CL1_N1_cluster Offline 3.1.5.0      Yes           12347  12348
#
```

The output of the `1scluster` command creates an error message in this case, as shown in Example 4-49.

Example 4-49 Check CAA: All cluster services stopped

```
# 1scluster -m
1scluster: Cluster services are not active on this node because it has been
stopped.
#
```

4.5.3 How to start and stop CAA and RSCT

CAA and RSCT are stopped and started together. CAA and RSCT are automatically started as part of an operating system start (if it is configured by PowerHA).

If you want to stop CAA and RSCT, you must use the **clmgr** command (at the time of writing, SMIT does not support this operation). To stop it, you must use the **STOP_CAA=yes** argument. This argument can be used for both CAA and RSCT, and the complete cluster or a set of nodes.

The information when you stopped CAA manually is preserved across an operating system restart. So, if you want to start PowerHA on a node where CAA and RSCT were stopped deliberately, you must use the **START_CAA** argument.

To start CAA and RSCT, you can use the **clmgr** command with the argument **START_CAA=yes**. This command also starts PowerHA.

Example 4-50 shows how to stop or start CAA and RSCT. All of these examples stop all three components or start all three components.

Example 4-50 Using clmgr to start and stop CAA and RSCT

To Stop CAA and RSCT:

- **clmgr off cluster STOP_CAA=yes**
- **clmgr off node system-a STOP_CAA=yes**

To Start CAA and RSCT:

- **clmgr on cluster START_CAA=yes**
 - **clmgr on node system-a START_CAA=yes**
-

Starting with AIX 7.1 TL4 or AIX 7.2, you can use the **clctrl** command to stop or start CAA and RSCT. To stop it, use the **-stop** option for the **clctrl** command. This also stops PowerHA. To start CAA and RSCT, you can use the **-start** option. If **-start** is used, only CAA and RSCT start. To start PowerHA, you must use the **clmgr** command, or use SMIT afterward.



Migration

This chapter covers the migration options from PowerHA V7.1.3 to PowerHA V7.2.

This chapter covers the following topics:

- ▶ Migration planning:
 - PowerHA SystemMirror V7.2.1 requirements
 - Deprecated features
 - Migration options
 - Migration steps
 - Migration matrix to PowerHA SystemMirror 7.2.1
- ▶ Migration scenarios from PowerHA V7.1.3:
 - PowerHA V7.1.3 test environment overview
 - Rolling migration from PowerHA V7.1.3
 - Offline migration from PowerHA V7.1.3
 - Snapshot migration from PowerHA V7.1.3
 - Nondisruptive upgrade from PowerHA V7.1.3
- ▶ Migration scenarios from PowerHA V7.2.0:
 - PowerHA V7.2.0 test environment overview
 - Offline migration from PowerHA V7.2.0
 - Rolling migration from PowerHA V7.2.0
 - Snapshot migration from PowerHA V7.2.0
 - Nondisruptive upgrade from PowerHA V7.2.0

5.1 Migration planning

Proper planning of the migration procedure of clusters to IBM PowerHA SystemMirror V7.2.1 is important to minimize the risk duration of the process itself. The following set of actions must be considered when planning the migration of existing PowerHA clusters.

Before beginning the migration procedure, always have a contingency plan in case any problems occur. Here are some general suggestions:

- ▶ Create a backup of rootvg.

In some cases of upgrading PowerHA, depending on the starting point, updating or upgrading the AIX base operating system is also required. Therefore, a preferred practice is to save your existing rootvg. One method is to create a clone by using `alt_disk_copy` on other available disks on the system. That way, a simple change to the bootlist and a restart can easily return the system to the beginning 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; make a copy of it and also save a copy from the cluster nodes for additional safety.

`/usr/es/sbin/cluster/snapshots`

- ▶ Save any user-provided scripts.

This most commonly refers to custom events, pre- and post-events, application controller, and application monitoring scripts.

- ▶ Save common configuration files needs for proper functioning, such as:

`/etc/hosts`

`/etc/cluster/rhosts`

`/usr/es/sbin/cluster/netmon.cf`

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 V7.2.1 requirements

Here are listed the software and hardware requirements that must be met before migrating to PowerHA SystemMirror V7.2.1.

Software requirements

The software requirements are as follows:

- ▶ IBM AIX 7.1 with Technology Level 3 with Service Pack 5, or later
- ▶ IBM AIX 7.1 with Technology Level 4 with Service Pack 2, or later
- ▶ IBM AIX 7.2 with Service Pack 1, or later
- ▶ IBM AIX 7.2 with Technology Level 1, or later

Hardware

Support is available only for POWER5 technologies and later.

5.1.2 Deprecated features

Starting with PowerHA V7.2.0, the IBM Systems Director plug-in is no longer supported or available. However, PowerHA V7.2.1 does provide a new GUI often referred to as the System Mirror User Interface (SMUI). More information about this feature can be found in Chapter 9, “IBM PowerHA SystemMirror User Interface” on page 303.

5.1.3 Migration options

There are four methods of performing a migration of a PowerHA cluster. Each of them is briefly described, and in more detail for the corresponding migration scenarios that are included in this chapter.

Offline	A migration method where PowerHA is brought offline on all nodes before performing the software upgrade. During this time, the cluster resource groups (RGs) are not available.
Rolling	A migration method from one PowerHA version to another during which cluster services are stopped one node at a time. That node is upgraded and reintegrated into the cluster before the next node is upgraded. It requires little downtime, mostly for moving the RGs between nodes to allow each node to be upgraded.
Snapshot	A migration method from one PowerHA version to another, during which you take a snapshot of the current cluster configuration, stop cluster services on all nodes, uninstall the current version of PowerHA and then install the preferred version of PowerHA SystemMirror, convert the snapshot by running the <code>c1convert_snapshot</code> utility, and restore the cluster configuration from the converted snapshot.
Nondisruptive	This method is by far the most preferred method of migration whenever possible. As its name implies, the cluster RGs remain available and the applications functional during the cluster migration. All cluster nodes are sequentially (one node at a time) set to an <i>unmanaged</i> state, allowing all RGs on that node to remain operational while cluster services are stopped. However, this method can generally be used only when applying service packs to the cluster, and not doing major upgrades. This option does <i>not</i> apply when the upgrade of the base operating system is also required, such as when migrating PowerHA to a version newer than 7.1.x from an older version.

Important: Nodes in a cluster running two separate versions of PowerHA is considered to be in a *mixed cluster state*. A cluster in this state does not support any configuration changes or synchronization until all the nodes are migrated. Be sure to complete either the rolling or nondisruptive migration as soon as possible to ensure stable cluster functions.

5.1.4 Migration steps

The following sections give an overview of the steps that are required to perform each type of migration. Detailed examples of each migration type can be found in 5.2.5, “Nondisruptive upgrade from PowerHA V7.1.3” on page 124.

Offline method

Some of these steps can be performed in parallel because the entire cluster is offline.

Important: Always start with the latest service packs that are available for PowerHA, AIX, and Virtual I/O Server (VIOS).

Complete the following steps:

1. Stop cluster services on all nodes and bring the RGs offline.
2. Upgrade AIX (as needed):
 - a. Ensure that the prerequisites are installed, such as bos.cluster.
 - b. Restart.
3. Upgrade PowerHA. This step can be performed on both nodes in parallel.
4. Review the /tmp/c1convert.log file.
5. Restart the cluster services.

Rolling method

A rolling migration provides the least amount of downtime by upgrading one node at a time.

Important: Always start with the latest service packs that are available for PowerHA, AIX, and VIOS.

Complete the following steps:

1. Stop cluster services on one node (move the RGs as needed).
2. Upgrade AIX (as needed) and restart.
3. Upgrade PowerHA.
4. Review the /tmp/c1convert.log file.
5. Restart the cluster services.
6. Repeat these steps for each node.

Snapshot method

Some of these steps can often be performed in parallel because the entire cluster is offline.

Additional specifics when migrating from PowerHA 6.1, including crucial interim fixes, can be found at [PowerHA SystemMirror interim fix Bundles information](#).

Important: Always start with the latest service packs that are available for PowerHA, AIX, and VIOS.

Complete the following steps:

1. Stop cluster services on all nodes and bring the RGs offline.
2. Create a cluster snapshot. Save copies of it off the cluster.
3. Upgrade AIX (as needed) and restart.
4. Upgrade PowerHA. This step can be performed on both nodes in parallel.

5. Review the /tmp/clconvert.log file.
6. Restart the cluster services.

Nondisruptive upgrade

This method applies only when the AIX level is already at appropriate levels to support PowerHA V7.2.1 or later. Complete the following steps on *one* node:

1. Stop cluster services by unmanaging the RGs.
2. Upgrade PowerHA (**update_a11**).
3. Start cluster services with an automatic manage of the RGs.

Important: When restarting cluster services with the Automatic option for managing RGs, the application start scripts are invoked. Make sure that the application scripts can detect that the application is already running, or copy them and put a dummy blank executable script in their place and then copy them back after start.

5.1.5 Migration matrix to PowerHA SystemMirror 7.2.1

Table 5-1 shows the migration options between versions of PowerHA.

Important: Migrating from PowerHA V6.1 to V7.2.1 is *not* supported. You must upgrade to either V7.1.x or V7.2.0 first.

Table 5-1 Migration matrix table

PowerHA ^a	To V7.1.1	To V7.1.2	To V7.1.3	To V7.2.0	V7.2.1
From V6.1	Update to SP17 then R ^b , S, O are all viable options			R ^b , S, O	N/A
From V7.1.0	R ^b , S, O	R ^b , S, O	R ^b , S, O	R ^b , S, O	N/A
From V7.1.1		R, S, O, N ^b	R, S, O, N ^b	R, S, O, N ^b	N/A
From V7.1.2			R, S, O, N ^b	R, S, O, N ^b	N/A
From V7.1.3				R, S, O, N ^b	R, S, O, N ^b
From V7.2.0					R, S, O, N ^b

a. R: Rolling, S: Snapshot, O: Offline, and N: Nondisruptive.

b. This option is available only if the beginning AIX level is high enough to support the newer version.

5.2 Migration scenarios from PowerHA V7.1.3

This section further details the test scenarios that are used in each of these migration methods:

- ▶ Rolling migration
- ▶ Snapshot migration
- ▶ Offline migration
- ▶ Nondisruptive upgrade

5.2.1 PowerHA V7.1.3 test environment overview

For the following scenarios, we use a two-node cluster with nodes *Jess* and *Cass*. It consists of a single RG that is configured in a typical hot-standby configuration. Our test configuration consists of the following hardware and software (see Figure 5-1):

- ▶ POWER8 S814 with firmware 850
- ▶ HMC 850
- ▶ AIX 7.2.0 SP2
- ▶ PowerHA V7.1.3 SP5
- ▶ Storwize V7000 V7.6.1.1

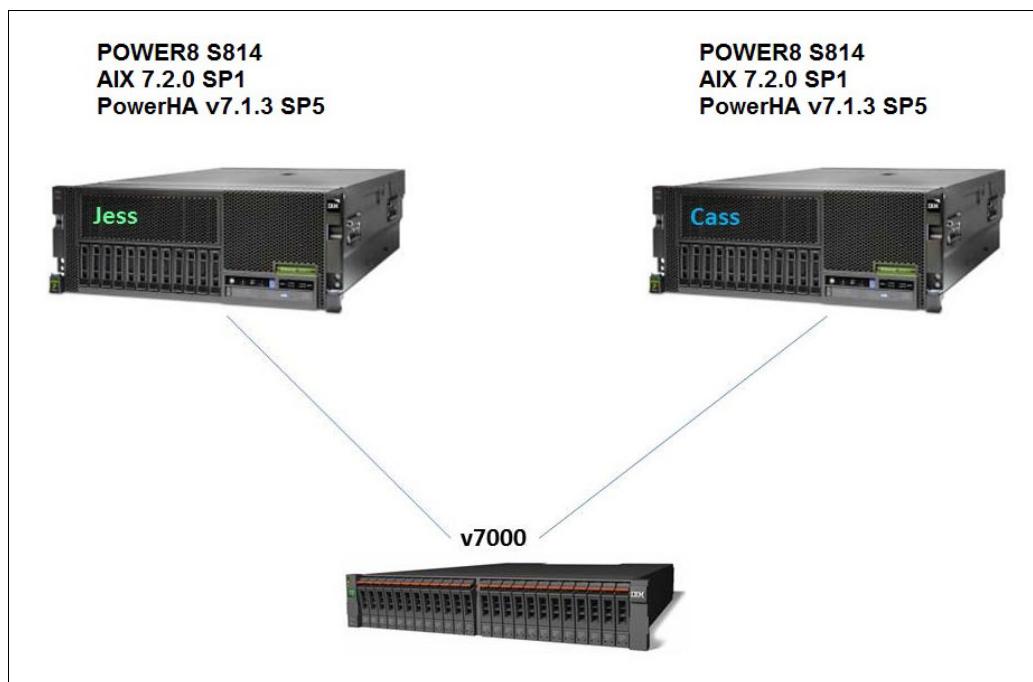


Figure 5-1 PowerHA V7.1.3 test migration cluster

5.2.2 Rolling migration from PowerHA V7.1.3

Here are the steps for a rolling migration from PowerHA V7.1.3.

Checking and documenting the initial stage

This step is described in “Checking and documenting the initial stage” on page 116 as being common to all migration scenarios and methods.

For the rolling migration, we begin with the standby node, *Cass*. Complete the following steps.

Tip: A demonstration of performing a rolling migration from PowerHA V7.1.3 to PowerHA V7.2.1 is this [YouTube video](#).

1. Stop cluster services on node Cass.

Run **smitty clstop** and choose the options that are shown in Figure 5-2. The OK response appears quickly. Make sure that the cluster node is in the ST_INIT state by reviewing the **lssrc -ls clstrmgrES|grep state** output.

Alternatively, you can accomplish this task by using the **clmgr** command:

```
clmgr stop node=Cass
```

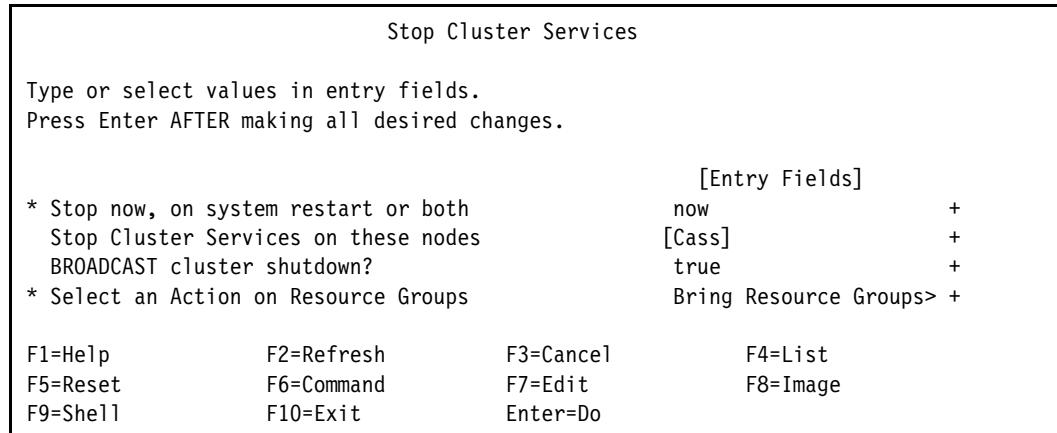


Figure 5-2 Stopping the cluster services

2. Upgrade AIX.

In our scenario, we have supported AIX levels for PowerHA V7.2.1 and do not need to perform this step. But if you do, a restart is required before continuing.

3. Verify that the clcomd daemon is active, as shown in Figure 5-3.

```
[root@Cass] /# lssrc -s clcomd
Subsystem      Group          PID      Status
clcomd        caa           3421236    active
```

Figure 5-3 Verify that clcomd is active

4. Upgrade PowerHA on node Cass. To upgrade PowerHA, run `e smitty update_all`, as shown in Figure 5-4. or run the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```

Update Installed Software to Latest Level (Update All)

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

<p>[TOP]</p> <p>* INPUT device / directory for software * SOFTWARE to update PREVIEW only? (update operation will NOT occur) COMMIT software updates? SAVE replaced files? AUTOMATICALLY install requisite software? EXTEND file systems if space needed? VERIFY install and check file sizes? DETAILED output? Process multiple volumes? ACCEPT new license agreements? Preview new LICENSE agreements?</p>	<p>[Entry Fields]</p> <p>. _update_all no + yes + no +</p>
<p>[MORE...6]</p>	
<p>F1=Help F2=Refresh F3=Cancel F4=List F5=Reset F6=Command F7>Edit F8=Image F9=Shell F10=Exit Enter=Do</p>	

Figure 5-4 Smitty update_all

Important: Set ACCEPT new license agreements? to yes.

5. Ensure that the file `/usr/es/sbin/cluster/netmon.cf` exists and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.
6. Start cluster services on node Cass by running `smitty clstart` or `clmgr start node=Cass`.

A message displays about cluster verification being skipped because of mixed versions, as shown in Figure 5-5 on page 119.

Important: While the cluster is a mixed cluster state, do *not* make any cluster changes or attempt to synchronize the cluster.

After starting, validate that the cluster is stable before continuing by running the following command:

```
lssrc -ls clstrmgrES |grep -i state.
```

```

Cluster services are running at different levels across
the cluster. Verification will not be invoked in this environment.

Starting Cluster Services on node: Cass
This may take a few minutes. Please wait...
Cass: Nov 8 2016 10:18:43 Starting execution of
/usr/es/sbin/cluster/etc/rc.c
luster
Cass: with parameters: -boot -N -A -C interactive -P cl_rc_cluster

```

Figure 5-5 Verification skipped

7. Repeat the previous steps for node *Jess*. However, when stopping cluster services, choose the Move Resource Groups option, as shown in Figure 5-6.

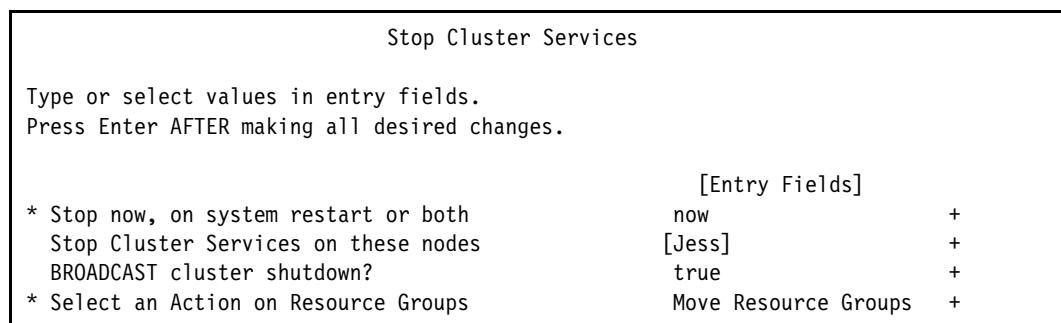


Figure 5-6 Run clstop and move the resource group

8. Upgrade AIX (if needed).

Important: If upgrading to AIX 7.2.0, see the [AIX 7.2 Release Notes](#) regarding RSCT filesets when upgrading.

In our scenario, we have supported AIX levels for PowerHA V7.2 and do not need to perform this step. But if you do, a restart is required before continuing.

9. Verify that the *clcomd* daemon is active, as shown in Figure 5-7.

[root@Jess] /# lssrc -s clcomd			
Subsystem	Group	PID	Status
clcomd	caa	50467008	active

Figure 5-7 Verify that clcomd is active

10. Upgrade PowerHA on node *Jess*. To upgrade PowerHA, run **smitty update_all**, as shown in Figure 5-4 on page 118, or run the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```

11. Ensure that the file */usr/es/sbin/cluster/netmon.cf* exists and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.

12. Start cluster services on node Jess by running one of the following commands:

- **smitty clstart**
- **clmgr start node=Jess**

13. Verify that the cluster completed the migration on both nodes by checking that version number is 17, as shown in Example 5-1.

Example 5-1 Verifying that the migration completed on both nodes

```
# clcmd odmget HACMPcluster |grep version
    cluster_version = 17
    cluster_version = 17

#clcmd odmget HACMPnode |grep version |sort -u
    version = 17
```

Important: Both nodes must show version=17; otherwise, the migration did not complete successfully. Call IBM Support.

Although the migration is complete, the resource is running on node Cass. If you want, move the RG back to node Jess, as shown in Example 5-2.

Example 5-2 Move the resource group back to node Jess

```
# clmgr move rg demorg node=Jess
Attempting to move resource group demorg to node Cass.
```

Waiting for the cluster to process the resource group movement request....

Waiting for the cluster to stabilize....

Resource group movement successful.

Resource group demorg is online on node Cass.

Cluster Name: Jess_cluster

Resource Group Name: demorg	
Node	Group State
Jess	ONLINE
Cass	OFFLINE

Important: Always test the cluster thoroughly after migration.

5.2.3 Offline migration from PowerHA V7.1.3

For an offline migration, you can perform many of the steps in parallel on all (both) nodes in the cluster. However, to accomplish this task, you must plan full cluster outage.

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Complete the following steps:

1. Stop cluster services on both nodes Jess and Cass by running **smitty clstop** and choosing the options that are shown in Figure 5-8. The OK response appears quickly.

As an alternative, you can also stop the entire cluster by running the following command:

```
clmgr stop cluster
```

Make sure that the cluster node is in the ST_INIT state by reviewing the **clcmd lssrc -ls clstrmgrES|grep state** output.

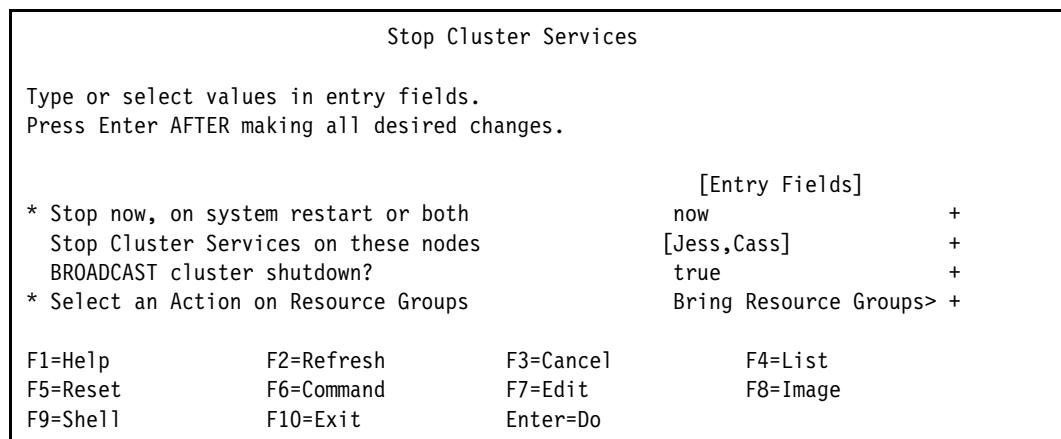


Figure 5-8 Stopping the cluster services

2. Upgrade AIX on both nodes.

Important: If upgrading to AIX 7.2.0, see the [AIX 7.2 Release Notes](#) regarding RSCT filesets when upgrading.

In our scenario, we have supported AIX levels for PowerHA v7.2.1 and do not need to perform this step. But if you do, a restart is required before continuing.

3. Verify that the clcmd daemon is active on both nodes, as shown in Figure 5-9.

```
# clcmd lssrc -s clcmd  
  
-----  
NODE Jess  
-----  
Subsystem      Group        PID      Status  
  clcmd        caa         20775182  active  
  
-----  
NODE Cass  
-----  
Subsystem      Group        PID      Status  
  clcmd        caa         5177840   active
```

Figure 5-9 Verify that clcmd is active

4. Upgrade to PowerHA V7.2.1 by running `smitty update_all` on both nodes or by running the following command from within the directory in which the updates are:
`install_all_updates -vY -d .`
5. Verify that the version numbers show correctly, as shown in Example 5-1 on page 120.
6. Ensure that the file `/usr/es/sbin/cluster/netmon.cf` exists on all nodes and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.
7. Restart cluster on both nodes by running `clmgr start cluster`.

Important: Always test the cluster thoroughly after migrating.

5.2.4 Snapshot migration from PowerHA V7.1.3

For a snapshot migration, you can perform many of the steps in parallel on all (both) nodes in the cluster. However, this requires a full cluster outage.

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Complete the following steps:

1. Stop cluster services on both nodes Jess and Cass by running `smitty clstop` and choosing the options that are shown in Figure 5-10. The OK response appears quickly. As an alternative, you can also stop the entire cluster by running `clmgr stop cluster`. Make sure that the cluster node is in the ST_INIT state by reviewing the `clcmd lssrc -ls clstrmgrES|grep state` output.

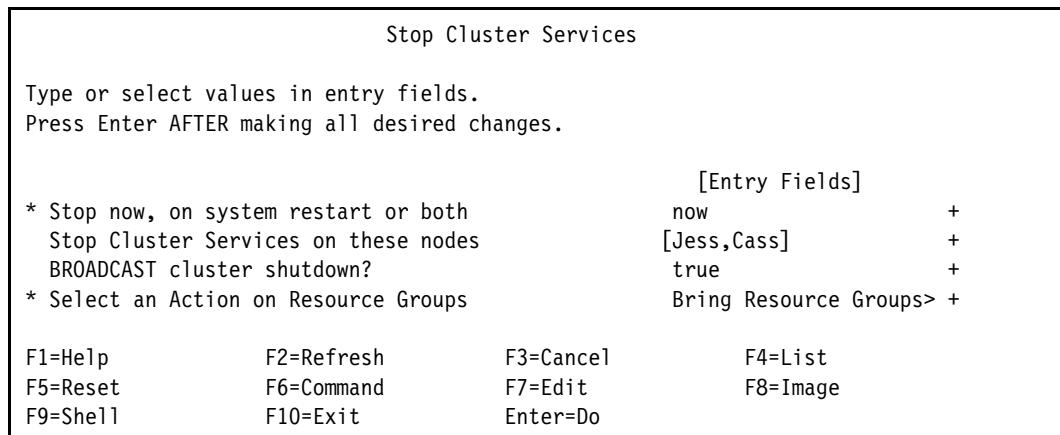


Figure 5-10 Stopping the cluster services

2. Create a cluster snapshot by running `smitty cm_add_snap.dialog` and completing the options, as shown in Figure 5-11.

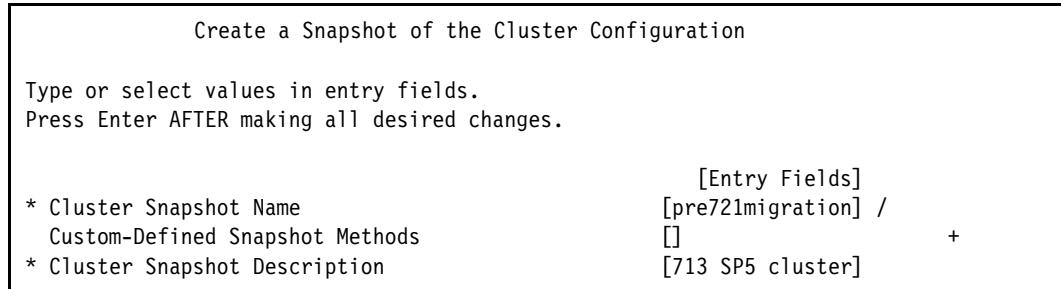


Figure 5-11 Creating a cluster snapshot

3. Upgrade AIX on both nodes.

Important: If upgrading to AIX 7.2.0, see the [AIX 7.2 Release Notes](#) regarding RSCT filesets when upgrading.

In our scenario, we have supported AIX levels for PowerHA V7.2 and do not need to perform this step. But if you do, a restart is required before continuing.

4. Verify that the clcomd daemon is active on both nodes, as shown in Figure 5-12.

```
# clcmd lssrc -s clcomd

-----
NODE Jess
-----
Subsystem      Group          PID      Status
  clcomd        caa           20775182  active

-----
NODE Cass
-----
Subsystem      Group          PID      Status
  clcomd        caa           5177840   active
```

Figure 5-12 Verify that clcomd is active

5. Next, uninstall PowerHA 6.1 on both nodes Jess and Cass by running `smitty remove` on `cluster.*`.
6. Install PowerHA V7.2.1 by running `smitty install_all` on both nodes.
7. Convert the previously created snapshot as follows:

```
/usr/es/sbin/cluster/conversion/clconvert_snapshot -v 7.1.3 -s pre721migration
Extracting ODM's from snapshot file... done.
Converting extracted ODM's... done.
Rebuilding snapshot file... done.
```

8. Restore the cluster configuration from the converted snapshot by running **smitty cm_apply_snap.select** and choosing the snapshot from the menu. The snapshot autofills the last menu, as shown in Figure 5-13.

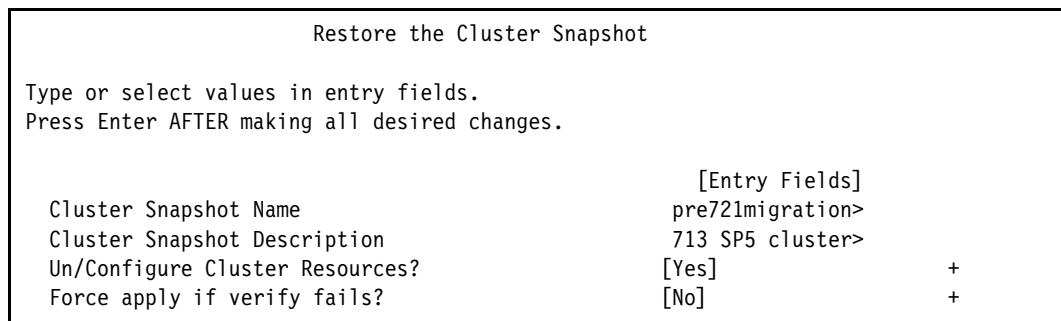


Figure 5-13 Restoring a cluster configuration from a snapshot

The restore process automatically re-creates and synchronizes the cluster.

9. Verify that the version numbers show correctly, as shown in Example 5-1 on page 120.
10. Ensure that the file `/usr/es/sbin/cluster/netmon.cf` exists on all nodes and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets might overwrite this file with an empty one.
11. Restart the cluster on both nodes by running **clmgr start cluster**.

Important: Always test the cluster thoroughly after migrating.

5.2.5 Nondisruptive upgrade from PowerHA V7.1.3

This method applies only when the AIX level is already at appropriate levels to support PowerHA V7.2.1 (or later).

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Complete the following steps:

1. Stop cluster services by performing an unmanage of the RGs on node Cass, as shown in Example 5-3.

Example 5-3 Stop a cluster node with the unmanage option

```
# clmgr stop node=Cass manage=unmanage
```

Warning: "WHEN" must be specified. Since it was not, a default of "now" will be used.

Broadcast message from root@Cass (tty) at 11:39:28 ...

PowerHA SystemMirror on Cass shutting down. Please exit any cluster applications...

Cass: 0513-044 The clevmgrdES Subsystem was requested to stop.

. "Cass" is now unmanaged.

```
Cass: Nov  8 2016 11:39:28/usr/es/sbin/cluster/utilities/clstop: called with  
flags -N -f
```

2. Upgrade PowerHA (**update_all**) by running the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```
3. Start cluster services by using an automatic manage of the RGs on Cass, as shown in Example 5-4.

Example 5-4 Start the cluster node with the automatic manage option

```
# clmgr start node=Cass  
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 Prior to Starting Cluster Services.

```
Cass: start_cluster: Starting PowerHA SystemMirror  
. "Cass" is now online.
```

```
Starting Cluster Services on node: Cass  
This may take a few minutes. Please wait...  
Cass: Nov  8 2016 11:43:40Starting execution of  
/usr/es/sbin/cluster/etc/rc.cluster  
Cass: with parameters: -boot -N -b -P cl_rc_cluster -A  
Cass:  
Cass: Nov  8 2016 11:43:40usage: cl_echo messageid (default) messageNov  8 2016  
11:43:40usage: cl_echo messageid (default) messageRETURN_CODE=0
```

Important: Restarting cluster services with the **Automatic** option for managing RGs invokes the application start scripts. Make sure that the application scripts can detect that the application is already running, or copy and put a dummy blank executable script in their place and then copy them back after start.

Repeat the steps on node Jess.

4. Stop cluster services by performing an unmanage of the RGs on node Jess, as shown in Example 5-5.

Example 5-5 Stop the cluster node with the unmanage option

```
# clmgr stop node=Jess manage=unmanage  
  
Warning: "WHEN" must be specified. Since it was not, a default of "now" will be  
used.  
Broadcast message from root@Jess (tty) at 11:52:48 ...  
  
PowerHA SystemMirror on Jess shutting down. Please exit any cluster applications...  
Jess: 0513-044 The clevmgrdES Subsystem was requested to stop.  
. "Jess" is now unmanaged.  
  
Jess: Nov  8 2016 11:52:48/usr/es/sbin/cluster/utilities/clstop: called with flags -N -f
```

5. Upgrade PowerHA (**update_a11**) by running the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```
6. Start cluster services by performing an automatic manage of the RGs on Jess, as shown in Example 5-6.

Example 5-6 Start a cluster node with the automatic manage option

```
# clmgr start node=Jess
```

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 Prior to Starting Cluster Services.

Jess: start_cluster: Starting PowerHA SystemMirror

...

"Jess" is now online.

Starting Cluster Services on node: Jessica

This may take a few minutes. Please wait...

Jess: Nov 8 2016 11:54:40Starting execution of

/usr/es/sbin/cluster/etc/rc.cluster

Jess: with parameters: -boot -N -b -P cl_rc_cluster -A

Jess:

Jess: Nov 8 2016 11:54:40usage: cl_echo messageid (default) messageNov 8 2016
11:54:40usage: cl_echo messageid (default) messageRETURN_CODE=0

Important: Restarting cluster services with the **Automatic** option for managing RGs invokes the application start scripts. Make sure that the application scripts can detect that the application is already running, or copy and put a dummy blank executable script in their place and then copy them back after start.

7. Verify that the version numbers show correctly, as shown in Example 5-1 on page 120.
8. Ensure that the file /usr/es/sbin/cluster/netmon.cf exists on all nodes and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.

5.3 Migration scenarios from PowerHA V7.2.0

This section further details test scenarios that are used in each of these migrations methods:

- ▶ Rolling migration
- ▶ Snapshot migration
- ▶ Offline migration
- ▶ Nondisruptive upgrade

5.3.1 PowerHA V7.2.0 test environment overview

For the following scenarios, we use a two-node cluster with nodes *Jess* and *Cass*. It consists of a single RG that is configured in a typical hot-standby, as shown in Figure 5-14:

- ▶ POWER8 S814 with firmware 850
- ▶ HMC 850
- ▶ AIX 7.2.0 SP1
- ▶ PowerHA V7.2.0 SP2
- ▶ Storwize V7000 V7.6.1.1

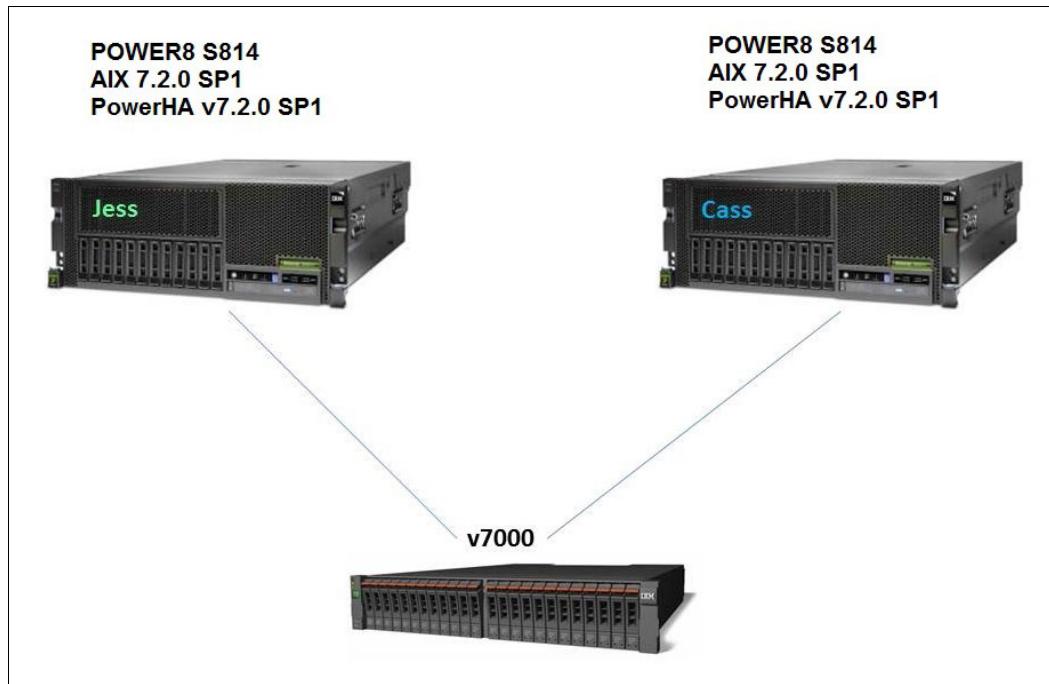


Figure 5-14 PowerHA V7.2.0 test migration cluster

5.3.2 Rolling migration from PowerHA V7.2.0

For the rolling migration, begin with the standby node *Cass*.

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Although the version level is different, the steps are identical as though starting from Version 7.2.0.

Complete the following steps:

1. Stop the cluster services on node Cass by running **smitty clstop** and choosing the options that are shown in Figure 5-15. The OK response appears quickly. Make sure that the cluster node is in the ST_INIT state by reviewing the **lssrc -ls clstrmgrES|grep state** output, as shown in Example 5-7.

Example 5-7 Cluster node state

```
# lssrc -ls clstrmgrES|grep state
Current state: ST_INIT
```

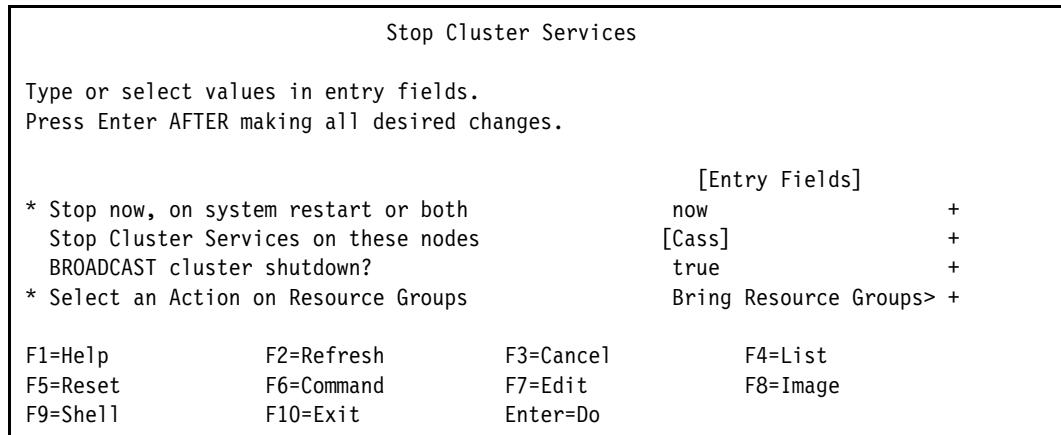


Figure 5-15 Stopping the cluster services

You can also stop cluster services by using the **clmgr** command:

```
clmgr stop node=Cass
```

2. Upgrade AIX.

In our scenario, we have supported AIX levels for PowerHA V7.2.1 and do not need to perform this step. But if you do, a restart is required before continuing.

3. Verify that the clcomd daemon is active, as shown in Figure 5-16.

```
[root@Cass] /# lssrc -s clcomd
Subsystem          Group           PID      Status
clcomd            caa            3421236  active
```

Figure 5-16 Verify that clcomd is active

4. Upgrade PowerHA on node Cass. To upgrade PowerHA, run **smitty update_all**, as shown in Figure 5-4 on page 118, or run the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```

Example 5-8 Install_all_updates

```
# install_all_updates -vY -d .
install_all_updates: Initializing system parameters.
install_all_updates: Log file is /var/adm/ras/install_all_updates.log
install_all_updates: Checking for updated install utilities on media.
install_all_updates: Processing media.
install_all_updates: Generating list of updatable installp filesets.
```

*** ATTENTION: the following list of filesets are installable base images that are updates to currently installed filesets. Because these filesets are base-level images, they will be committed automatically. After these filesets are installed, they can be down-leveled by performing a force-overwrite with the previous base-level. See the `installp` man page for more details. ***

```
cluster.es.client.clcomd 7.2.1.0
cluster.es.client.lib 7.2.1.0
cluster.es.client.rte 7.2.1.0
cluster.es.client.utils 7.2.1.0
cluster.es.cspoc.cmds 7.2.1.0
cluster.es.cspoc.rte 7.2.1.0
cluster.es.migcheck 7.2.1.0
cluster.es.server.diag 7.2.1.0
cluster.es.server.events 7.2.1.0
cluster.es.server.rte 7.2.1.0
cluster.es.server.testtool 7.2.1.0
cluster.es.server.utils 7.2.1.0
cluster.license 7.2.1.0
```

<< End of Fileset List >>

`install_all_updates`: The following filesets have been selected as updates to currently installed software:

```
cluster.es.client.clcomd 7.2.1.0
cluster.es.client.lib 7.2.1.0
cluster.es.client.rte 7.2.1.0
cluster.es.client.utils 7.2.1.0
cluster.es.cspoc.cmds 7.2.1.0
cluster.es.cspoc.rte 7.2.1.0
cluster.es.migcheck 7.2.1.0
cluster.es.server.diag 7.2.1.0
cluster.es.server.events 7.2.1.0
cluster.es.server.rte 7.2.1.0
cluster.es.server.testtool 7.2.1.0
cluster.es.server.utils 7.2.1.0
cluster.license 7.2.1.0
```

<< End of Fileset List >>

`install_all_updates`: Performing `installp` update.

```
+-----+
               Pre-installation Verification...
+-----+
Verifying selections...done
Verifying requisites...done
Results...
```

SUCCESSES

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

```

Selected Filesets
-----
cluster.es.client.clcomd 7.2.1.0          # Cluster Communication Infras...
cluster.es.client.lib 7.2.1.0              # PowerHA SystemMirror Client ...
cluster.es.client.rte 7.2.1.0              # PowerHA SystemMirror Client ...
cluster.es.client.utils 7.2.1.0             # PowerHA SystemMirror Client ...
cluster.es.cspoc.cmds 7.2.1.0              # CSPOC Commands
cluster.es.cspoc.rte 7.2.1.0              # CSPOC Runtime Commands
cluster.es.migcheck 7.2.1.0                # PowerHA SystemMirror Migrati...
cluster.es.server.diag 7.2.1.0              # Server Diags
cluster.es.server.events 7.2.1.0            # Server Events
cluster.es.server.rte 7.2.1.0               # Base Server Runtime
cluster.es.server.testtool 7.2.1.0           # Cluster Test Tool
cluster.es.server.utils 7.2.1.0              # Server Utilities
cluster.license 7.2.1.0                    # PowerHA SystemMirror Electro...

<< End of Success Section >>

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

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

installp: APPLYING software for:
          cluster.license 7.2.1.0

. . . . . << Copyright notice for cluster.license >> . . . . .
Licensed Materials - Property of IBM

5765H3900
Copyright International Business Machines Corp. 2001, 2016.

All rights reserved.
US Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corp.
. . . . . << End of copyright notice for cluster.license >> . . .

Filesets processed: 1 of 13 (Total time: 2 secs).

installp: APPLYING software for:
          cluster.es.migcheck 7.2.1.0

. . . . . << Copyright notice for cluster.es.migcheck >> . . . . .

```

Licensed Materials - Property of IBM

5765H3900

Copyright International Business Machines Corp. 2010, 2016.

All rights reserved.

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

. << End of copyright notice for cluster.es.migcheck >>.

Filesets processed: 2 of 13 (Total time: 6 secs).

installlp: APPLYING software for:

cluster.es.cspoc.rte 7.2.1.0
cluster.es.cspoc.cmds 7.2.1.0

. << Copyright notice for cluster.es.cspoc >>

Licensed Materials - Property of IBM

5765H3900

Copyright International Business Machines Corp. 1985, 2016.

All rights reserved.

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

. << End of copyright notice for cluster.es.cspoc >>.

Filesets processed: 4 of 13 (Total time: 10 secs).

installlp: APPLYING software for:

cluster.es.client.rte 7.2.1.0
cluster.es.client.utils 7.2.1.0
cluster.es.client.lib 7.2.1.0
cluster.es.client.clcomd 7.2.1.0

. << Copyright notice for cluster.es.client >>

Licensed Materials - Property of IBM

5765H3900

Copyright International Business Machines Corp. 1985, 2016.

All rights reserved.

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

Licensed Materials - Property of IBM

5765H3900

Copyright International Business Machines Corp. 2008, 2016.

All rights reserved.

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

. << End of copyright notice for cluster.es.client >>.

Filesets processed: 8 of 13 (Total time: 24 secs).

```
installpp: APPLYING software for:  
          cluster.es.server.testtool 7.2.1.0  
          cluster.es.server.rte 7.2.1.0  
          cluster.es.server.utils 7.2.1.0  
          cluster.es.server.events 7.2.1.0  
          cluster.es.server.diag 7.2.1.0
```

. << Copyright notice for cluster.es.server >>

Licensed Materials - Property of IBM

5765H3900

Copyright International Business Machines Corp. 1985, 2016.

All rights reserved.

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

. << End of copyright notice for cluster.es.server >>.

0513-095 The request for subsystem refresh was completed successfully.
Finished processing all filesets. (Total time: 1 mins 0 secs).

Some configuration files could not be automatically merged into the system
during the installation. The previous versions of these files have been
saved in a configuration directory as listed below. Compare the saved files
and the newly installed files to determine whether you need to recover
configuration data. Consult product documentation to determine how to
merge the data.

Configuration files which were saved in /usr/lpp/save.config:
/usr/es/sbin/cluster/utilities/clexit.rc

Please wait...

```
/usr/sbin/rsct/install/bin/ctposti  
0513-071 The ctrmc Subsystem has been added.  
0513-059 The ctrmc Subsystem has been started. Subsystem PID is 12583318.  
0513-059 The IBM.ConfigRM Subsystem has been started. Subsystem PID is  
11665748.  
cthagsctrl: 2520-208 The cthags subsystem must be stopped.  
0513-029 The cthags Subsystem is already active.  
Multiple instances are not supported.  
0513-095 The request for subsystem refresh was completed successfully.  
done  
+-----+  
          Summaries:  
+-----+
```

Installation Summary

Name	Level	Part	Event	Result
------	-------	------	-------	--------

cluster.license	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.migcheck	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.migcheck	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.cspoc.rte	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.cspoc.cmds	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.cspoc.rte	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.client.rte	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.client.utils	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.client.lib	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.client.clcomd	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.client.rte	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.client.lib	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.client.clcomd	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.server.testtool	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.server.rte	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.server.utils	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.server.events	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.server.diag	7.2.1.0	USR	APPLY	SUCCESS
cluster.es.server.rte	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.server.utils	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.server.events	7.2.1.0	ROOT	APPLY	SUCCESS
cluster.es.server.diag	7.2.1.0	ROOT	APPLY	SUCCESS

```

install_all_updates: Checking for recommended maintenance level 7200-00.
install_all_updates: Executing /usr/bin/oslevel -rf, Result = 7200-00
install_all_updates: Verification completed.
install_all_updates: Log file is /var/adm/ras/install_all_updates.log
install_all_updates: Result = SUCCESS

```

5. Ensure that the file /usr/es/sbin/cluster/netmon.cf exists and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.
6. Start cluster services on node Cass by running **smitty clstart** or **clmgr start node=Cass**.

During the start, a message displays about cluster verification being skipped because of mixed versions, as shown in Figure 5-17.

Cluster services are running at different levels across the cluster. Verification will not be invoked in this environment.

```

Starting Cluster Services on node: Cass
This may take a few minutes. Please wait...
Cass: Nov 8 2016 11:13:48 Starting execution of /usr/es/sbin/cluster/etc/rc.c
luster
Cass: with parameters: -boot -N -A -C interactive -P cl_rc_cluster

```

Figure 5-17 Verification skipped

Important: While the cluster is this mixed cluster state, do *not* make any cluster changes or attempt to synchronize the cluster.

After starting, validate that the cluster is stable before continuing by running the following command:

```
lssrc -ls clstrmgrES |grep -i state
```

7. Repeat the previous steps for node *Jess*. However, when stopping cluster services, choose the Move Resource Groups option, as shown in Figure 5-18.

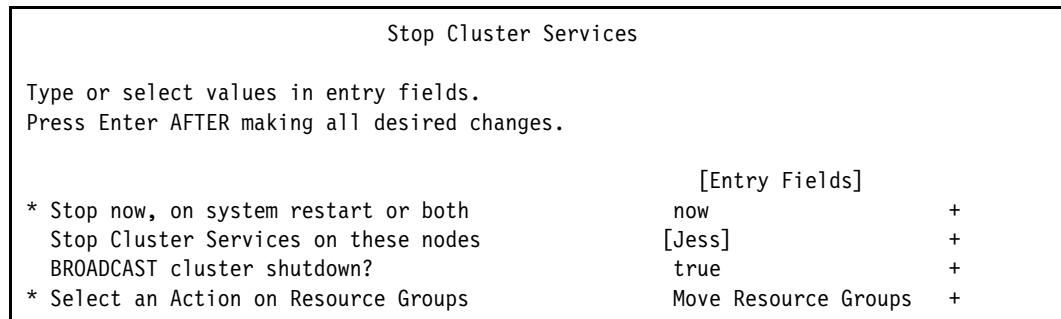


Figure 5-18 Run *clstop* and move the resource group

8. Upgrade AIX (if needed).

Important: If upgrading to AIX 7.2.0, see the [AIX 7.2 Release Notes](#) regarding RSCT filesets when upgrading.

In our scenario, we have supported AIX levels for PowerHA V7.2.1 and do not need to perform this step. But if you do, a restart is required before continuing.

9. Verify that the *clcomd* daemon is active, as shown in Figure 5-19.

```
[root@Jess] /# lssrc -s clcomd
Subsystem      Group          PID      Status
  clcomd       caa           50467008   active
```

Figure 5-19 Verify that *clcomd* is active

10. Upgrade PowerHA on node *Jess*. To upgrade PowerHA, run **smitty update_all**, as shown in Figure 5-4 on page 118, or run the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```

11. Ensure that the file `/usr/es/sbin/cluster/netmon.cf` exists and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.

12. Start cluster services on node *Jess* by running **smitty clstart** or **clmgr start node=Jess**.

13. Verify that the cluster completed the migration on both nodes by checking that the version number is 17, as shown in Example 5-9.

Example 5-9 Verifying the cluster version on both nodes

```
# clcmd odmget HACMPcluster |grep version  
    cluster_version = 17  
    cluster_version = 17  
  
#clcmd odmget HACMPnode |grep version |sort -u  
version = 17
```

Important: Both nodes must show version=17, otherwise the migration did not complete. Call IBM Support.

14. Although the migration is complete, the resource is running on node Cass. If you want, move the RG back to node Jess, as shown in Example 5-10.

Example 5-10 Move the resource group back to node Jess

```
# clmgr move rg demorg node=Jess  
Attempting to move resource group demorg to node Cass.  
  
Waiting for the cluster to process the resource group movement request....  
  
Waiting for the cluster to stabilize.....  
  
Resource group movement successful.  
Resource group demorg is online on node Cass.  
  
Cluster Name: Jess_cluster  
  
Resource Group Name: demorg  
Node           Group State  
-----  
Jess           ONLINE  
Cass          OFFLINE
```

Important: Always test the cluster thoroughly after migrating.

5.3.3 Offline migration from PowerHA V7.2.0

For an offline migration, you can perform many of the steps in parallel on all (both) nodes in the cluster. However, this means that you must plan a full cluster outage.

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Although the version level is different, the steps are identical as though starting from Version 7.2.0.

Complete the following steps:

1. Stop cluster services on both nodes Jess and Cass by running **smitty clstop** and choosing the options that are shown in Figure 5-20. The OK response appears quickly.

As an alternative, you can also stop the entire cluster by running **clmgr stop cluster**.

Make sure the cluster node is in the ST_INIT state by reviewing the **clcmd lssrc -s clstrmgrES|grep state** output.

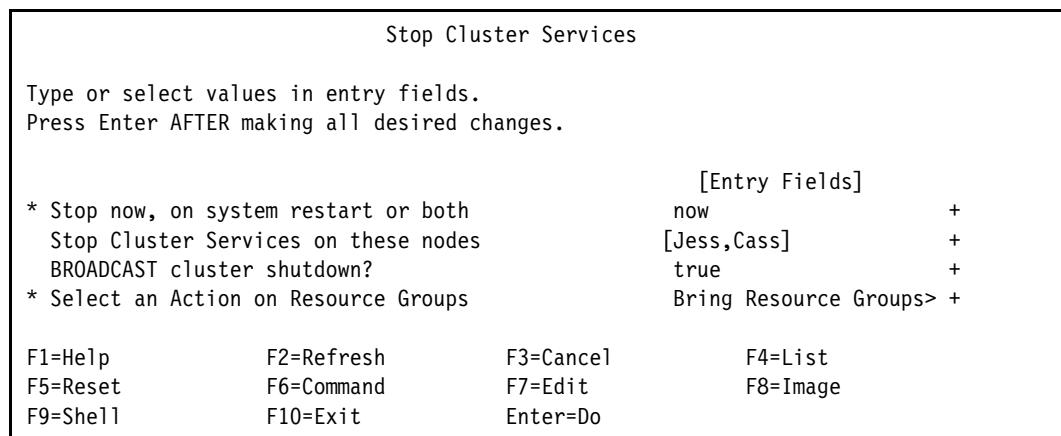


Figure 5-20 Stopping the cluster services

2. Upgrade AIX on both nodes.

Important: If upgrading to AIX 7.2.0, see the [AIX 7.2 Release Notes](#) regarding RSCT filesets when upgrading.

In our scenario, we have supported AIX levels for PowerHA V7.2.1 and do not need to perform this step. But if you do, a restart is required before continuing.

3. Verify that the clcomd daemon is active on both nodes, as shown in Figure 5-21.

```
# clcmd lssrc -s clcomd

-----
NODE Jess
-----
Subsystem      Group          PID      Status
  clcomd        caa           20775182  active

-----
NODE Cass
-----
Subsystem      Group          PID      Status
  clcomd        caa           5177840   active
```

Figure 5-21 Verify that clcomd is active

4. Upgrade to PowerHA V7.2.1 by running `smitty update_all` on both nodes, as shown in Figure 5-4 on page 118, or by running the following command from within the directory in which the updates are (see Example 5-8 on page 128):


```
install_all_updates -vY -d .
```
5. Ensure that the file `/usr/es/sbin/cluster/netmon.cf` exists on all nodes and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one. Restart the cluster on both nodes by running `clmgr start cluster`.
6. Verify that the version numbers show correctly, as shown in Example 5-9 on page 135.

Important: Always test the cluster thoroughly after migrating.

5.3.4 Snapshot migration from PowerHA V7.2.0

For a snapshot migration, you can perform many of the steps in parallel on all (both) nodes in the cluster. However, this migration requires a full cluster outage.

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Complete the following steps:

1. Stop cluster services on both nodes Jess and Cass by running `smitty clstop` and choosing to bring the RG offline. In our case, we chose to stop the entire cluster by running `clmgr stop cluster`. as shown in Figure 5-22.

```
# 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.

Broadcast message from root@Jessica (tty) at 14:08:31 ...

PowerHA SystemMirror on Jessica shutting down. Please exit any cluster applications...
Cass: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.
Cass: 0513-044 The clevmgrdES Subsystem was requested to stop.
Jess: 0513-004 The Subsystem or Group, clinfoES, is currently inoperative.
Jess: 0513-044 The clevmgrdES Subsystem was requested to stop.
...

The cluster is now offline.

Cass: Nov  8 2016 14:08:31 /usr/es/sbin/cluster/utilities/clstop: called with flags -N -g
Jess: Nov  8 2016 14:08:37 /usr/es/sbin/cluster/utilities/clstop: called with flags -N -g
```

Figure 5-22 Stopping cluster services by way of the `clmgr`

Make sure that the cluster node is in the ST_INIT state by reviewing the `clcmd lssrc -ls clstrmgrES|grep state` output.

2. Create a cluster snapshot by running `smitty cm_add_snap.dialog` and completing the options, as shown in Figure 5-23.

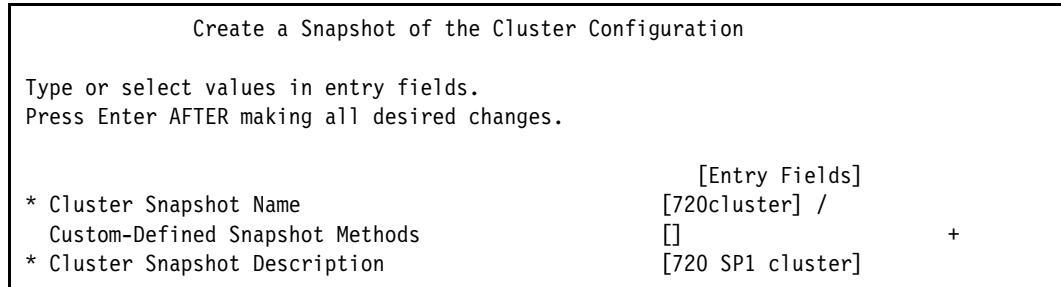


Figure 5-23 Creating a 720 cluster snapshot

3. Upgrade AIX on both nodes.

Important: If upgrading to AIX 7.2.0, see the [AIX 7.2 Release Notes](#) regarding RSCT filesets when upgrading.

In our scenario, we have supported AIX levels for PowerHA V7.2.1 and do not need to perform this step. But if you do, a restart is required before continuing.

4. Verify that the clcomd daemon is active on both nodes, as shown in Figure 5-24.

```
# clcmd lssrc -s clcomd

-----
NODE Jess
-----
Subsystem      Group          PID      Status
clcomd        caa           2102992  active

-----
NODE Cass
-----
Subsystem      Group          PID      Status
clcomd        caa           5110698  active
```

Figure 5-24 Verifying that clcomd is active

5. Uninstall PowerHA 6.1 on both nodes Jess and Cass by running `smitty remove` on `cluster.*`.
6. Install PowerHA V7.2.1 by running `smitty install_all` on both nodes.
7. Convert the previously created snapshot:

```
/usr/es/sbin/cluster/conversion/clconvert_snapshot -v 7.2 -s 720cluster
Extracting ODM's from snapshot file... done.
Converting extracted ODM's... done.
Rebuilding snapshot file... done.
```

8. Restore the cluster configuration from the converted snapshot by running **smitty cm_apply_snap.select** and choosing the snapshot from the menu. It completes the last menu, as shown in Figure 5-25.

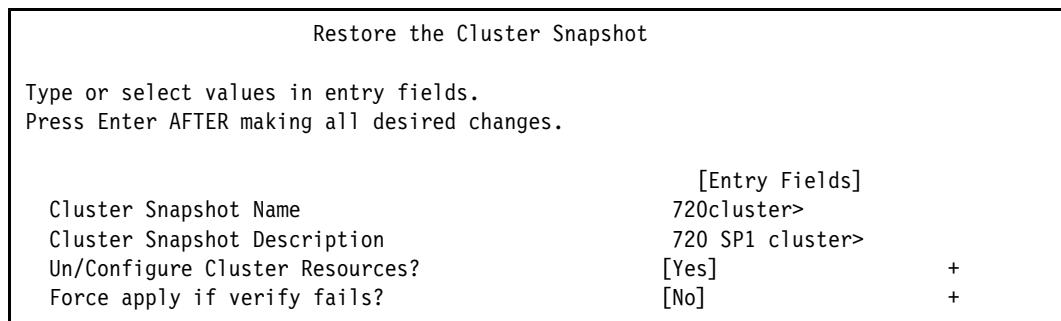


Figure 5-25 Restoring a cluster configuration from a snapshot

The restore process automatically re-creates and synchronizes the cluster.

9. Verify that the version numbers show correctly, as shown in Example 5-1 on page 120.
10. Ensure that the file `/usr/es/sbin/cluster/netmon.cf` exists on all nodes and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.
11. Restart the cluster on both nodes by running **clmgr start cluster**.

Important: Always test the cluster thoroughly after migrating.

5.3.5 Nondisruptive upgrade from PowerHA V7.2.0

This method applies only when the AIX level is already at the appropriate levels to support PowerHA V7.2.1 or later.

Tip: To see a demonstration of performing an offline migration from PowerHA V7.1.3 to PowerHA V7.2.1, see this [YouTube video](#).

Although the version level is different, the steps are identical as though starting from Version 7.2.0.

1. Stop cluster services by performing an unmanage of the RGs on node Cass, as shown in Example 5-11.

Example 5-11 Stop the cluster node with the unmanage option

```
# clmgr stop node=Cass manage=unmanage
```

Warning: "WHEN" must be specified. Since it was not, a default of "now" will be used.

Broadcast message from root@Cass (tty) at 14:27:38 ...

PowerHA SystemMirror on Cass shutting down. Please exit any cluster applications...

Cass: 0513-044 The clevmgrdES Subsystem was requested to stop.

.

"Cass" is now unmanaged.

```
Cass: Nov 8 2016 14:27:38/usr/es/sbin/cluster/utilities/clstop: called with  
flags -N -f
```

2. Upgrade PowerHA (**update_all**) by running the following command from within the directory in which the updates are (see Example 5-8 on page 128):

```
install_all_updates -vY -d .
```
3. Start the cluster services with an automatic manage of the RGs on Cass, as shown in Example 5-12.

Example 5-12 Start the cluster node with the automatic manage option

```
# clmgr start node=Cass  
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 Prior to Starting Cluster Services.
```

```
Cass: start_cluster: Starting PowerHA SystemMirror
```

```
.  
"Cass" is now online.
```

```
Starting Cluster Services on node: Cass  
This may take a few minutes. Please wait...  
Cass: Nov 8 2016 14:43:49Starting execution of  
/usr/es/sbin/cluster/etc/rc.cluster  
Cass: with parameters: -boot -N -b -P cl_rc_cluster -A  
Cass:  
Cass: Nov 8 2016 14:43:49usage: cl_echo messageid (default) messageNov 8 2016  
14:43:49usage: cl_echo messageid (default) messageRETURN_CODE=0
```

Important: Restarting cluster services with the **Automatic** option for managing RGs invokes the application start scripts. Make sure that the application scripts can detect that the application is already running, or copy and put a dummy blank executable script in their place and then copy them back after start.

Repeat the steps on node Jess.

4. Stop the cluster services by performing an unmanage of the RGs on node Jess, as shown Example 5-13.

Example 5-13 Stop the cluster node with the unmanage option

```
# clmgr stop node=Jess manage=unmanage
```

```
Warning: "WHEN" must be specified. Since it was not, a default of "now" will be  
used.
```

```
Broadcast message from root@Jess (tty) at 14:52:58 ...
```

```
PowerHA SystemMirror on Jess shutting down. Please exit any cluster  
applications...
```

```
Jess: 0513-044 The clevmgrdES Subsystem was requested to stop.
```

```
.  
"Jess" is now unmanaged.
```

```
Jess: Nov 8 2016 14:52:58/usr/es/sbin/cluster/utilities/clstop: called with  
flags -N -f
```

5. Upgrade PowerHA (**update_all**) by running the following command from within the directory in which the updates are:

```
install_all_updates -vY -d .
```

A summary of the PowerHA filesets update is shown in Example 5-14.

Example 5-14 Updating the PowerHA filesets

```
+-----+  
          Summaries:  
+-----+  
  
Installation Summary  
-----  


| Name                       | Level   | Part | Event | Result  |
|----------------------------|---------|------|-------|---------|
| cluster.license            | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.migcheck        | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.migcheck        | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.cspoc.rte       | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.cspoc.cmds      | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.cspoc.rte       | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.client.rte      | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.client.utils    | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.client.lib      | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.client.clcomd   | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.client.rte      | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.client.lib      | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.client.clcomd   | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.server.testtool | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.server.rte      | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.server.utils    | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.server.events   | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.server.diag     | 7.2.1.0 | USR  | APPLY | SUCCESS |
| cluster.es.server.rte      | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.server.utils    | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.server.events   | 7.2.1.0 | ROOT | APPLY | SUCCESS |
| cluster.es.server.diag     | 7.2.1.0 | ROOT | APPLY | SUCCESS |


```
install_all_updates: Checking for recommended maintenance level 7200-00.
install_all_updates: Executing /usr/bin/oslevel -rf, Result = 7200-00
install_all_updates: Verification completed.
install_all_updates: Log file is /var/adm/ras/install_all_updates.log
install_all_updates: Result = SUCCESS
```



---


```

6. Start the cluster services by performing an automatic manage of the RGs on Jess, as shown in Example 5-15.

Example 5-15 Start the cluster node with the automatic manage option

```
# clmgr start node=Jess
```

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 Prior to Starting Cluster Services.  
Jess: start_cluster: Starting PowerHA SystemMirror  
...  
"Jess" is now online.
```

```
Starting Cluster Services on node: Jessica  
This may take a few minutes. Please wait...  
Jess: Nov  8 2016 14:54:40Starting execution of  
/usr/es/sbin/cluster/etc/rc.cluster  
Jess: with parameters: -boot -N -b cl_rc_cluster -A  
Jess:  
Jess: Nov  8 2016 14:54:40usage: cl_echo messageid (default) messageNov  8 2016  
14:54:40usage: cl_echo messageid (default) messageRETURN_CODE=0
```

Important: Restarting cluster services with the **Automatic** option for managing RGs invokes the application start scripts. Make sure that the application scripts can detect that the application is already running, or copy and put a dummy blank executable script in their place and then copy them back after start.

7. Verify that the version numbers show correctly, as shown in Example 5-1 on page 120.
8. Ensure that the file /usr/es/sbin/cluster/netmon.cf exists on all nodes and that it contains at least one pingable IP address because the installation or upgrade of PowerHA filesets can overwrite this file with an empty one.



Resource Optimized High Availability

This chapter describes one feature: Resource Optimized High Availability (ROHA). This feature is a new feature of PowerHA SystemMirror Standard and Enterprise Edition V7.2.

This chapter covers the following topics:

- ▶ Resource Optimized High Availability concept and terminology
- ▶ New PowerHA SystemMirror SMIT configuration panels for Resource Optimized High Availability
- ▶ New PowerHA SystemMirror verification enhancement for Resource Optimized High Availability
- ▶ Planning for one Resource Optimized High Availability cluster environment
- ▶ Resource acquisition and release process introduction
- ▶ Introduction to resource acquisition
- ▶ Introduction to release of resources
- ▶ Example 1: Setting up one Resource Optimized High Availability cluster (without On/Off CoD)
- ▶ Test scenarios of Example 1 (without On/Off CoD)
- ▶ Example 2: Setting up one Resource Optimized High Availability cluster (with On/Off CoD)
- ▶ Test scenarios for Example 2 (with On/Off CoD)
- ▶ Hardware Management Console high availability introduction
- ▶ Test scenario for HMC failover
- ▶ Managing, monitoring, and troubleshooting

6.1 Resource Optimized High Availability concept and terminology

With this feature, PowerHA SystemMirror can manage dynamic LPAR (DLPAR) and Capacity of Demand (CoD) resources. CoD resources are composed of Enterprise Pool CoD (EPCoD) resources and On/Off CoD resources.

Enterprise Pool CoD resources

EPCoD resources are resources that can be freely moved among servers in the same pool where the resources are best used. Physical resources (such as CPU or memory) are not moved between servers; what is moved is the privilege to use them. You can grant this privilege to any server of the pool so that you can flexibly manage the pool of resources and acquire the resources where they are most needed.

On/Off CoD resource

On/Off CoD resources are preinstalled and inactive (and unpaid for) physical resources for a given server: Processors or memory capacity. On/Off CoD is a type of CoD license enabling temporary activation of processors and memory. PowerHA SystemMirror can dynamically activate these resources and can make them available to the system so that they are allocated when needed to the LPAR through a DLPAR operation.

Dynamic logical partitioning

DLPAR represents the facilities in some IBM Power Systems that provide the ability to logically attach and detach a managed system's resources to and from an LPAR's operating system without restarting.

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 way, you can tune the capacity of the logical partition flexibly when your application requires more resources, without having to pay for idle capacity until you need it (for On/Off CoD), or without keeping acquired resources if you do not use them (for Enterprise Pool CoD).

You can configure cluster resources so that the logical partition with minimally allocated resources serves as a standby node, and the application is on another LPAR node that has more resources than the standby node. This way, you do not use any additional resources that the frames have until the resources are required by the application.

PowerHA SystemMirror uses the system-connected HMC to perform DLPAR operation and manage CoD resources.

Table 6-1 displays all available types of the CoD offering. Only two of them are dynamically managed and controlled by PowerHA SystemMirror: EPCoD and On/Off CoD.

Table 6-1 CoD offering and PowerHA

CoD offering	PowerHA SystemMirror V6.1 Standard and Enterprise Edition	PowerHA SystemMirror V7.1 or V7.2 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

CoD offering	PowerHA SystemMirror V6.1 Standard and Enterprise Edition	PowerHA SystemMirror V7.1 or V7.2 Standard and Enterprise Edition
On/Off CoD (temporary) Processor	Yes	Yes
Utility CoD (temporary) Memory and Processor	Utility CoD automatically is 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 DLPAR operation.	
CUoD (permanent) Memory & Processor	CUoD are used if available through DLPAR operation. PowerHA does not handle this kind of resource directly.	

Trial CoD

Trial CoD are temporary resources, but they are not set to On or Off to follow dynamic needs. When Trial CoD standard or exception code is entered into the HMC, these resources are On immediately, and elapsed time starts immediately. The amount of resources that is granted by Trial CoD directly enters the available DLPAR resources. It is as though these were configured as DLPAR resources.

Therefore, PowerHA SystemMirror can dynamically control the Trial CoD resource after customer manually enters a code to activate the resource through HMC.

6.1.1 Environment requirement for Resource Optimized High Availability

Here are the requirements to implement ROHA:

- ▶ PowerHA SystemMirror V7.2 Standard Edition or Enterprise Edition
- ▶ AIX 6.1 TL09 SP5, or AIX 7.1 TL03 SP5, or AIX 7.1 TL4 or AIX 7.2 or later
- ▶ HMC requirements:
 - To use the Enterprise Pool CoD license, your system must be using HMC 7.8 firmware or later.
 - Configure the backup HMC for EPCoD with high availability.
 - For the EPCoD User Interface (UI) in HMC, the HMC must have a minimum of 2 GB of memory.
- ▶ Hardware requirements for using Enterprise Pool CoD license:
 - POWER7+: 9117-MMD (770 D model), 9179-MHD (780 D model), that uses FW780.10 or later.
 - POWER8: 9119-MME (E870), 9119-MHE (E880), that uses FW820 or later.

6.2 New PowerHA SystemMirror SMIT configuration panels for Resource Optimized High Availability

To support the ROHA feature, PowerHA SystemMirror provides some new SMIT menu and `clmgr` command options. These options include the following functions:

- ▶ HMC configuration
- ▶ Hardware Resource Provisioning for Application Controller
- ▶ Cluster tunables configuration

Figure 6-1 shows a summary of the SMIT menu navigation for all new ROHA panels. For the new options of `clmgr` command, see 6.14.1, “The `clmgr` interface to manage Resource Optimized High Availability” on page 237.

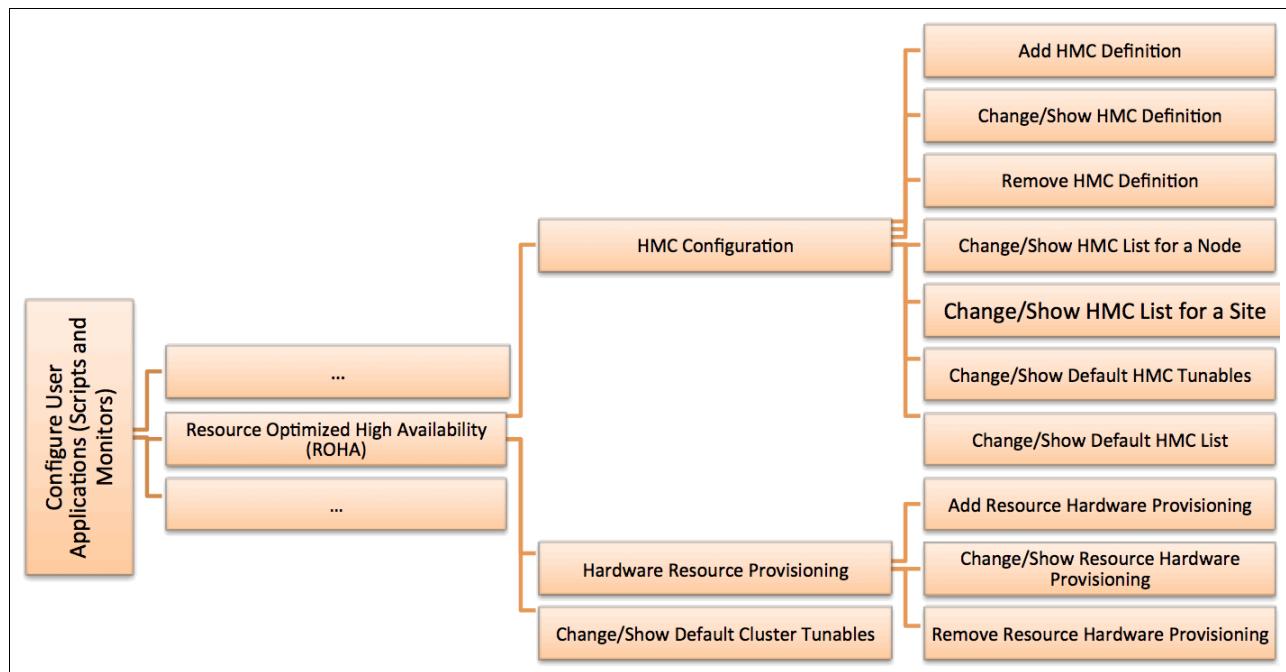


Figure 6-1 New Resource Optimized High Availability panels

6.2.1 Entry point to Resource Optimized High Availability

Start `smit sysmirror` and select **Cluster Applications and Resources** → **Resources** → **Configure User Applications (Scripts and Monitors)**. This panel is a menu window with a title menu option and four item menu options. Only the third item is the entry point to ROHA configuration (Figure 6-2).

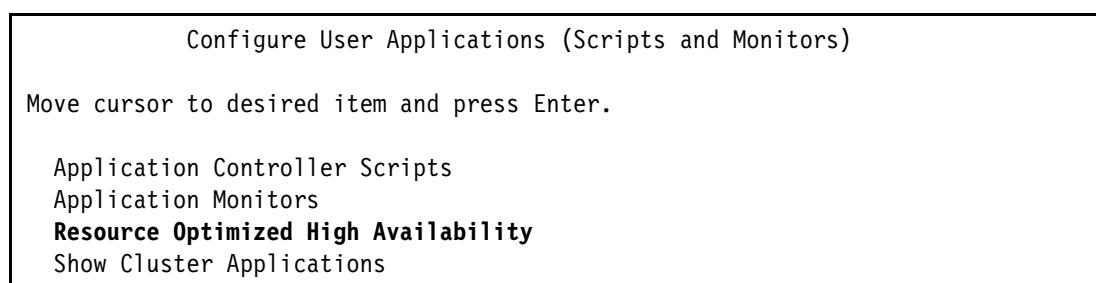


Figure 6-2 Entry point to the Resource Optimized High Availability menu

Table 6-2 shows the context-sensitive help for the ROHA entry point.

Table 6-2 Context-sensitive help for the Resource Optimized High Availability entry point

Name and fast path	Context-sensitive help (F1)
Resource Optimized High Availability # smitty cm_cfg_roha	Choose this option to configure ROHA. ROHA performs dynamic management of hardware resources (memory and CPU) for the account of PowerHA SystemMirror. This dynamic management of resources uses three types of mechanism: DLPAR, On/Off CoD, and Enterprise Pool CoD. If the resources that are available on the central electronic complex are not sufficient, and cannot be obtained through a DLPAR operation, it is possible to fetch external pools of resources that are provided by CoD: Either On/Off or Enterprise Pool. On/Off CoD can result in extra costs, and a formal agreement from the user is required. The user must configure Hardware Management Consoles (HMC) to for acquisition/release of resources.

6.2.2 Resource Optimized High Availability panel

Start **smit sysmirror** and select **Cluster Applications and Resources → Resources → Configure User Applications (Scripts and Monitors) → Resource Optimized High Availability**. The next panel is a menu window with a title menu option and three item menu options. Its fast path is **cm_cfg_roha** (Figure 6-3).

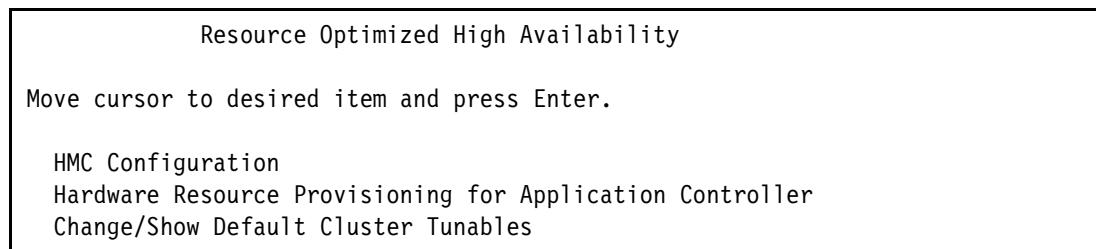


Figure 6-3 Resource Optimized High Availability panel

Table 6-3 shows the help information for the ROHA panel.

Table 6-3 Context-sensitive help for the Resource Optimized High Availability panel

Name and fast path	Context-sensitive help (F1)
HMC Configuration # smitty cm_cfg_hmc	This option configures the HMC that is used by your cluster configuration, and to optionally associate the HMC to your cluster's nodes. If no HMC is associated with a node, PowerHA SystemMirror uses the default cluster configuration.
Change/Show Hardware Resource Provisioning for Application Controller # smitty cm_cfg_hr_prov	This option changes or shows CPU and memory resource requirements for any Application Controller that runs in a cluster that uses DLPAR, CoD, or Enterprise Pool CoD capable nodes, or a combination.
Change/Show Default Cluster Tunables # smitty cm_cfg_def_c1_tun	This option modifies or views the DLPAR, CoD, and Enterprise Pool CoD configuration parameters.

6.2.3 HMC configuration

Start `smit sysmirror`. Select **Cluster Applications and Resources** → **Resources** → **Configure User Applications (Scripts and Monitors)** → **Resource Optimized High Availability** → **HMC Configuration**. The next panel is a menu window with a title menu option and seven item menu options. Its fast path is `cm_cfg_hmc` (Figure 6-4).

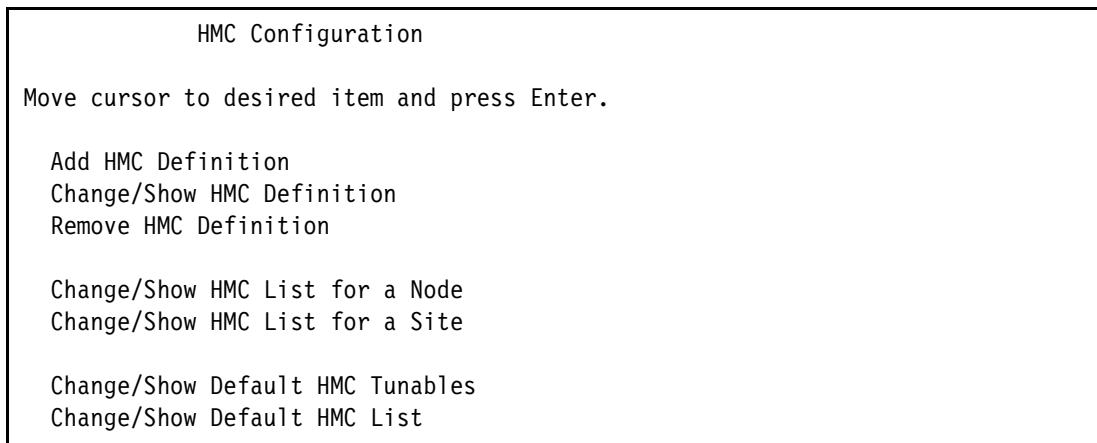


Figure 6-4 HMC configuration menu

Table 6-4 shows the help information for the HMC configuration.

Table 6-4 Context-sensitive help for the HMC configuration

Name and fast path	Context-sensitive help (F1)
Add HMC Definition # smitty <code>cm_cfg_add_hmc</code>	Choose this option to add an HMC and its communication parameters, and add this new 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 <code>cm_cfg_ch_hmc</code>	Choose this option to modify or view an HMC host name and communication parameters.
Remove HMC Definition # smitty <code>cm_cfg_rm_hmc</code>	Choose this option to remove an HMC, and then remove it from the default list.
Change/Show HMC List for a Node # smitty <code>cm_cfg_hmcs_node</code>	Choose this option to modify or view the list of an HMC of a node.
Change/Show HMC List for a Site # smitty <code>cm_cfg_hmcs_site</code>	Choose this option to modify or view the list of an HMC of a site.

Name and fast path	Context-sensitive help (F1)
Change/Show Default HMC Tunables # smitty cm_cfg_def_hmc_tun	Choose this option to modify or view the HMC default communication tunables.
Change/Show Default HMC List # smitty cm_cfg_def_hmc	Choose 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.

HMC Add/Change/Remove Definition

Note: Before you add HMC, you must build password-less communication from AIX nodes to the HMC. For more information, see 6.4.1, “Consideration before Resource Optimized High Availability configuration” on page 163.

To add HMC, select **Add HMC Definition**. The next panel is a dialog window with a title dialog header and several dialog command options. Its fast path is **cm_cfg_add_hmc**. Each item has a context-sensitive help window that you access by pressing F1, and can have an associated list (press F4).

Figure 6-5 shows the menu to add the HMC definition and its entry fields.

Add HMC Definition

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

* HMC name	[Entry Fields] [] +
DLPAR operations timeout (in minutes)	[] #
Number of retries	[] #
Delay between retries (in seconds)	[] #
Nodes	[] +
Sites	[] +
Check connectivity between HMC and nodes	Yes

Figure 6-5 Add HMC Definition menu

Table 6-5 shows the help and information list for adding the HMC definition.

Table 6-5 Context-sensitive help and associated list for Add HMC Definition menu

Name	Context-sensitive help (F1)	Associated list (F4)
HMC name	Enter the host name for the HMC. An IP address is also accepted here. Both IPv4 and IPv6 addresses are supported.	Yes (single-selection). The list is obtained by running the following command: <code>/usr/sbin/rsct/bin/rmcdo mainstatus -s ctrmc -a IP</code>
DLPAR operations timeout (in minutes)	Enter a timeout in minutes by using DLPAR commands run on an HMC (-w parameter). This -w parameter exists only on the <code>chhwres</code> command when allocating or releasing resources. It 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 -1 is displayed in this field, it indicates that the default value is used.	None
Number of retries	Enter a number of times one HMC command is retried before the HMC is considered as non-responding. 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 -1 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 -1 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 as 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 connectivity between the HMC and nodes	Select Yes to check communication links between nodes and HMC.	<Yes> <No>. The default is Yes.

If Domain Name Service (DNS) is configured in your environment and DNS can do resolution for HMC IP and host name, then you can use F4 to select one HMC to perform the add operation.

Figure 6-6 shows an example of selecting one HMC from the list to perform the add operation.

HMC name
Move cursor to desired item and press Enter.
e16hmc1 is 9.3.207.130
e16hmc3 is 9.3.207.133
F1=Help F2=Refresh F3=Cancel
Esc+8=Image Esc+0=Exit Enter=Do
/=Find n=Find Next

Figure 6-6 Select one HMC from the HMC list to perform an add HMC operation

PowerHA SystemMirror also supports entering the HMC IP address to add the HMC. Figure 6-7 shows an example of entering one HMC IP address to add the HMC.

Add HMC Definition	
Type or select values in entry fields.	
Press Enter AFTER making all desired changes.	
* HMC name	[Entry Fields] [9.3.207.130] +
DLPAR operations timeout (in minutes)	[]
Number of retries	[]
Delay between retries (in seconds)	[]
Nodes	[] +
Sites	[] +
Check connectivity between HMC and nodes	Yes

Figure 6-7 Enter one HMC IP address to add an HMC

Change>Show HMC Definition

To show or modify an HMC, select **Change>Show HMC Definition**. The next panel is a selector window with a selector header that lists all existing HMC names. Its fast path is **cm_cfg_ch_hmc** (Figure 6-8).

HMC name
Move cursor to desired 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 6-8 Select an HMC from a list during a change or show HMC configuration

Press Enter on an existing HMC to modify it. The next panel is the one that is presented in Figure 6-9. You cannot change the name of the HMC.

Change/Show HMC Definition	
Type or select values in entry fields. Press Enter AFTER making all desired 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 6-9 Change/Show HMC Definition of SMIT menu

Remove HMC Definition

To delete an HMC, select **Remove HMC Definition**. The panel that is shown in Figure 6-10 is the same selector window. Press Enter on an existing HMC name to remove it. Its fast path is **cm_cfg_rm_hmc**.

HMC name		
Move cursor to desired item and press Enter.		
e16hmc1 e16hmc3		
F1=Help Esc+8=Image /=Find	F2=Refresh Esc+0=Exit n=Find Next	F3=Cancel Enter=Do

Figure 6-10 Select one HMC to remove

Figure 6-11 shows the removed HMC definition.

Remove HMC Definition
Type or select values in entry fields. Press Enter AFTER making all desired changes.
* HMC name [Entry Fields] e16hmc1

Figure 6-11 Remove one HMC

Change>Show HMC List for a Node

To show or modify the HMC list for a node, select **Change>Show HMC List for a Node**. The next panel (Figure 6-12) is a selector window with a selector header that lists all existing nodes. Its fast path is **cm_cfg_hmc_node**.

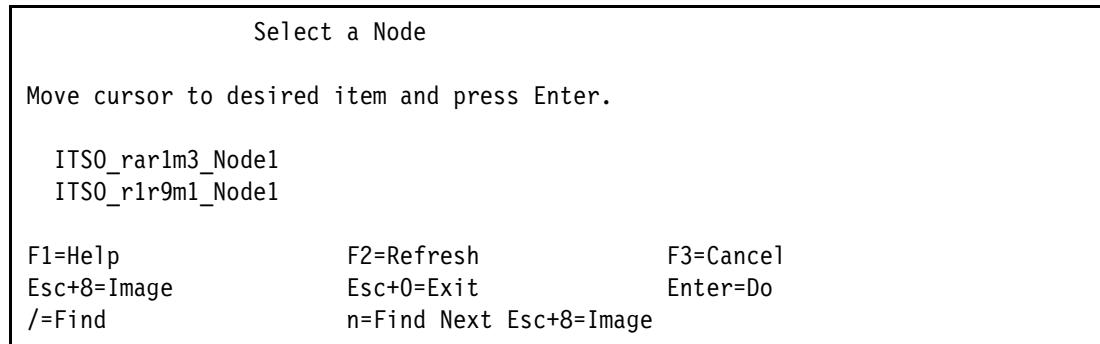


Figure 6-12 Select a node to change

Press Enter on an existing node to modify it. The next panel (Figure 6-13) is a dialog window with a title dialog header and two dialog command options.

You cannot add or remove an HMC from this list. You can only reorder (set in the correct precedence order) the HMCs that are used by the node.

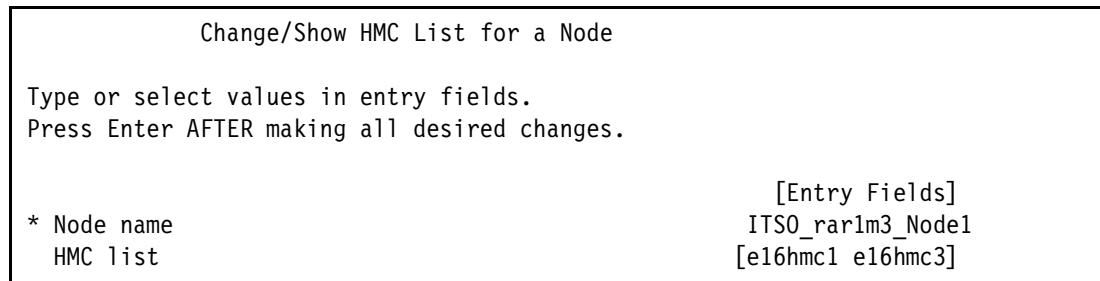


Figure 6-13 Change>Show HMC list for a Node

Table 6-6 shows the help information to change or show the HMC list for a node.

Table 6-6 Context-sensitive help for Change or Show HMC list for a Node

Name and fast path	Context-sensitive help (F1)
Node name	This is the 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

To show or modify the HMC list for a node, select **Change>Show HMC List for a Site**. The next panel (Figure 6-14) is a selector window with a selector header that lists all existing sites. Its fast path is **cm_cfg_hmcs_site**.

Select a Site

Move cursor to desired item and press Enter.

site1
site2

F1=Help F2=Refresh F3=Cancel
Esc+8=Image Esc+0=Exit Enter=Do
/=Find n=Find Next

Figure 6-14 Select a Site menu for Change/Show HMC List for as Site

Press Enter on an existing site to modify it. The next panel (Figure 6-15) is a dialog window with a title dialog header and two dialog command options.

Change/Show HMC List for a Site

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

* Site Name	[Entry Fields]
HMC list	site1 [e16hmc1 e16hmc3]

Figure 6-15 Change>Show HMC List for a Site menu

You cannot add or remove an HMC from the list. You can only reorder (set in the correct precedence order) the HMCs used by the site. See Table 6-7.

Table 6-7 Site and HMC usage list

Name and fast path	Context-sensitive help (F1)
Site name	This is 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

To show or modify default HMC communication tunables, select **Change>Show Default HMC Tunables**. The next panel (Figure 6-16) is a dialog window with a title dialog header and three dialog command options. Its fast path is **cm_cfg_def_hmc_tun**. Each item has a context-sensitive help window (press F1) and can have an associated list (press F4).

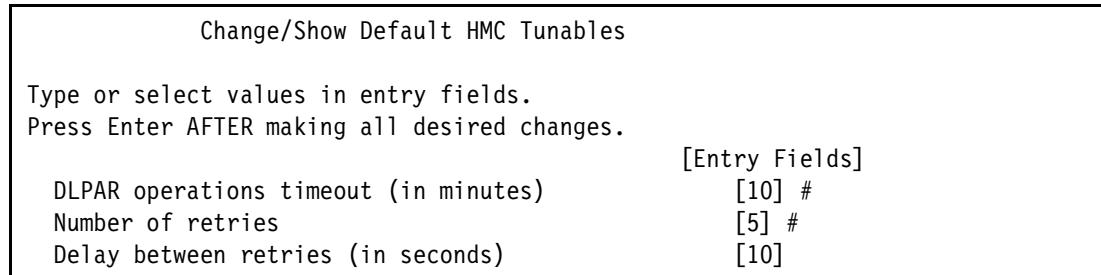


Figure 6-16 Change>Show Default HMC Tunables menu

Change>Show Default HMC List

To show or modify the default HMC list, select **Change>Show Default HMC List**. The next panel (Figure 6-17) is a dialog window with a title dialog header and one dialog command option. Its fast path is **cm_cfg_def_hmcsl**. Each item has a context-sensitive help window (press F1) and can have an associated list (press F4).

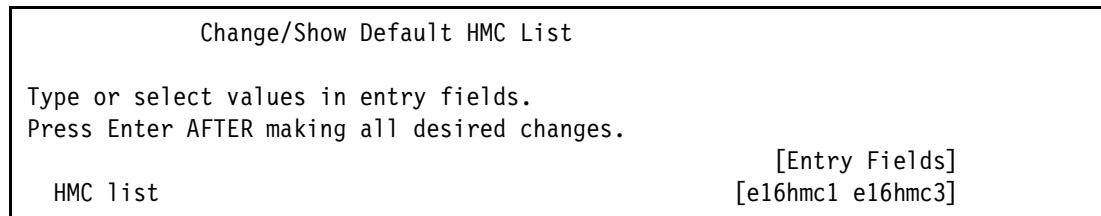


Figure 6-17 Change>Show Default HMC list menu

6.2.4 Hardware resource provisioning for application controller

To provision hardware, complete the following steps:

1. Start **smit sysmirror**. Select **Cluster Applications and Resources → Resources → Configure User Applications (Scripts and Monitors) → Resource Optimized High Availability → Hardware Resource Provisioning for Application Controller**. The next panel (Figure 6-18) is a menu window with a title menu option and three item menu options.

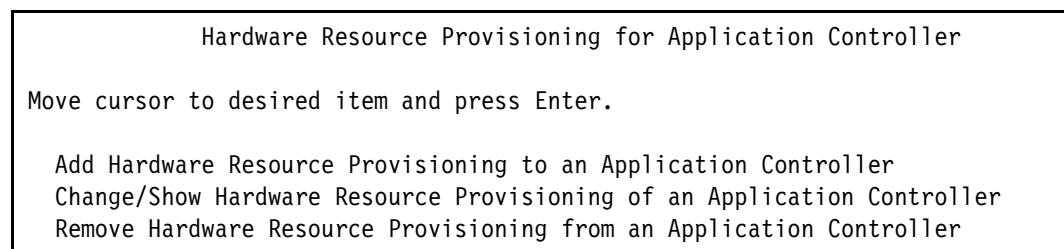


Figure 6-18 Hardware Resource Provisioning for Application Controller menu

2. Choose 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**.

In case you choose Add or Change/Show, the On/Off CoD Agreement is displayed, as shown in Figure 6-19. However, this is displayed only if the user has not yet agreed to it. If the user has already agreed to it, it is not displayed.

On/Off CoD Agreement

Figure 6-19 is a dialog panel with a dialog header and one dialog command option.

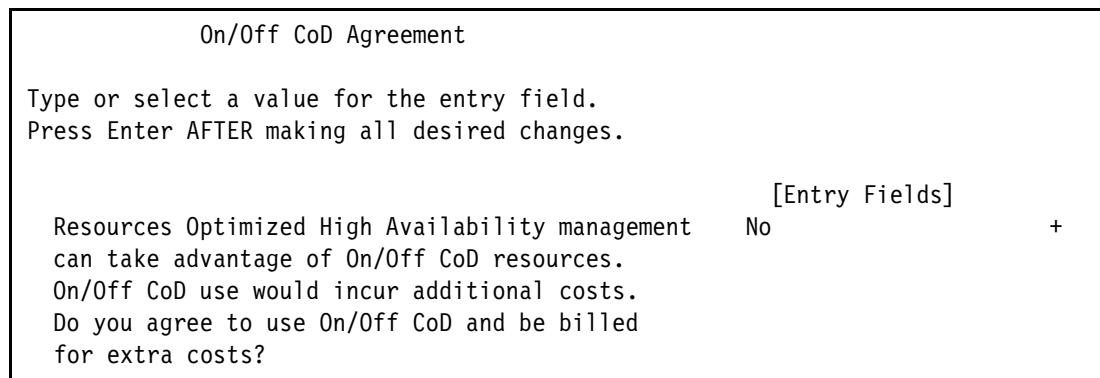


Figure 6-19 On/Off CoD Agreement menu

To accept the On/Off CoD Agreement, complete the following steps:

1. Enter Yes to have PowerHA SystemMirror use On/Off Capacity On Demand (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 have 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 half Enterprise Pool CoD, keep the default value of No.
 - If there is not Enterprise Pool CoD or On/Off CoD, PowerHA manages only the server's permanent resources through DLPAR, so also keep the default value.

This option can be modified later in the **Change>Show Default Cluster Tunables** panel, as shown in Figure 6-22 on page 160.

Add Hardware Resource Provisioning to an Application Controller

The panel that is shown in Figure 6-20 is a selector window with a selector header that lists all existing application controllers.

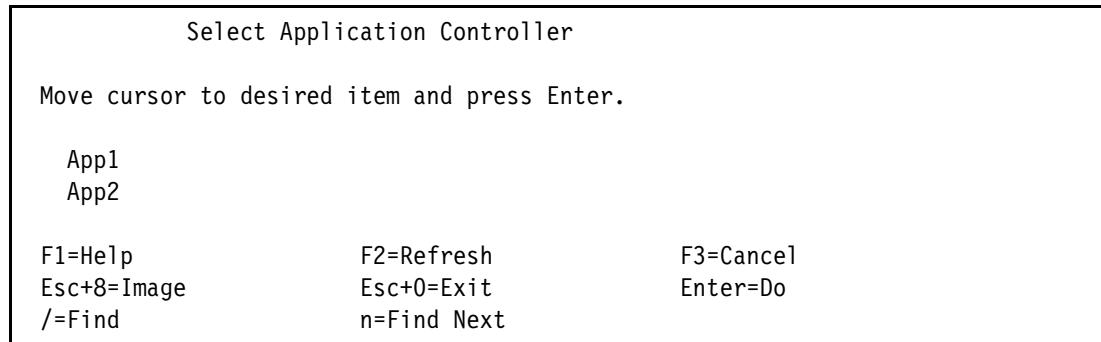


Figure 6-20 Select Application Controller menu

To create a *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 6-21.

To modify or remove a Hardware Resource Provisioning for an Application Controller, the list displays application controllers that already have hardware resource provisioning.

Press Enter on an existing application controller to modify it. The next panel is a dialog window with a title dialog header and three dialog command options. Each item has a context-sensitive help window (press F1) and can have an associated list (press F4).

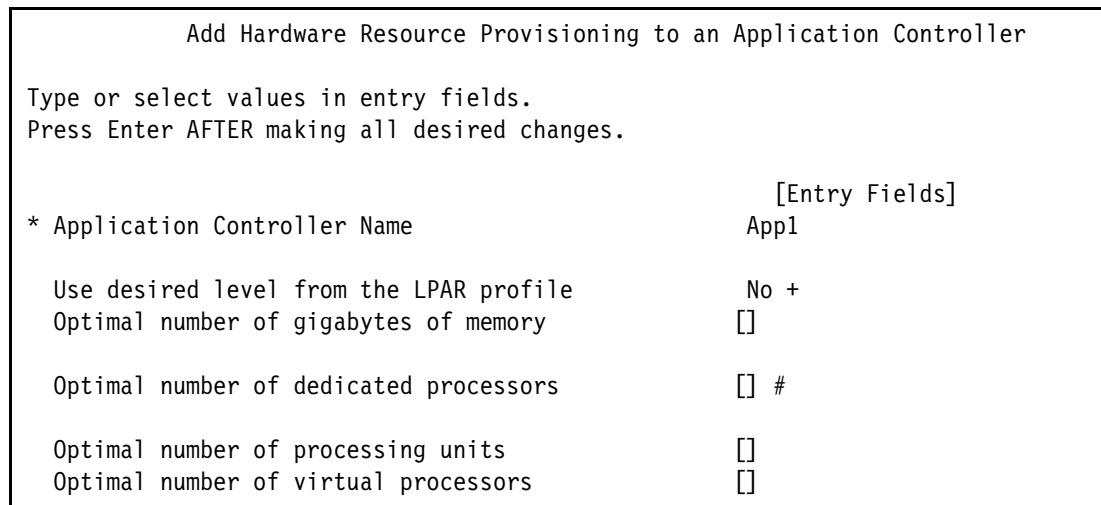


Figure 6-21 Add Hardware Resource Provisioning to an Application Controller menu

Table 6-8 shows the help for adding hardware resources.

Table 6-8 Context-sensitive help for add hardware resource provisioning

Name and fast path	Context-sensitive help (F1)
Application Controller Name	This is the application controller for which you configure DLPAR and CoD resource provisioning.
Use desired level from the LPAR profile	<p>There is no default value. You must 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 desired level of the LPAR's profile. By choosing Yes, you trust the desired level of 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 enable you to better control 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 having this tunable set to Yes, the allocation that is performed lets the LPAR reach the LPAR desired 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 specified. In this case, the allocation that is performed lets the LPAR reach the desired value of the profile that is added to the optimal values.</p>
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 desired level from the LPAR profile value is set to No.</p> <p>Enter the value in multiples of $\frac{1}{4}$, $\frac{1}{2}$, $\frac{3}{4}$, 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 resource group (RG) recovery actions to move the RG with this application to another node. 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 LPAR with Dedicated Processing Mode.</p> <p>This Optimal number of dedicated processors value can be set only if the Used desired 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. Alternatively, PowerHA SystemMirror can allocate fewer CPUs depending on the Start RG even if resources are insufficient cluster tunable.</p> <p>For more information about how to acquire mobile resources at the RG onlining stage, see 6.6, "Introduction to resource acquisition" on page 175.</p> <p>For more information about how to release mobile resources at the RG offline stage, see 6.7, "Introduction to release of resources" on page 185.</p>

Name and fast path	Context-sensitive help (F1)
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. This attribute is only for nodes running on LPAR with Shared Processing Mode.</p> <p>This Optimal number of processing units value can be set only if the Used desired level from the LPAR profile value is set to No. 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. Alternatively, PowerHA SystemMirror can allocate fewer CPUs depending on the Start RG even if resources are insufficient cluster tunable. For more information about how to acquire mobile resources at the RG onlining stage, see 6.6, “Introduction to resource acquisition” on page 175. For more information about how to release mobile resources at the RG offline stage, see 6.7, “Introduction to release of resources” on page 185.</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. This attribute is only for nodes running on LPAR with Shared Processing Mode.</p> <p>This Optimal number of dedicated or virtual processors value can be set only if the Used desired 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. 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, select **Change>Show**. The next panel is the same selector window, as shown in Figure 6-21 on page 157. Press Enter on an existing application controller to modify it. The next panel is the same dialog window shown in Figure 6-21 on page 157 (except the title, which is different).

To delete an application controller configuration, select **Remove**. The next panel is the same selector window that was shown previously. Press Enter on an existing application controller to remove it.

If Use desired 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.

6.2.5 Change>Show Default Cluster Tunable

Start smit sysmirror. 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 6-22) is a dialog window with a title dialog header and seven dialog command options. Each item has a context-sensitive help window (press F1) and can have an associated list (press F4). Its fast path is **cm_cfg_def_cl_tun**.

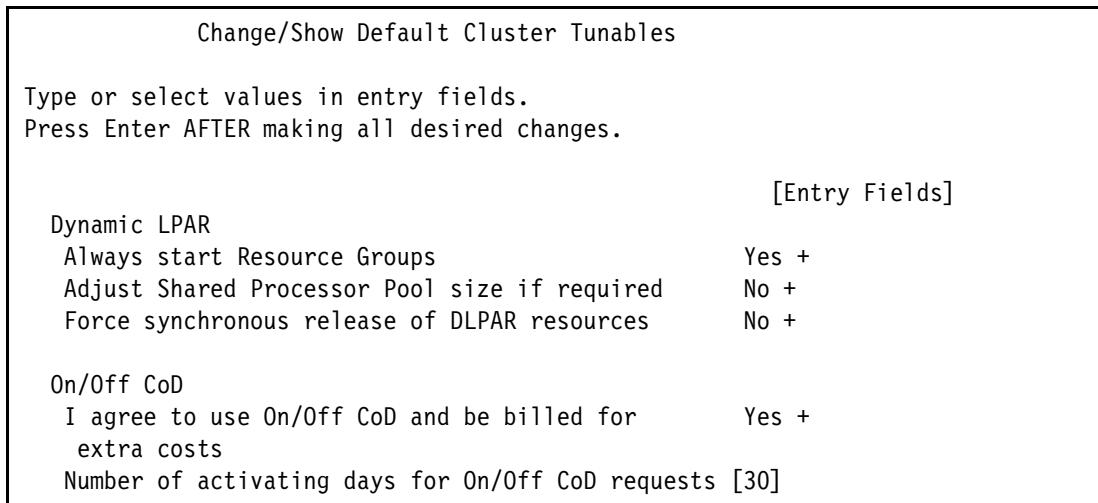


Figure 6-22 Change>Show Default Cluster Tunables menu

Table 6-9 shows the help for the cluster tunables.

Table 6-9 Context-sensitive help for Change>Show Default Cluster Tunables

Name and fast path	Context-sensitive help (F1)
Always start Resource Groups	Enter Yes to have PowerHA SystemMirror start RGs even if there is any error in ROHA resources activation. This 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. Enter No to prevent starting Resources Groups if any error during ROHA resources acquisition. <i>The default is Yes.</i>
Adjust Shared Processor Pool size if required	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 electronic complex so that changing the maximum size of a particular Shared-Processors Pool is not done to the detriment of other Shared-Processors Pools. <i>The default is No.</i>
Force synchronous release of DLPAR resources	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 leading practice is to have asynchronous release in order to not delay the takeover. <i>The default is No.</i>

Name and fast path	Context-sensitive help (F1)
I agree to use On/Off CoD and be billed for extra costs	Enter Yes to have PowerHA SystemMirror use On/Off Capacity On Demand (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. <i>The default is No.</i>
Number of activating days for On/Off CoD requests	Enter a number of activating days for On/Off CoD requests. If the requested available resources are insufficient for this duration, then the longest-can-do allocation is performed. Try to allocate the amount of resources that is requested for the longest duration. To do that, 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 not yet activated. <i>The default is 30.</i>

6.3 New PowerHA SystemMirror verification enhancement for Resource Optimized High Availability

The ROHA function allows PowerHA SystemMirror to automatically or manually check for environment discrepancies. The **c1verify** tool is improved to check ROHA-related configuration integrity.

Customers can use the verification tool to ensure that their environment is correct regarding their ROHA setup. Discrepancies are called out by PowerHA SystemMirror, and the tool assists customers to 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* of 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 6-10 shows the general verification list.

Table 6-10 General verification list

Item	Configuration time	Synchronization time
Check that all RG active and standby nodes are on different CECs. This enables the asynchronous mode of releasing resources.	Info	Warning
This code cannot run on an IBM POWER4.	Error	Error

HMC communication verification

Table 6-11 shows the HMC communication verification list.

Table 6-11 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 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 non-SSH accessible HMC.	Warning	Error
Check that all HMCs share the same level (the same version of HMC).	Warning	Warning
Check that all HMCs administer the central electronic complex hosting the current node. Configure two HMCs administering the central electronic complex hosting the current node. If not, PowerHA gives a warning message.	Warning	Warning
Check whether the HMC level supports FSP Lock Queuing.	Info	Info

Capacity on Demand verification

Table 6-12 shows the CoD verification.

Table 6-12 Capacity on Demand verification

Item	Configuration Time	Synchronization Time
Check that all CECs are CoD capable.	Info	Warning
Check whether CoD is enabled.	Info	Warning

Power enterprise pool verification

Table 6-13 shows the enterprise pool verification list.

Table 6-13 Power enterprise pool verification

Item	@Info	@Sync
Check that all CECs 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. This enables the asynchronous mode of releasing resources.	Info	Info
Check that all HMCs are at level 7.8 or later.	Info	Warning
Check that the central electronic complex has unlicensed resources.	Info	Warning

Resource provisioning verification

Table 6-14 shows the resource provisioning verification information.

Table 6-14 Resource provisioning verification

Item	@info	@Sync
Check that for one given node the total of optimal memory (of RG on this node) that is added to the profile's minimum does not exceed the profile's maximum.	Warning	Error
Check that for one given node the total of optimal CPU (of RG on this node) that is added to the profile's minimum does not exceed the profile's maximum.	Warning	Error
Check that for one given node the total of optimal PU (of RG 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

6.4 Planning for one Resource Optimized High Availability cluster environment

Before completing the ROHA configuration, read the following considerations.

6.4.1 Consideration before Resource Optimized High Availability configuration

This section describes a few considerations to know before a ROHA configuration.

Tips for an Enterprise Pool license

If you have ordered IBM Power Systems Enterprise Pool license for your servers, and you want to use the resource with your PowerHA SystemMirror cluster, then you must create the Enterprise Pool manually.

Before you create the Enterprise Pool, get the configuration extensible markup language (XML) file from the IBM CoD, and the deactivation code from the IBM CoD project office at this [IBM website](#).

The configuration XML file is used to enable and generate mobile resources.

The deactivation code is used to deactivate some of the permanent resources to inactive mode. The number is the same independent of how many mobile resources are on the server's order.

For example, in one order, there are two Power Systems servers. Each one has 16 static CPUs, eight mobile CPUs, and eight inactive CPUs, for a total of 32 CPUs. When you power them on the first time, you see that each server has 24 permanent CPUs, 16 static CPUs, plus 8 mobile CPUs.

After you create the Enterprise Pool with the XML configuration file, you see that there are 16 mobile CPUs that are generated in the Enterprise Pool, but the previous eight mobile CPUs are still in permanent status in each server. This results in the server's status being different from its original order. This brings some issues in future post-sales activities.

There are two steps to complete the Enterprise Pool implementation:

1. Create the Enterprise Pool with the XML configuration file.
2. Deactivate some permanent resources (the number is the same as mobile resources) to inactive with the deactivation code.

After you finish these two steps, each server has 16 static CPUs and 16 inactive CPUs, and the Enterprise Pool has 16 mobile CPUs. Then, the mobile CPUs can be assigned to each of the two servers through the HMC GUI or the command-line interface.

Note: These two steps will be combined into one step in the future. At the time of writing, you must perform each step separately.

How to get the deactivation code and use it

The following steps explain how to get the deactivation code and how to use it:

1. Send an email to the IBM CoD project office (pcod@us.ibm.com). You need to provide the following information or attach the servers order:
 - Customer name
 - Each server's system type and serial number
 - Configuration XML file
2. In reply to this note, you receive from the CoD project office a de-activation code for the servers. The de-activation code lowers the number of activated resources to align it with your server order.

Note: This de-activation code updates the IBM CoD website after you receive the note. This de-activation code has RPROC and RMEM. RPROC is for reducing processor resources, and RMEM is for reducing memory resources.

3. Enter this de-activation code in the corresponding servers through the HMC, as shown in Figure 6-23 (shows the menu for Enter CoD Code).



Figure 6-23 Menu for Enter CoD Code

4. After entering the de-activation code, you must send a listing of the updated Vital Product Data (VPD) output to the CoD Project office at pcod@us.ibm.com.

Collect the VPD by using the HMC command line, as shown in Example 6-1.

Example 6-1 Collecting the VPD information case

Collect the VPD using the HMC command line instruction for every server:

Processor:`lscod -m your_system_name -t code -r proc -c mobile`

Memory:`lscod -m your_system_name -t code -r mem -c mobile`

5. With the receipt of the `lscod` profile, the Project Office updates the CoD database records and closes out your request.

For more information about how to use the configuration XML file to create Power Enterprise Pool and some management concept, see *Power Enterprise Pools on IBM Power Systems*, REDP-5101.

Configuring redundant HMCs or add EP's master and backup HMC

Section 6.12, “Hardware Management Console high availability introduction” on page 226 introduces HMC high availability design in PowerHA SystemMirror. For the ROHA solution, the HMC is critical, so configuring redundant HMCs is advised.

If there is a Power Enterprise Pool that is configured, configure a backup HMC for Enterprise Pool and add both of them into PowerHA SystemMirror by running the `c1mgr add hmc <hmc>` command or through the SMIT menu. Thus, PowerHA SystemMirror can provide the failover function if the master HMC fails. Section 6.12.1, “Switching to the backup HMC for the Power Enterprise Pool” on page 228 introduces some prerequisites when you set up the Power Enterprise Pool.

Note: At the time of writing, Power Systems Firmware supports a pair of HMCs to manage one Power Enterprise Pool: One is in master mode, and the other one is in backup mode.

Note: At the time of writing, for one Power Systems server, IBM only supports at most two HMCs to manage it.

Verifying the communication between the Enterprise Pool HMC IP and AIX LPARs

If you want PowerHA SystemMirror to control the Power Systems Enterprise Pool mobile resource for RG automatically, you must be able to ping the HMC's host name from an AIX environment. For example, in our testing environment, the master HMC and backup HMC of Power Enterprise Pool is e16hmc1 and e16hmc3. You can get the information by using the **clmgr view report roha** command in AIX or by using the **lscodpool** command in the HMC command line, as shown in Example 6-2 and Example 6-3.

Example 6-2 Show HMC information with the clmgr view report ROHA through AIX

```
...
Enterprise pool 'DEC_2CEC'
    State: 'In compliance'
    Master HMC: 'e16hmc1' --> Master HMC name of EPCoD
    Backup HMC: 'e16hmc3' --> Backup HMC name of EPCoD
    Enterprise pool memory
        Activated memory: '100' GB
        Available memory: '100' GB
        Unreturned memory: '0' GB
    Enterprise pool processor
        Activated CPU(s): '4'
        Available CPU(s): '4'
        Unreturned CPU(s): '0'
    Used by: 'rar1m3-9117-MMD-1016AAP'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
```

Example 6-3 Show EPCoD HMC information with lscodpool through the HMC

```
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_mtms=7042-CR5*06K0036,mobile_procs=
4,avail_mobile_procs=1,unreturned_mobile_procs=0,mobile_mem=102400,avail_mobile_me
m=60416,unreturned_mobile_mem=0
```

Before PowerHA SystemMirror acquires the resource from EPCoD or releases the resource back to EPCoD, PowerHA tries to check whether the HMC is accessible by using the **ping** command. So, AIX must be able to perform the resolution between the IP address and the host name. You can use /etc/hosts, the DNS, or other technology to achieve resolution. For example, on AIX, run **ping e16hmc1** and **ping e16hmc3** to check whether the resolution works.

If the HMCs are in the DNS configuration, configure these HMCs into PowerHA SystemMirror by using their names, and not their IPs.

Entering the On/Off CoD code before using the resource

If you purchased the On/Off CoD code and want to use it with PowerHA SystemMirror, you must enter the code to activate it before you use it. The menu is shown in Figure 6-23 on page 165.

No restrictions for the deployment combination with Enterprise Pool

In one PowerHA SystemMirror cluster, there is no restriction for its nodes deployment with EPCoD:

- ▶ It supports all the nodes in one server and shares mobile resources from one EPCoD.
- ▶ It supports the nodes in different servers and shares one EPCoD.
- ▶ It supports the nodes in different servers and in different EPCoDs.
- ▶ It supports the nodes in different servers and some of them in EPCoD, and others has no EPCoD.

No restrictions for the LPAR CPU type combination in one cluster

One PowerHA SystemMirror cluster supports the following combinations:

- ▶ All nodes are in dedicated processor mode.
- ▶ All nodes are in shared processor mode.
- ▶ Some of the processors are dedicated processor mode and others are shared.

In Figure 6-24, before the application starts, PowerHA SystemMirror checks the current LPAR processor mode. If it is dedicated, then two available CPUs are its target. If it is shared mode, then 1.5 available CPUs and three available VPs are its target.

Add Hardware Resource Provisioning to an Application Controller	
Type or select values in entry fields. Press Enter AFTER making all desired changes.	
* Application Controller Name	[Entry Fields] AppController1
Use desired level from the LPAR profile Optimal number of gigabytes of memory	No + [30]
Optimal number of dedicated processors	[2] #
Optimal number of processing units	[1.5]
Optimal number of virtual processors	[3]

Figure 6-24 Mixed CPU type in one PowerHA SystemMirror cluster

Preferred practice after changing a partition's LPAR name

If you change one partition's LPAR name, the profile is changed, but AIX does not recognize this change automatically. You must shut down the partition and activate it with its profile (AIX IPL process), then after restart, the LPAR name information can be changed.

PowerHA SystemMirror gets the LPAR name from the `uname -L` command's output and uses this name to do DLPAR operations through the HMC. LPAR names of the LPAR hosting cluster node are collected and persisted into HACMPdynresop so that this information is always available.

Note: There is one enhancement to support a DLPAR name update for AIX commands such as **uname -L** or **lparstat -i**. The requirements are as follows:

- ▶ Hardware firmware level SC840 or later (for E870 and E880)
- ▶ AIX 7.1 TL4 or 7.2 or later
- ▶ HMC V8 R8.4.0 (PTF MH01559) with mandatory interim fix (PTF MH01560)

Building password-less communication from the AIX nodes to the HMCs

In order for LPARs to communicate with the HMC, you must use SSH. All the LPAR nodes must have SSH set up.

Setting up SSH for password-less communication with the HMC requires that the user run **ssh-keygen** on each LPAR node to generate a public and private key pair. The public key must then be copied to the HMC's public authorized keys file. Then, the ssh from the LPAR can contact the HMC without you needing to type in a password. Example 6-4 shows an example to set up HMC password-less communication.

Example 6-4 Setting up the HMC password-less communication

```
# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (//.ssh/id_rsa):
Created directory ' //.ssh'.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in // .ssh / id_rsa .
Your public key has been saved in // .ssh / id_rsa . pub .
The key fingerprint is:
70:0d:22:c0:28:e9:71:64:81:0f:79:52:53:5a:52:06 root@epvioc3
The key's randomart image is:
...
# cd /.ssh/
# ls
id_rsa      id_rsa.pub
# export MYKEY=`cat /.ssh/id_rsa.pub` 
# ssh hscroot@172.16.15.42 mkauthkeys -a \"$MYKEY\" 
The authenticity of host '172.16.15.42 (172.16.15.42)' can't be established.
RSA key fingerprint is b1:47:c8:ef:f1:82:84:cd:33:c2:57:a1:a0:b2:14:f0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '172.16.15.42' (RSA) to the list of known hosts.
```

Keeping synchronization turned OFF for the Sync current configuration Capability setting

With later versions of the HMC, the administrator can enable the synchronization of the current configuration to the profile. If you use ROHA, the CPU and memory setting in the LPAR's profile is modified with each operation. For consistency, turn off this synchronization when ROHA is enabled (Figure 6-25).

Partition Properties - ITSO_S1Node1

General	Hardware	Virtual Adapters	SR-IOV Logical Ports	Settings	Other
Name:	* ITSO_S1Node1				
ID:	4				
Environment:	AIX or Linux				
State:	Running				
Attention LED:	Off				
Resource configuration:	Configured				
OS version:	AIX 7.1 7100-04-00-0000				
Current profile:	ITSO_profile				
System:	9117-MMD*1016AAP				
<input type="checkbox"/> Allow performance information collection					
<input type="checkbox"/> Allow this partition to be suspended.					
<input type="checkbox"/> Virtual Trusted Platform Module (VT TPM)					
<i>Warning: VT TPM Trusted Key is the default key.</i>					
Sync current configuration Capability <input type="button" value="Sync turned OFF"/>					
OK	Cancel	Help			

Figure 6-25 Sync current configuration capability

Setting the LPAR's minimum and maximum parameters

When you configure an LPAR on the HMC (outside of PowerHA SystemMirror), you provide LPAR minimum and LPAR maximum values for the number of CPUs and amount of memory.

The stated minimum values of the resources must be available when an LPAR node starts. If more resources are available in the free pool on the frame, an LPAR can allocate up to the stated wanted values. During dynamic allocation operations, the system does not allow the values for CPU and memory to go below the minimum or above the maximum amounts that are specified for the LPAR.

PowerHA SystemMirror obtains the LPAR minimum and LPAR maximum amounts and uses them to allocate and release CPU and memory when application controllers are started and stopped on the LPAR node.

In the planning stage, you must consider how many resources are needed to satisfy all the RGs online carefully and set LPAR's minimal and maximum parameters correctly.

Using pre-event and post-event scripts

Existing pre-event and post-event scripts that you might be using in a cluster with LPARs (before using the CoD integration with PowerHA SystemMirror) might need to be modified or rewritten if you plan to configure CoD and DLPAR requirements in PowerHA SystemMirror.

Keep in mind the following considerations:

- ▶ PowerHA SystemMirror performs all the DLPAR operations before the application controllers are started and after they are stopped. You might need to rewrite the scripts to account for this situation.
- ▶ Because PowerHA SystemMirror takes care of the resource calculations, and requests additional resources from the DLPAR operations and, if allowed, from CUoD, you can dispose the portions of your scripts that perform those functions.
- ▶ PowerHA SystemMirror considers the On/Off CoD possibility, for example, even though the cluster is configured in a single frame. If your cluster is configured within one frame, then modifying the scripts as stated before is sufficient.
- ▶ However, if a cluster is configured with LPAR nodes that are on two frames, you might still require the portions of the existing pre-event and post-event scripts that deal with dynamically allocating resources from the free pool on one frame to the node on another frame, if the application requires these resources.

Note: When you deal with EPCoD or On/Off CoD resources, it does not matter if there is one or two frames. For case scenarios with EPCoD or On/Off CoD, you activate (for On/Off) and acquire (for EPCoD), and modify the portion of code that deals with On/Off activation and EPCoD acquisition.

About the elapsed time of the DLPAR operation

When you plan a PowerHA SystemMirror cluster with the ROHA feature, you must consider the DLPAR release time.

While initially bringing the RG online, PowerHA SystemMirror must wait for all the resources acquisition to complete before it can start the user's application.

While performing a takeover (failover to the next priority node, for example), PowerHA SystemMirror tries to perform some operations (DLPAR or adjust CoD and EPCoD resource) in parallel to the release of resources on the source node and the acquisition of resources on target node if the user allows it in the tunables (the value of Force synchronous release of DLPAR resources is No).

Table 6-15 on page 171 shows the testing results of the DLPAR operation. The result might be different in other environments, particularly if the resource is being used.

There is one LPAR, its current running CPU resource size is 2C, and the running memory resource size is 8 GB. The DLPAR operation includes add and remove.

Table 6-15 Elapsed time of the DLPAR operation

Incremental value By DLPAR	Add CPU (in seconds)	Add memory (in seconds)	Remove CPU (in seconds)	Remove memory (in minutes and seconds)
2C and 8 GB	5.5 s	8 s	6 s	88 s (1 m 28 s)
4C and 16 GB	7 s	12 s	9.8 s	149 s (2 m 29 s)
8C and 32 GB	13 s	27 s	23 s	275 s (4 m 35 s)
16C and 64 GB	18 s	34 s	33 s	526 s (8 m 46 s)
32C and 128 GB	24 s	75 s	52 s	1010 s (16 m 50 s)
48C and 192 GB	41 s	179 s	87 s	1480 s (24 m 40 s)

AIX ProbeVue maximum pinned memory setting

ProbeVue is a dynamic tracing facility of AIX. You can use it for both performance analysis and problem debugging. ProbeVue uses the Vue programming language to dynamically specify trace points and provide the actions to run at the specified trace points. This feature is enabled by default. There is one restriction regarding ProbeVue's maximum pinned memory and DLPAR remove memory operation: The Max Pinned Memory For ProbeVue tunable *cannot* cross the 40% limit of system running memory.

For example, you configure one profile for an LPAR with 8 GB (minimum) and 40 GB (wanted). When you activate this LPAR, the maximum pinned memory of ProbeVue is set to 4 GB (10% of system running memory), as shown in Example 6-5.

From AIX 7.1 TL4 onward, the tunables are derived based on the available system memory. MAX pinned memory is set to 10% of the system memory. It cannot be adjusted when you restart the operating system or adjust the memory size with the DLPAR operation.

Example 6-5 Current maximum pinned memory for ProbeVue

```
# probevctrl -l
Probevue Features: on --> ProbeVue is enabled at this time
MAX pinned memory for Probevue framework(in MB): 4096 --> this is the value we are
discussing
...
```

Now, if you want to reduce the memory 40 - 8 GB, run the following command:

```
chhwres -r mem -m r1r9m1-9117-MMD-1038B9P -o r -p ITSO_S2Node1 -q 32768
```

The command fails with the error that is shown in Example 6-6.

Example 6-6 Error information when you reduce the memory through the DLPAR

```
hscroot@e16hmc3:~> chhwres -r mem -m r1r9m1-9117-MMD-1038B9P -o r -p ITSO_S2Node1
-q 32768
HSCL2932 The dynamic removal of memory resources failed: The operating system
prevented all of the requested memory from being removed. Amount of memory
removed: 0 MB of 32768 MB. The detailed output of the OS operation follows:
```

0930-050 The following kernel errors occurred during the
DLPAR operation.

0930-023 The DR operation could not be supported by one or more kernel extensions.

Consult the system error log for more information

....

Please issue the lshwres command to list the memory resources of the partition and to determine whether it is pending and runtime memory values match. If they do not match, problems with future memory-related operations on the managed system might occur, and it is recommended that the rsthwres command to restore memory resources be issued on the partition to synchronize its pending memory value with its runtime memory value.

From AIX, the error report also generates some error information, as shown in Example 6-7 and Example 6-8.

Example 6-7 AIX error information when you reduce the memory through the DLPAR

47DCD753	1109140415 T S PROBEVUE	DR: memory remove failed by ProbeVue rec
252D3145	1109140415 T S mem	DR failed by reconfig handler

Example 6-8 Detailed information about the DR_PVUE_MEM_Rem_ERR error

LABEL:	DR_PVUE_MEM_Rem_ERR
IDENTIFIER:	47DCD753

Date/Time: Mon Nov 9 14:04:56 CST 2015
Sequence Number: 676
Machine Id: 00F638B94C00
Node Id: ITSO_S2Node1
Class: S
Type: TEMP
WPAR: Global
Resource Name: PROBEVUE

Description

DR: memory remove failed by ProbeVue reconfig handler

Probable Causes

Exceeded one or more ProbeVue Configuration Limits or other

Failure Causes

Max Pinned Memory For Probevue tunable would cross 40% limit

Recommended Actions

Reduce the Max Pinned Memory For Probevue tunable

Detail Data

DR Phase Name

PRE

Current System Physical Memory

42949672960 --> This is 40 GB, which is the current running memory size.

Memory that is requested to remove

34359738368 --> This is 32 GB, which you want to remove.

ProbeVue Max Pinned Memory tunable value

4294967296 --> This is 4 GB, which is current maximum pinned memory for ProbeVue.

In the ROHA solution, it is possible that PowerHA SystemMirror removes memory to a low value, such as in the procedure of Automatic resource release after OS failure. To avoid this situation, consider the following items:

- ▶ If you want to enable the ProbeVue component, set the maximum pinned memory less or equal to (40% *minimum memory value of one LPAR's profile). For example, in this case, the minimum memory size is 8 GB, so 40% is 3276.8 MB.

Therefore, you can set the maximum pinned memory size with the command that is shown in Table 6-9.

Example 6-9 Change max_total_mem_size

```
# probevctrl -c max_total_mem_size=3276
```

Attention: The command "/usr/sbin/bosboot -a" must be run for the change to take effect in the next boot.

Set it to 3276 MB, which is less than 3276.8 (8 GB*40%). This change takes effect immediately. But if you want this change to take effect after the next start, you need to run **/usr/sbin/bosboot -a** before the restart.

- ▶ If you do not want the ProbeVue component online, you can turn it off with the command that is shown in Example 6-10.

Example 6-10 Turn off ProbeVue

```
# probevctrl -c trace=off
```

Attention: The command "/usr/sbin/bosboot -a" must be run for the change to take effect in the next boot.

This change takes effect immediately. But if you want this change to take effect after the next start, you need to run **/usr/sbin/bosboot -a** before the restart.

6.4.2 Configuration steps for Resource Optimized High Availability

After finishing all of the preparations and considerations outside of PowerHA SystemMirror, you must configure PowerHA SystemMirror.

First, you must configure some generic elements for the PowerHA SystemMirror cluster:

- ▶ Cluster name
- ▶ Nodes in the cluster
- ▶ CAA repository disk
- ▶ Shared VG
- ▶ Application controller
- ▶ Service IP
- ▶ RG
- ▶ Other user-defined contents, such as pre-event or post-event

Then, you can configure the ROHA-related elements:

- ▶ HMC configuration (see 6.2.3, "HMC configuration" on page 148):
 - At least one HMC. Two HMCs are better.
 - Optionally, change the cluster HMC tunables.
 - Optionally, change the HMC at the node or site level.
- ▶ Optimal resources for each application controller (see 6.2.4, "Hardware resource provisioning for application controller" on page 155).

- ▶ Optionally, change the cluster ROHA tunables (see 6.2.5, “Change/Show Default Cluster Tunable” on page 160).
- ▶ Run Verify and Synchronize, review the warning or error messages, and fix them.
- ▶ Show the ROHA report by running the `clmgr view report roha` command and review the output.

6.5 Resource acquisition and release process introduction

This section introduces the steps of the resource acquisition and release in a ROHA solution.

6.5.1 Steps for allocation and for release

Figure 6-26 shows the steps of allocation and release. During failover, resources are released on the active node (same as stopping RGs, which are the red numbers in the diagram) and resources are acquired on the backup node (same as starting RGs, which are the green numbers in the diagram). Figure 6-26 shows the process when CoD Pool and Enterprise Pool are used. On some CECs, none or only one or both of those reservoirs can be used.

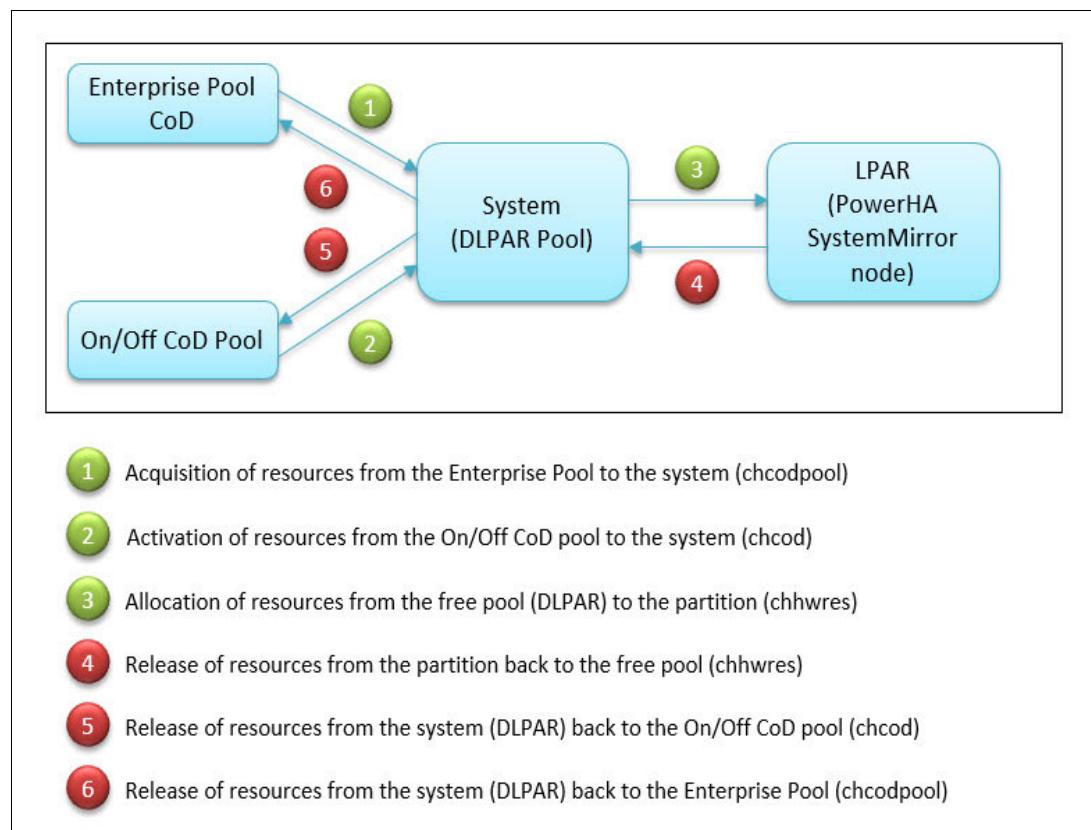


Figure 6-26 Allocations and releases steps

For the resource releasing process, in some cases, PowerHA SystemMirror tries to return EPCoD resources before doing the DLPAR remove operation from the LPAR, and this generates *unreturned* resource on this server. This is an asynchronous process and is helpful to speed up RG takeover. The unreturned resource is reclaimed after the DLPAR remove operation is completed.

6.6 Introduction to resource acquisition

Figure 6-27 shows the process of acquisition for memory and processor. Resources are acquired together for a list of applications. It is a four-step process:

1. Query (blue boxes): The required resources are computed based on the LPAR configuration, and the information that is provided by the PowerHA SystemMirror state (if applications are currently running) and applications. Then, the script contacts the HMC to get information about available ROHA resources.
2. Compute (purple box): Based on this information, PowerHA SystemMirror determines the total amount of required resources that are needed on the node for the list of RGs that are to be started on the node.
3. Identify (green box): PowerHA SystemMirror determines how to perform this allocation for the node by looking at each kind of allocation to be made, that is, which part must come from Enterprise Pool CoD resources, and which part must come from On/Off CoD resources to provide some supplementary resources to the central electronic complex, and which amount of resources must be allocated from the central electronic complex to the LPAR through a DLPAR operation.
4. Apply (orange boxes): After these decisions are made, the script contacts the HMC to acquire resources. First, it acquires Enterprise Pool CoD resources and activates On/Off CoD resources, and then allocates all DLPAR resources. The amount of DLPAR resources that are allocated is persisted in the HACMPdynresop ODM object for release purposes.

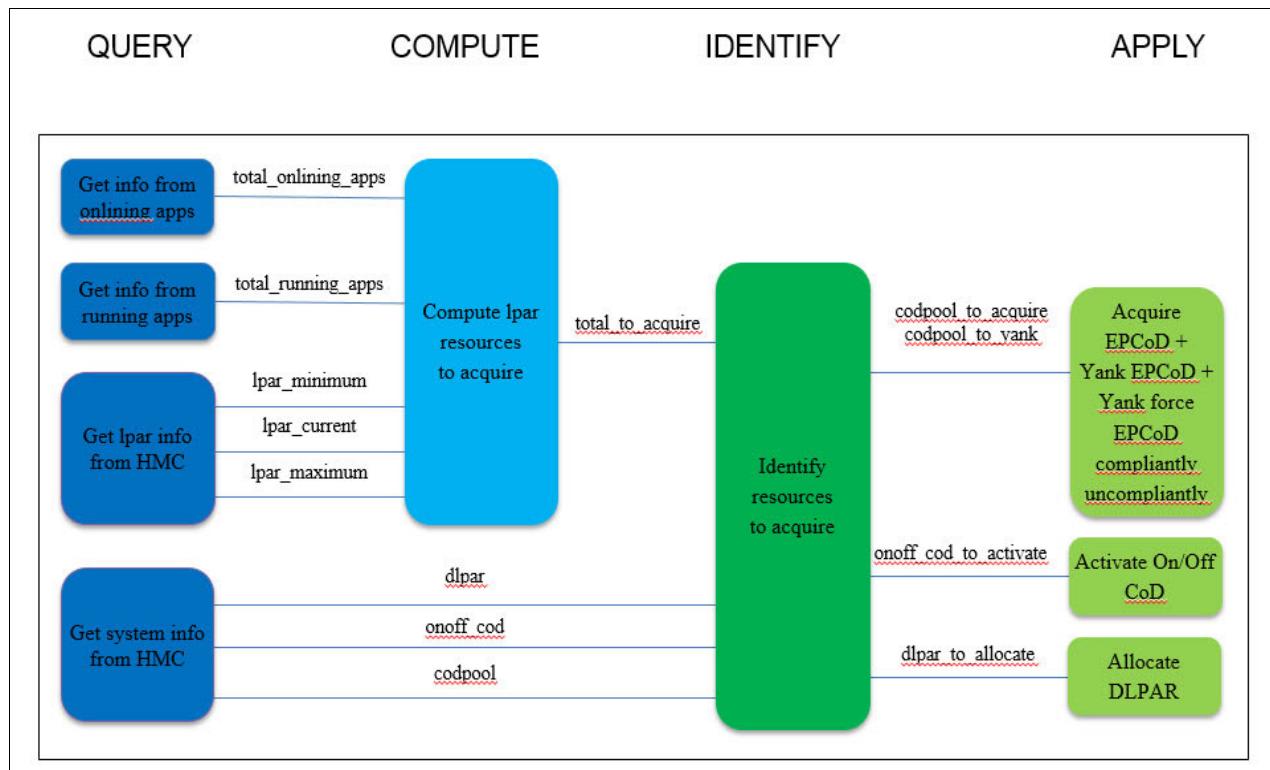


Figure 6-27 Four steps to acquire resources

There are many reasons for success. The script immediately returns whether the applications are not configured with optimal resources. The script also exits if there are already enough resources allocated. Finally, the script exits when the entire process of acquisition succeeds.

However, the script can fail and return an error if one of the following situations occurs:

- ▶ The maximum LPAR size as indicated in the LPAR profile is exceeded and the Always Start RGs tunable is set to No.
- ▶ The shared processor pool size is exceeded and the Adjust SPP size if the required tunable is set to No.
- ▶ There are not enough free resources on the central electronic complex, or on the Enterprise Pool CoD, or on the On/Off CoD, and the Always Start RGs tunable is set to No.
- ▶ Any one step of the acquisition fails (see the previous four steps). Thus, successful actions that were previously performed are rolled back, and the node is reset to its initial allocation state.

In a shared processor partition, more operations must be done. For example, account for both virtual CPUs and processing units instead of only a number of processors. To activate On/Off CoD resources or acquire Enterprise Pool CoD resources, decimal processing units are converted to integers and decimal gigabytes of memory should be converted to integers.

On shared processor pool partitions, the maximum pool size can be automatically adjusted, if necessary and if authorized by the user.

6.6.1 Query

In the query step, PowerHA SystemMirror gets the information that is listed in the following sections.

Getting information from onlining apps

Onlining applications see the applications being brought online. The process is achieved by summing values that are returned by an ODM request to the HACMPserver object containing the applications resources provisioning.

Getting information from running apps

Running applications see the applications currently running on the node. The process is achieved by calling the **c1RGinfo** command to obtain all the running applications and summing values that are returned by an ODM request on all those applications.

Getting LPAR information from HMC

The minimum, maximum, and currently allocated resources for the partition are listed through the HMC command **1shwres**.

Getting the DLPAR resource information from HMC

Some people think only the available resources (the query method is shown in Table 6-16) can be used for the DLPAR operation.

Table 6-16 Get server's available resource from HMC

Memory	<code>1shwres -m <cec> --level sys -r mem -F curr_avail_sys_mem</code>
CPU	<code>1shwres -m <cec> --level sys -r proc -F curr_avail_sys_proc_units</code>

Tip: The **1shwres** commands are given in Table 6-16 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Strictly speaking, this situation is *not* correct. Two kinds of cases must be considered:

- ▶ There are stopped partitions on the central electronic complex.

A stopped partition still keep its resources because the resources do not appear as Available in the central electronic complex. As a matter of fact, the resources are available for other LPARs. Therefore, if you stopped a partition on the central electronic complex, the resource that stopped the partition must be available for the DLPAR operation.

Figure 6-28 shows that there is no available CPU resource when you run the `lshwres` command. But in fact, there is 0.5 CPU, which LPAR `rar1m34` is holding, and this LPAR is in the Not Activated state. The free CPU resources are 0.5 CPU (0+0.5).

The screenshot shows two windows side-by-side. On the left is a table of partitions under 'Systems Management > Servers > rar1m3-9117-MMD-1016AAP'. The columns are Select, Name, ID, Status, and Processing Units. Partitions listed are: rar1m3v1 (Running, 1), sarnoth (Not Activated, 0), piehole (Running, 1), ITO_S1Node1 (Running, 1.5, selected), ITO_rar1m3_Node2 (Not Activated, 0), rar1m33 (Not Activated, 0), and rar1m34 (Not Activated, 0.5). A pink arrow points from the 'Processing Units' column of rar1m34 to the 'Available system processing units' field in the dialog on the right. On the right is a 'Add or Remove Processor Resources' dialog for 'ITSO_S1Node1'. It shows fields for Available system processing units (0.0), Available system processing units (with releasable amount from other partitions) (0.5), Minimum (0.5), Assigned (1.5), and Virtual processors (1). Below these are command-line options: `lshwres -m <cec> --level sys -r mem -F curr_avail_sys_mem` and an 'Uncapped Weight' field set to 0. The dialog also includes an 'Options' section with Timeout (minutes) set to 5 and Detail level set to 1. Buttons at the bottom include OK, Cancel, and Help.

Figure 6-28 Describing the difference between available resources and free resources

- ▶ There are uncapped mode partitions in the central electronic complex.

In an uncapped shared processor partition, considering only the maximum processor unit is not correct.

Consider the following case, where one LPAR's profile includes the following configuration:

- Minimum processor unit: 0.5
- Assigned processor unit: 1.5
- Maximum processor unit: 3
- Minimum virtual processor: 1
- Assigned virtual processor: 6
- Maximum virtual processor: 8

This LPAR can acquire six processor units if the workload increases, and if these resources are available in the central electronic complex. Also, this value is above the limit that is set by the Maximum processor unit, which has a value of 3.

But in any case, allocation beyond the limit of the maximum processor unit is something that is performed at the central electronic complex level, and cannot be controlled at the PowerHA SystemMirror level.

But it is true that the calculation of available resources could consider what is really being used in the CEC, and should not consider the Maximum processor unit as an intangible maximum. The real maximum comes from the number of Assigned Virtual Processor.

PowerHA SystemMirror supports the *uncapped mode*, but does not play a direct role in this support because this mode is used at the central electronic complex level. There is no difference in uncapped mode compared with the capped mode for PowerHA SystemMirror.

Based on the previous considerations, the formula to calculate free resources (memory and processor) for the DLAR operation is shown in Figure 6-29.

$free_mem = configurable_sys_mem - sys_firmware_mem - \sum_{lpars}^{activated} curr_mem - \sum_{lpars}^{shutdowned} run_mem$
$free_proc = configurable_{sysproc_units} - \sum_{lpars}^{activated} curr_{proc_units} - \sum_{lpars}^{shutdowned} run_{proc} - \sum_{spp_pools}^{used} reserved$

Figure 6-29 Formula to calculate free resources of one central electronic complex

Note: You read the level of *configured* resources (`configurable_sys_mem` in the formula), and you remove from that the level of *reserved* resources (`sys_firmware_mem` in the formula), then you end up with the level of resources that is needed to run one started partition.

Moreover, when computing the free processing units of a CEC, you consider the *reserved processing units* of any used Shared Processor Pool (the `reserved` in the formula).

Getting On/Off CoD resource information from the HMC

The available On/Off CoD resources for the CEC is listed through the HMC `lscod` command. The state is Available or Running (a request is ongoing). Table 6-17 shows the commands that PowerHA SystemMirror uses to get On/Off resource information. You do not need to run these commands.

Table 6-17 Get On/Off CoD resources' status from HMC

Memory	<code>lscod -m <cec> -t cap -c onoff -r mem -F mem_onoff_state:avail_mem_for_onoff</code>
CPU	<code>lscod -m <cec> -t cap -c onoff -r proc -F proc_onoff_state:avail_proc_for_onoff</code>

Tip: The `lscod` commands are given in Table 6-17 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Acquiring Power Enterprise Pool resource information from the HMC

The available Enterprise Pool CoD resources for the pool can be acquired by running the HMC **lscodpool** command. Table 6-18 shows the commands that PowerHA SystemMirror uses to get the EPCoD information. You do not need to run these commands.

Table 6-18 Get the EPCoD available resources from the HMC

Memory	<code>lscodpool -p <pool> --level pool -F avail_mobile_mem</code>
CPU	<code>lscodpool -p <pool> --level pool -F avail_mobile_procs</code>

Tip: The **lscodpool** commands are given in Table 6-18 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Note: If the execution of this command fails (either because the link is down or other errors), after the last retry but before trying another HMC, PowerHA SystemMirror changes the master HMC for its Enterprise Pool.

6.6.2 Resource computation

After the query step, PowerHA SystemMirror starts performing computations to satisfy the PowerHA SystemMirror application controller's needs. It is likely that some resources must be allocated from the CEC to the LPAR. Figure 6-30 shows the computation of the amount of resources to be allocated to the partition. This computation is performed for all types of resources, and it accounts for the following items:

- ▶ The configuration of the partition (minimum, current, and maximum amount of resources)
- ▶ The optimal resources that are configured for the applications currently running on the partition
- ▶ The optimal resources that are configured for the applications that are being brought online

In Figure 6-30, case 2b is the normal case. The currently allocated resources level matches the blue level, which is the level of resources for the application controllers currently running. PowerHA SystemMirror adds the yellow amount to the blue amount.

But in some cases, where these two levels do not match, consider having a “start fresh” policy. This policy performs a readjustment of the allocation to the exact needs of the currently running application controllers that are added to the application controllers that are being brought online (always provides an optimal amount of resources to application controllers). Those alternative cases can occur when the user has manually released (case 2a) or acquired (case 2c) resources.

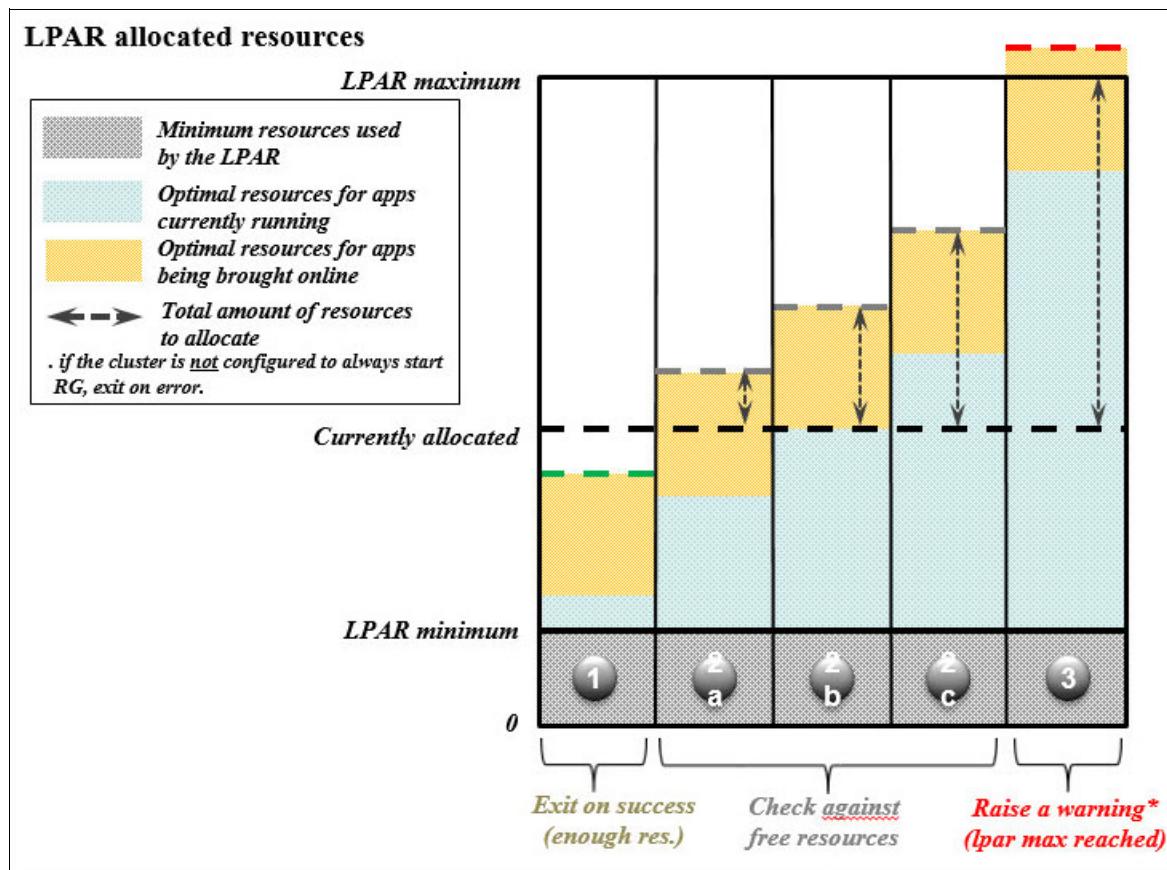


Figure 6-30 Computation policy in the resource acquisition process

Here is information about the cases:

- ▶ In case 1, PowerHA SystemMirror keeps the allocated level current to satisfy your needs. This case can occur when a partition is at its profile's wanted level, which is greater than its profile's minimum.
- ▶ In case 2a, the readjustment consists of allocating only the missing part of the application controllers that are being brought online.
- ▶ In case 2c, the readjustment consists of allocating the missing part of the application controllers currently running that is added to the application controllers that are being brought online.
- ▶ In case 3, the needed resources cannot be satisfied by this partition. It exceeds the partition profile's maximum. In this particular case, two behaviors can happen here depending on the Always Start RGs tunable. If enabled, PowerHA SystemMirror tries to allocate all that can be allocated, raises a warning, and continues. If disabled, PowerHA SystemMirror stops and returns an error.

In shared processor partitions, both virtual CPUs and processing units are computed. In shared processor partitions that are part of a Shared Processor Pool, the need for computation is checked against the PU/VP ratio and adjusted as needed. If it is less than need, everything is fine and the process continues. If it is greater than needed, set the Adjust SPP size if required tunable to No. The process stops and returns an error. Otherwise, it raises a warning, changes the pools size to the new size, and goes on.

6.6.3 Identifying the method of resource allocation

In the resource compute step, the amount of resources that are needed by the LPAR is computed, so now you must identify how to achieve the wanted amount. PowerHA SystemMirror considers multiple strategies in the following order:

1. Consider the CEC current free pool for DLPAR operations. This section explains how these available resources are computed.
2. If resources are still insufficient, consider the Enterprise Pool of resources, first by considering the available amount of Enterprise Pool in the local frame, then by considering the amount of Enterprise Pool that is acquired by the other frame (and see whether it can be returned back to be available on the local frame), then by considering the amount of Enterprise Pool of all frames sharing this Enterprise Pool (and see whether they can be returned back compliantly or uncomppliantly to be available on the local frame).
3. If resources are still insufficient, consider the CoD pool of resources if a license was activated, and if any On/Off CoD resources are available.

When the correct strategy is chosen, there are three types of resource allocations to be done:

1. Release on other CECs: You might need to release EPCoD resources on other CEC so that these resources are made available on the local CEC.
2. Acquisition/Activation to the CEC: Resources can come from the Enterprise Pool CoD or the On/Off CoD pools.
3. Allocation to the partition: Resources come from the CEC to the LPAR.

Figure 6-31 shows the computation for the DLPAR, CoD, and Enterprise Pool CoD for the amount of resources to acquire. The computation is performed for all types of resources. In shared processor partitions, only processing units are computed this way, and accounts for the following items:

- ▶ The total amount of resources to acquire for the node (computed previously).
- ▶ The available amount of DLPAR resources on the CEC.
- ▶ The available amount of On/Off CoD resources on the CEC.
- ▶ The available amount of Enterprise Pool CoD resources in the pool to which the CEC belongs.
- ▶ The amount of Enterprise Pool CoD resources that are acquired on other frames, and that it is possible to return back.

Figure 6-31 shows the identified policy in the resource acquisition process.

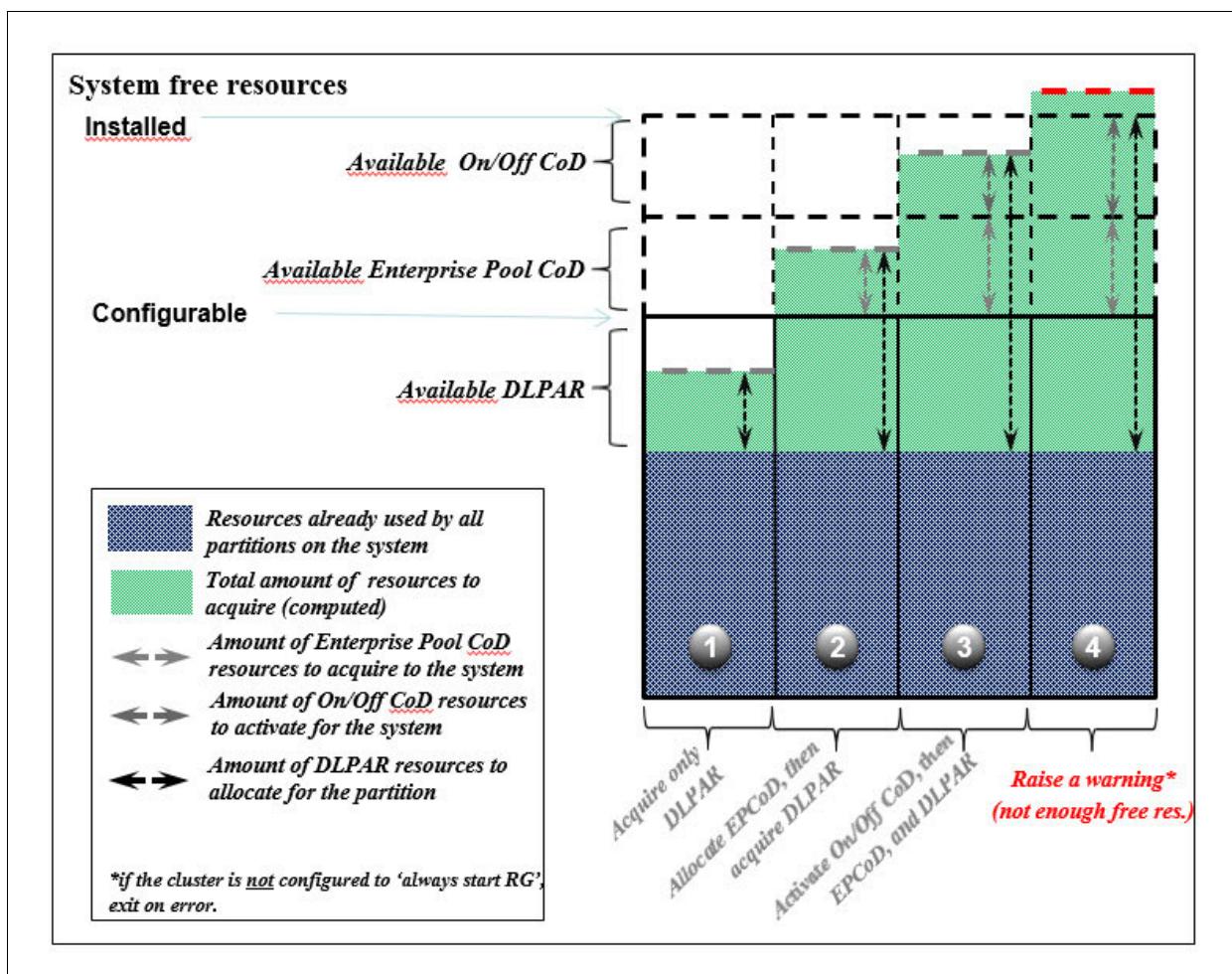


Figure 6-31 Identify the policy in the resource acquisition process

There are four possible cases:

1. There are sufficient DLPAR resources to fulfill the optimal configuration. No EPCoD resources or On/Off CoD resources will be allocated to the CEC. A portion of the available DLPAR resources will be allocated to the node.
2. A portion of available Enterprise Pool CoD resources will be allocated to the CEC, and then all DLPAR resources will be allocated. No On/Off CoD resources will be activated.
Alternative case: If there are no available EPCoD resources, a portion of available On/Off CoD resources will be activated instead, and then all DLPAR resources will be allocated.
3. All available Enterprise Pool CoD resources will be allocated to the CEC, then a portion of On/Off CoD resources will be activated, and then all DLPAR resources will be allocated.
Alternative case: If there are no available EPCoD resources, a portion of available On/Off CoD resources will be activated instead, and then all DLPAR resources will be allocated (as in case 2).
4. All available Enterprise Pool CoD resources will be allocated to the CEC, then all On/Off CoD resources will be activated, and then all DLPAR resources will be allocated.
Alternative case: If the cluster has not been configured to automatically start the RGs even if resources are insufficient, do not allocate or acquire any resources because this exceeds the available resources for this CEC and exits on error instead.

In shared processor partitions, PowerHA SystemMirror accounts for the minimum ratio of assigned processing units to assigned virtual processors for the partition that is supported by the CEC. In an IBM POWER6® server, the ratio is 0.1 and in an IBM POWER7® server, the ratio is 0.05.

For example, if the current assigned processing unit in the partition is 0.6 and the current assigned virtual processor is 6, and PowerHA SystemMirror acquires virtual processors, it raises an error because it breaks the minimum ratio rule. The same occurs when PowerHA SystemMirror releases the processing units. PowerHA SystemMirror must compare the expected ratio to the configured ratio.

6.6.4 Acquiring the resource

After finishing the steps in 6.6.3, “Identifying the method of resource allocation” on page 181, PowerHA SystemMirror performs the acquire operation.

Acquiring the Power Enterprise Pool resource

The Enterprise Pool CoD resources are allocated by the HMC **chcodpool** command. Table 6-19 shows the commands that PowerHA SystemMirror uses to assign EPCoD resources to one server. You do not need to run these commands.

Table 6-19 Acquiring the EPCoD mobile resources

Memory	<code>chcodpool -p <pool> -m <system> -o add -r mem -q <mb_of_memory></code>
CPU	<code>chcodpool -p <pool> -m <system> -o add -r proc -q <cpu></code>

Tip: The **chcodpool** commands are given in Table 6-19 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Activating the On/Off CoD resources

On/Off CoD resources are activated by the HMC **chcod** command. Table 6-20 shows the commands that PowerHA SystemMirror uses to assign the On/Off CoD resource to one server. You do not need to run these commands.

Table 6-20 Acquire On/Off available resources

Memory	chcod -m <cec> -o a -c onoff -r mem -q <mb_of_memory> -d <days>
CPU	chcod -m <cec> -o a -c onoff -r proc -q <cpu> -d <days>

Tip: The **chcod** commands are given in Table 6-20 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Note: For acquiring the Power Enterprise Pool and the On/Off CoD resources, every amount of memory resources is expressed in MB but aligned in GB of memory (for example, 1024 or 4096), and every number of processing units is aligned on the whole upper integer.

All Power Enterprise Pool and On/Off CoD resources that are acquired are in the CEC's free pool, and these are automatically added to the target LPAR by using DLPAR.

Allocating the DLPAR resources

DLPAR resources are allocated by using the HMC **chhwres** command. Table 6-21 shows the commands that PowerHA SystemMirror uses to assign resources from the server's free pool to one LPAR. You do not need to run these commands.

Table 6-21 Assign resources from the server's free pool to target LPAR

Dedicate Memory	chhwres -m <cec> -p <1par> -o a -r mem -q <mb_of_memory>
Dedicate CPU	chhwres -m <cec> -p <1par> -o a -r proc --procs <cpu>
Shared CPU	chhwres -m <cec> -p <1par> -o a -r proc --procs <vp> --proc_units <pu>

Tip: The **chhwres** commands are given in Table 6-21 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

For shared processor partitions in a Shared-Processors Pool that is not the default pool, it might be necessary to adjust the maximum processing units of the Shared Processor Pool. To do so, use the operation that is shown in Example 6-11, which uses the HMC **chhwres** command. The enablement of this adjustment is authorized or not by a tunable.

Example 6-11 shows the command that PowerHA SystemMirror uses to change the Shared-Processor Pool's maximum processing units. You do not need to run this command.

Example 6-11 DLPAR command line from HMC

```
chhwres -m <cec> -o s -r procpool --poolname <pool> -a max_pool_proc_units=<pu>
```

Tip: The `chhwres` commands are given in Example 6-11 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

6.7 Introduction to release of resources

When the RGs are stopped, PowerHA SystemMirror computes the amount of resources to be released and is responsible for performing the release of ROHA resources. There are four steps when releasing resources:

1. The *query step*, which appears in purple. In this step, PowerHA SystemMirror queries all the information that is needed for the compute, identify, and release steps.
2. The *compute step*, which appears in blue. In this step, PowerHA SystemMirror computes how many resources must be released through DLPAR. In this step, PowerHA SystemMirror uses a “fit to remaining RGs” policy, which consists in computing amounts of resources to be released by accounting for currently allocated resources and total optimal resources that are needed by RGs remaining on the node. In any case, and as it was done before, PowerHA SystemMirror does not release more than optimal resources for the RGs being released.
3. The *identify step*, which appears in green. In this step, PowerHA SystemMirror identifies how many resources must be removed from the LPAR, and identify how many resources must be released to the On/Off CoD and to the Power Enterprise Pool.
4. In the *apply step*, you remove resources from the LPAR and release resources from the CEC to On/Off CoD and Power Enterprise Pool, which appears in light green. In this step, PowerHA SystemMirror performs the DLPAR remove operation and then releases On/Off CoD resources and EPCoD resources. You can release up to the amount, but no more, of the DLPAR resources being released.

These steps are shown in Figure 6-32.

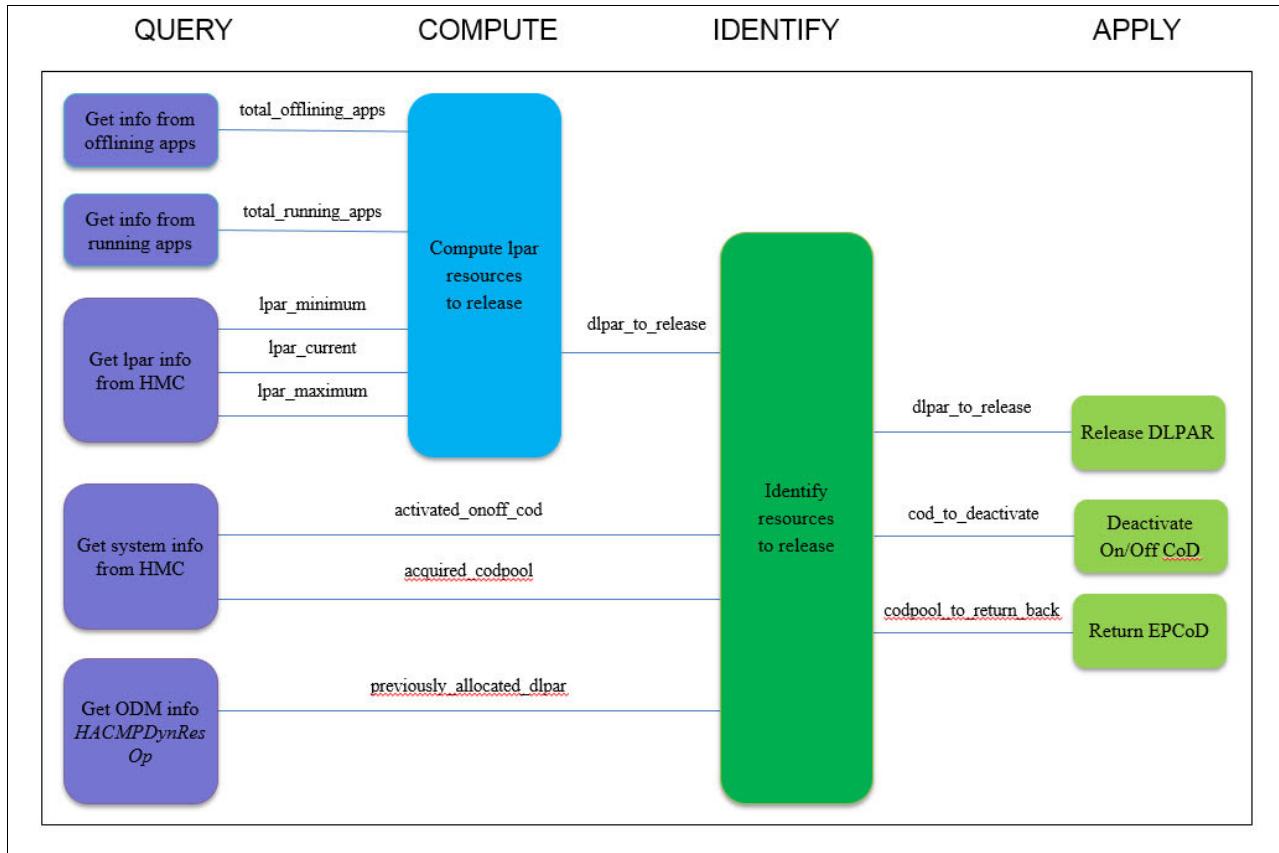


Figure 6-32 Four steps to release resources

6.7.1 Query

In the query step, PowerHA SystemMirror gets the information that is described in the following sections for the compute step.

Getting information from offline apps

Offline applications see the one being brought offline. At this step, check that the release of resources is needed. It means that at least one application is configured with optimal resources.

Getting information from running apps

Running applications see the ones currently running on the node. It is achieved by calling the `c1RGinfo` binary to obtain all the running applications and summing values that are returned by an ODM request on all those applications.

Getting LPAR information from the HMC

The minimum, maximum, and currently allocated resources for the partition are listed by the HMC `1shwres` command.

Get On/Off CoD resource information from the HMC

The *active* On/Off CoD resources for the CEC are listed by the HMC **1scod** command. Table 6-22 shows the commands that PowerHA SystemMirror uses to get On/Off CoD information. You do not need to run these commands.

Table 6-22 Get On/Off active resources in this server from the HMC

Memory	<code>1scod -m <cec> -t cap -c onoff -r mem -F activated_onoff_mem</code>
CPU	<code>1scod -m <cec> -t cap -c onoff -r proc -F activated_onoff_proc</code>

Tip: The **1scod** commands are given in Table 6-22 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Getting Power Enterprise Pool resource information from the HMC

The *allocated* Enterprise Pool CoD resources for the pool are listed by the HMC **1scodpool** command. Table 6-23 shows the commands that PowerHA SystemMirror uses to get EPCoD information. You do not need to run these commands.

Table 6-23 Get EPCoD resource information

Memory	<code>1scodpool -p <pool> --level pool -F mobile_mem</code> <code>1scodpool -p <cec> --level sys --filter "names=server name" -F mobile_mem</code>
CPU	<code>1scodpool -p <pool> --level pool -F mobile_procs</code> <code>1scodpool -p <cec> --level sys --filter "names=server name" -F mobile_procs</code>

Tip: The **1scodpool** commands are given in Table 6-23 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

Resource computation

The level of resources to be left on the LPAR is computed by using the fit to remaining RGs policy. What is above this level is released, and it accounts for the following information:

1. The configuration of the LPAR (minimum, current, and maximum amount of resources).
2. The optimal resources that are configured for the applications currently running on the LPAR. PowerHA SystemMirror tries to fit to the level of remaining RGs running on the node.
3. The optimal amount of resources of the stopping RGs because you do not de-allocate more than this.

Two cases can happen, as shown in Figure 6-33:

1. Release resources to a level that enables the remaining applications to run at optimal level. PowerHA SystemMirror applies the remaining RGs policy here to compute and provide the optimal amount of resources to the remaining applications.
2. Do not release any resources because the level of currently allocated resources is already under the level that is computed by the fit to remaining RGs policy.

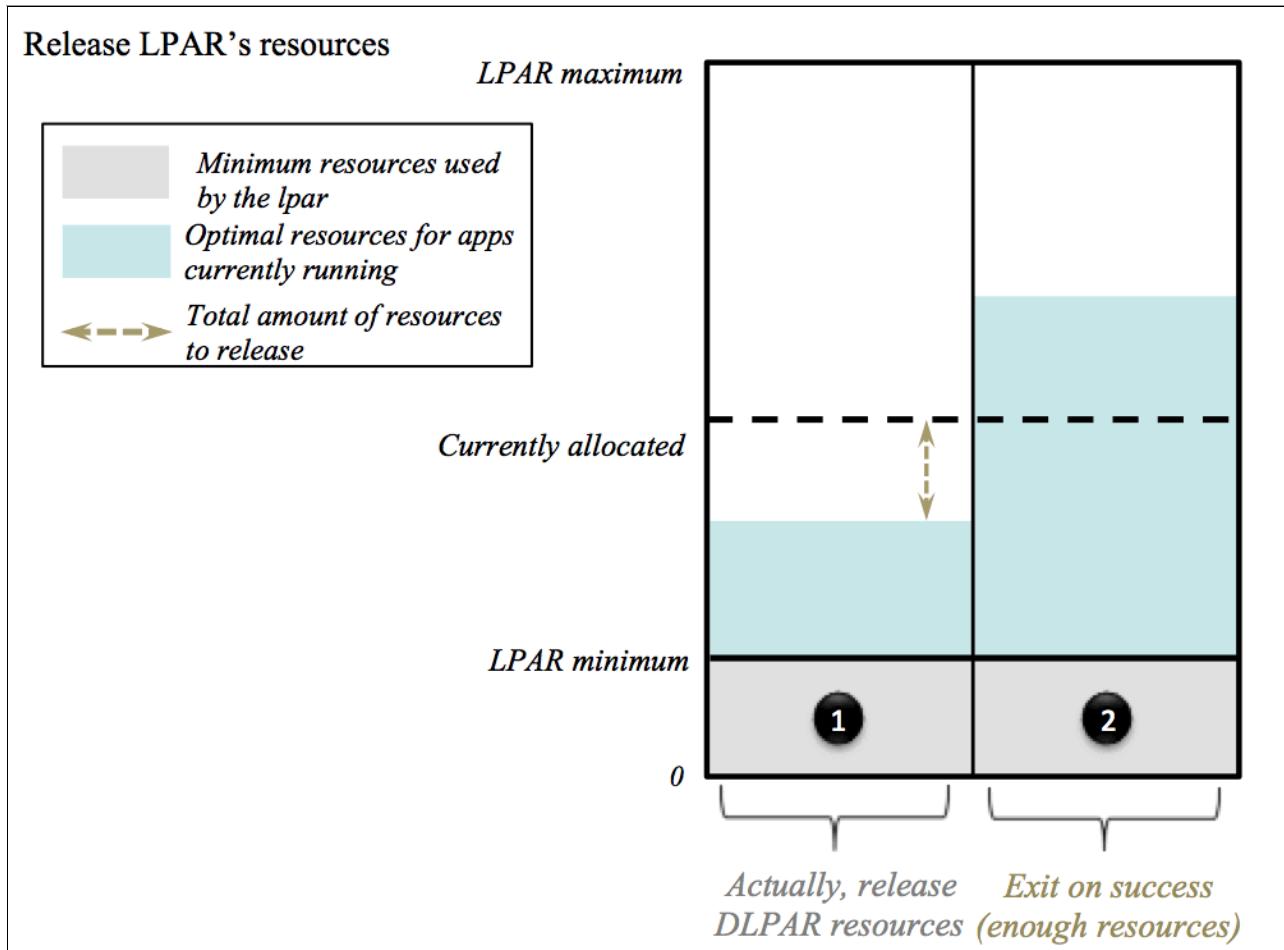


Figure 6-33 Resource computation in the releasing process

Releasing resources from the LPAR to the CEC

DLPAR resources are released by the HMC `chhwres` command. Table 6-24 shows the commands that PowerHA SystemMirror uses to release resources from the LPAR. You do not need to run these commands.

Table 6-24 Release resources from the LPAR to the CEC through the HMC

Dedicate memory	<code>chhwres -m <cec> -p <1par> -o r -r mem -q <mb_of_memory></code>
Dedicate CPU	<code>chhwres -m <cec> -p <1par> -o r -r proc --procs <cpu></code>
Shared CPU	<code>chhwres -m <cec> -p <1par> -o r -r proc --procs <vp> --proc_units <pu></code>

Tip: The `chhwres` commands are given in Table 6-24 as examples, but it is not necessary for the user to run these commands. These commands are embedded in to the ROHA run time, and run as part of the ROHA acquisition and release steps.

A timeout is given with the `-w` option and this timeout is set to the configured value at the cluster level (DLPAR operations timeout) added with 1 minute per GB. So, for example, to release 100 GB, if the default timeout value is set to 10 minutes, the timeout is set to 110 minutes ($10 + 100$).

For large memory releases, for example, instead of making one 100 GB release request, make 10 requests of a 10 GB release. You can see the logs in the `hacmp.out` log file.

Identifying the resource to release

The diagram that is shown in Figure 6-34 shows three cases of DLPAR, CoD, and Enterprise Pool CoD release for memory and processors.

At release, the de-allocation order is reversed, On/Off CoD resources are preferably released, preventing the user from paying for extra costs. Figure 6-34 shows the process.

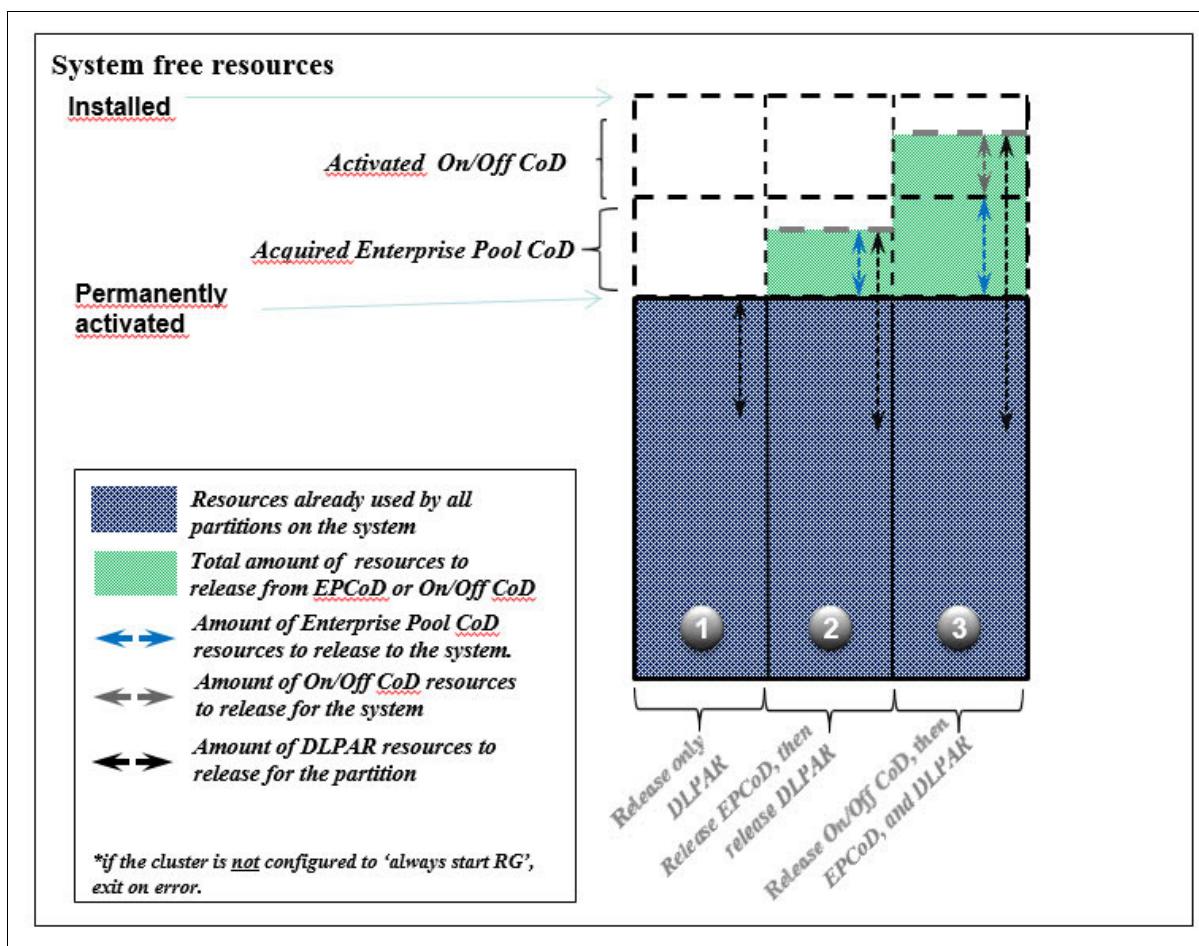


Figure 6-34 Identifying the source to release

There are three cases in the identify step:

1. There are no On/Off CoD or Enterprise Pool resources that are used by the CEC. Therefore, no resources need to be released to On/Off CoD or the Enterprise Pool.
2. There are On/Off resources that are allocated to the CEC. Some of the On/Off CoD resources are deactivated up to the amount, but no more than, of the DLPAR resources being released.
3. There are both On/Off CoD resources and Enterprise Pool resources that are allocated to the CEC. Then, On/Off CoD resources are deactivated up to the amount, but no more than, of the DLPAR resources to be released.

Alternative case: If there are no On/Off CoD resources that are activated on the CEC, only return back Enterprise Pool resources to the pool.

Generating “unreturned” resources

In this step, if some EPCoD resource is identified, it is possible for PowerHA SystemMirror to release them to EPCoD immediately, and before the DLPAR remove operation even starts.

PowerHA SystemMirror raises an asynchronous process to do the DLPAR remove operation. PowerHA SystemMirror does not need to wait for the DLPAR operation to complete. So, PowerHA SystemMirror on standby mode can bring the online RGs quickly.

This asynchronous process happens only under the following two conditions:

1. If there are only two nodes in the cluster and those two nodes are on different managed systems, or if there are more than two nodes in the cluster and the operation is a move to target node and the source node is on another managed system.
2. If you set the Force synchronous release of DLPAR resources as the default, which is No, see 6.2.5, “Change/Show Default Cluster Tunable” on page 160.

About the “unreturned” resources

The unreturned resource is one function of EPCoD. This function enables you to remove Mobile CoD resources from a server that the server cannot reclaim because they are still in use; these resources become unreturned resources. From the EPCoD pool point of view, the resource is back and can be assigned to other nodes. This function can allow the standby node to acquire the resource and application to use them while the resource is being released by the primary node.

When an unreturned resource is generated, a grace period timer starts for the unreturned Mobile CoD resources on that server, and EPCoD is in Approaching out of compliance (within server grace period) status. After the releasing operation completes physically on the primary node, the unreturned resource is reclaimed automatically, and the EPCoD's status is changed back to In compliance.

Note: For more information about the Enterprise Pool's status, see the [IBM Knowledge Center](#).

Releasing resources

This section describes the release resource concept.

Deactivating the On/Off CoD resource

CoD resources are deactivated through the HMC command-line interface **chcod**. PowerHA SystemMirror runs the command automatically.

Releasing (or returning back) the Enterprise Pool CoD resource

Enterprise Pool CoD resources are returned back to the pool by using the HMC `chcodpool` command. PowerHA SystemMirror runs the command automatically.

6.7.2 Synchronous and asynchronous mode

Because release requests take time, PowerHA SystemMirror tries to release DLPAR resources asynchronously. In asynchronous mode, the process of release is run in the background and gives priority back to other tasks.

By default, the release is asynchronous. This default behavior can be changed with a cluster tunable.

But synchronous mode is automatically computed as follows:

- ▶ All nodes of a cluster are on the same CEC.
- ▶ Otherwise, the backup LPARs of the given list of RGs are on the same CEC.

For example, if one PowerHA SystemMirror cluster includes two nodes, the two nodes are deployed on different servers and the two servers share one Power Enterprise Pool. In this case, if you are keeping asynchronous mode, you can benefit from the RG move scenarios because EPCoD's unreturned resource feature and asynchronous release mode can reduce takeover time.

During RG offline, operations to release resources to EPCoD pool can be done even if physical resources are not free on the server at that time. The freed resources are added back to the EPCoD pool as available resources immediately so that the backup partition can use these resources to bring the RG online at once.

6.7.3 Automatic resource release process after an operating system crash

Sometimes, the ROHA resources have not been released by one node before the node failed or crashed. In this kind of cases, an automatic mechanism is implemented to release these resources when the node restarts.

A history of what was allocated for the partition is kept in the AIX ODM object database, and PowerHA SystemMirror uses it to release the same amount of resources at boot time.

Note: You do not need to start PowerHA SystemMirror service to activate this process after an operating system restart because this operation is triggered by the `/usr/es/sbin/cluster/etc/rc.init` script, which is in the `/etc/inittab` file.

6.8 Example 1: Setting up one Resource Optimized High Availability cluster (without On/Off CoD)

This section describes how to set up a ROHA cluster without On/Off CoD.

6.8.1 Requirement

We have two IBM Power 770 D model servers, and they are in one Power Enterprise Pool. We want to deploy one PowerHA SystemMirror cluster with two nodes that are in different servers. We want the PowerHA SystemMirror cluster to manage the server's free resources and EPCoD mobile resource to automatically satisfy the application's hardware requirements before we start it.

6.8.2 Hardware topology

Figure 6-35 shows the hardware topology.

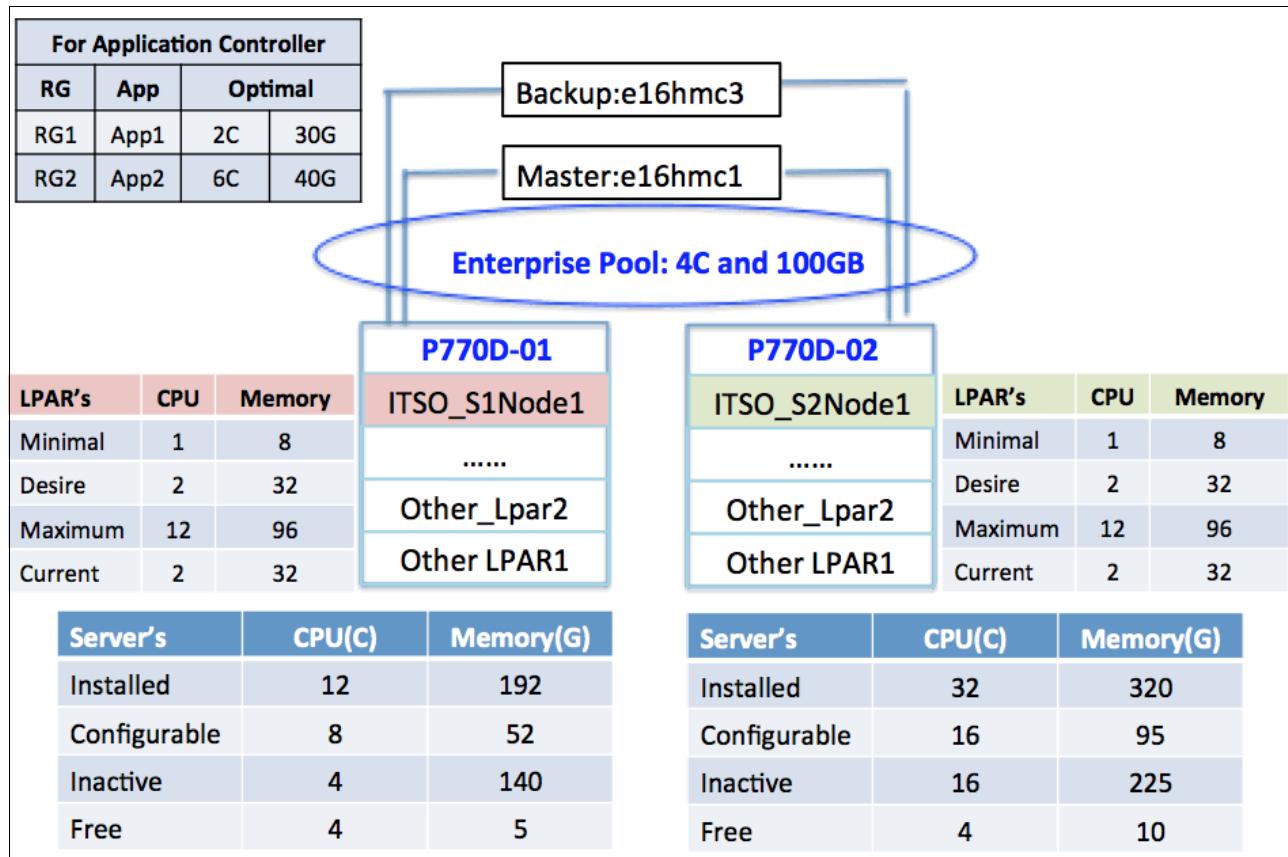


Figure 6-35 Hardware topology for example 1

The topology includes the following components for configuration:

- ▶ Two Power 770 D model servers, named P770D-01 and P770D-02.
- ▶ 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.
- ▶ P770D-01 has four inactive CPUs, 140 GB of inactive memory, four available CPUs, and 5 GB of free memory.
- ▶ P770D-02 has 16 inactive CPUs, 225 GB of inactive memory, four available CPUs, and 10 GB of free memory.
- ▶ This topology also includes the profile configuration for each LPAR.

There are two HMCs to manage the EPCoD 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 requirement.

6.8.3 Cluster configuration

This section describes the cluster configuration.

Topology and resource group configuration

Table 6-25 shows the cluster's attributes.

Table 6-25 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	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	
Resource Group	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 Failover Policy: Failover To Next Priority Node In The List Fallback Policy: Never Fallback	

Resource Optimized High Availability 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 6-26 and Table 6-27.

Table 6-26 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
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 6-27 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.

Additionally, in /etc/hosts, there are resolution details between the HMC IP and the HMC host name, as shown in Example 6-12.

Example 6-12 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 6-28 and Table 6-29.

Table 6-28 Configuration for HMC1

Items	Value
I agree to use On/Off CoD and be billed for extra costs	No (default)
Application Controller Name	AppController1
Use wanted level from the LPAR profile	No
Optimal number of gigabytes of memory	30
Optimal number of dedicated processors	2

Table 6-29 Configuration for HMC2

Items	Value
I agree to use On/Off CoD and be billed for extra costs	No (default)
Application Controller Name	AppController2
Use 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 6-30.

Table 6-30 Configuration for HMC1

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.

6.8.4 Showing the Resource Optimized High Availability configuration

Example 6-13 shows the output of the `clmgr view report roha` command.

Example 6-13 Output of the clmgr view report roha command

```
Cluster: ITSO_ROHA_cluster of NSC type <---NSC means No Site Cluster
          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
            HMC(s): 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'  desired '32'  current
              '32'  maximum '96'
              Processing mode: Dedicated
              Processors:        minimum '1'  desired '2'  current '2'
              maximum '12'
            ROHA provisioning for resource groups
              No ROHA provisioning.
          Node: ITSO_S2Node1
            HMC(s): 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'  desired '32'  current
              '32'  maximum '96'
              Processing mode: Dedicated
              Processors:        minimum '1'  desired '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 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 resource
group bring online or offline scenarios because we only want to simulate ONLY
Enterprise Pool scenarios. Please 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' CPU(s) <-- this 4CPU is assigned to
P770D-01 manually to simulate 4 free processor resource
    Left: '20' CPU.days
    Yes: 'DEC_2CEC'

Enterprise pool
    Yes: 'DEC_2CEC' <-- this is 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 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 resource
group 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 is assigned to any of 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 CPU(s): '4'
    Available CPU(s): '4'
    Unreturned CPU(s): '0'
Used by: 'rar1m3-9117-MMD-1016AAP'
    Activated memory: '0' GB
    Unreturned memory: '0' GB
    Activated CPU(s): '0' CPU(s)
    Unreturned CPU(s): '0' CPU(s)
Used by: 'r1r9m1-9117-MMD-1038B9P'
    Activated memory: '0' GB
    Unreturned memory: '0' GB
    Activated CPU(s): '0' CPU(s)
    Unreturned CPU(s): '0' CPU(s)
```

6.9 Test scenarios of Example 1 (without On/Off CoD)

Based on the cluster configuration in 6.5, “Resource acquisition and release process introduction” on page 174, this section introduces several testing scenarios:

- ▶ Bringing two resource groups online
- ▶ Moving one resource group to another node
- ▶ Restarting with the current configuration after the primary node crashes

6.9.1 Bringing two resource groups online

When PowerHA SystemMirror starts the cluster service on the primary node (ITSO_S1Node1), the two RGs are online. The procedure that is related to ROHA is described in Figure 6-36 on page 199.

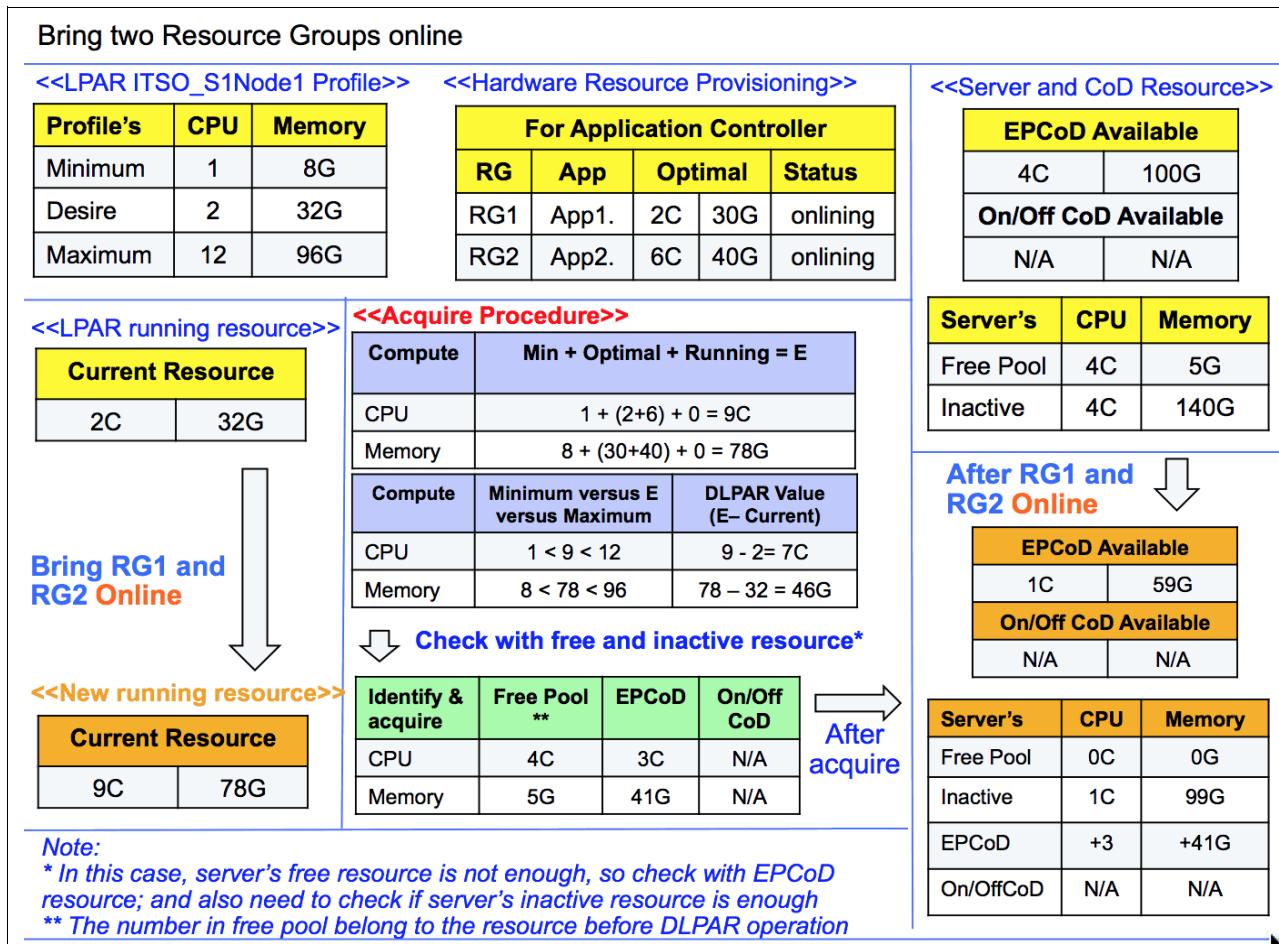


Figure 6-36 Resource acquisition procedure to bring two resource groups online

Section 6.6, “Introduction to resource acquisition” on page 175 introduces four steps for PowerHA SystemMirror to acquire resources. In this case, the following section provides the detailed description for the four steps.

Query step

PowerHA SystemMirror queries the server, the EPCoD, the LPARs, and the current RG information. The data is shown in yellow in Figure 6-36.

Compute step

In this step, PowerHA SystemMirror computes how many resources to be added through DLPAR. It needs 7C and 46 GB. The purple table shows the process in Figure 6-36. 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 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, $9 - 2 = 7$, we get the CPU number to add through the DLPAR.

Identify and acquire step

After the compute step, PowerHA SystemMirror identifies how to satisfy the requirement. For CPU, it gets the remaining 4C of this server and 3C from EPCoD. For memory, it gets the remaining 5 GB of this server and 41 GB from EPCoD. The process is shown in the green table in Figure 6-36 on page 199. 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 through the HMC (by running the **chcodpool** command). At this time, there are seven CPUs in the free pool, then PowerHA SystemMirror assigns all of them to LPAR (ITSO_S1Node1) through 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, then adds all the free pool's resources to the LPAR through DLPAR. To describe the process clearly, the free pool means only the available resources of one server before adding the EPCoD's resources to it.

The orange tables (Figure 6-36 on page 199) show the result after the resource acquisition, and include the LPAR's running resource, EPCoD, and the server's resource status.

Tracking the hacmp.out log to know what is happening

By reviewing the **hacmp.out** log, we know that the resources (seven CPUs and 41 memory) cost 53 seconds, as shown in Example 6-14.

Example 6-14 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:clmanageroha[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.
==== HACMPProhahparam 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 Unit(s) |
-----+
  Name          |    rar1m3-9117-MMD-1016AAP |    --> Server name
  State         |                            Operating
  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
<hr/>			
LPAR (dedicated)	Memory (GB)	CPU(s)	
Name	ITSO_S1Node1		
State	Running		
Minimum	8.00	1	
Desired	32.00	2	
Assigned	32.00	2	
Maximum	96.00	12	
<hr/>			
ENTERPRISE POOL	Memory (GB)	CPU(s)	
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
<hr/>			
TRIAL CoD	Memory (GB)	CPU(s)	
State	Not Running	Not Running	
Activated	0.00	0	
Days left	0	0	
Hours left	0	0	
<hr/>			
ONOFF CoD	Memory (GB)	CPU(s)	
State	Available	Running	
Activated	0.00	4	--> just ignore it
Unreturned	0.00	0	
Available	140.00	4	
Days available	9927	9944	
Days left	0	20	
Hours left	0	2	
<hr/>			
OTHER	Memory (GB)	CPU(s)	
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 Desired	Memory (GB)	CPU(s) PU(s)/VP(s)

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 CPU(s) =====
--> compute CPU process
minimal + optimal + running = total <=> current <=> maximum
1 + 8 + 0 = 9 <=> 2 <=> 12 : => 7 CPU(s)

```

```

===== 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 PU(s) for partition:       4.00 Processing Unit(s)
Total Enterprise Pool CPU(s) to allocate:     3.00 CPU(s)
Total Enterprise Pool CPU(s) to yank:         0.00 CPU(s)
Total On/Off Cod CPU(s) to activate:          0.00 CPU(s) for 0 days
Total DLPAR CPU(s) to acquire:                7.00 CPU(s)
===== End =====
--> assign EPCod resource to server
clhmccmd: 41.00 GB of Enterprise Pool CoD have been allocated.
clhmccmd: 3 CPU(s) of Enterprise Pool CoD have been allocated.
--> assign all resource to LPAR
clhmccmd: 46.00 GB of DLPAR resources have been acquired.
clhmccmd: 7 VP(s) or CPU(s) and 0.00 PU(s) 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 CPU(s)      On/Off CoD processor: 0.00 CPU(s)
Enterprise Pool processor: 3.00 CPU(s)
INFO: received rc=0.
Success on 1 attempt(s).
===== 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 HACMPsynresop ODM changed in PowerHA SystemMirror V7.2.1. 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.

Resource Optimized High Availability report update

The `clmgr view report roha` command output (Example 6-15) shows updates on the resources of P770D-01 and the Enterprise Pool.

Example 6-15 The update in the Resource Optimized High Availability report shows the resource acquisition process for example 1

```

# clmgr view report roha
...
Managed System 'rar1m3-9117-MMD-1016AAP' --> this is P770D-01 server
Hardware resources of 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 CPU(s): '4'
        Available CPU(s): '1'
        Unreturned CPU(s): '0'
    Used by: 'rar1m3-9117-MMD-1016AAP'
        Activated memory: '41' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '3' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)

```

Testing summary

The total time to bring the two RGs online is 68 s (from 09:11:27 to 09:12:35), and it includes the resource acquisition time, as shown in Example 6-16.

Example 6-16 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
```

6.9.2 Moving one 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_S2Node1).

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 at the primary node (ITSO_S1Node1)

Figure 6-37 describes the offline procedure at the primary node.

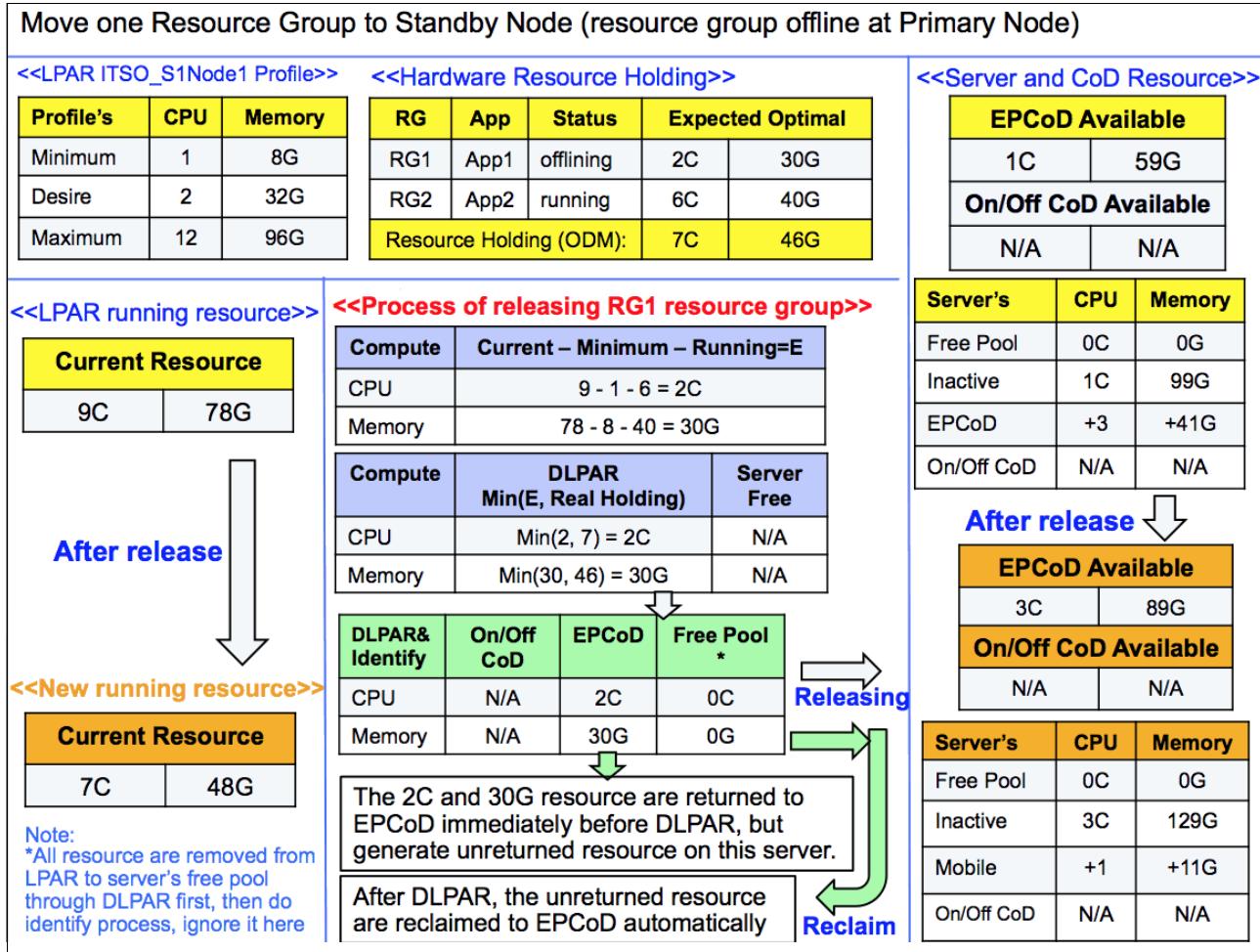


Figure 6-37 Resource group offline procedure at the primary node during the resource group move

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 tables in Figure 6-37.

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 tables show the process, as shown in Figure 6-37:

- ▶ 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 (current running) - 1 (Min) - 6 (RG2 still running) = 2C. Two CPUs can be released.
- ▶ PowerHA accounts for that sometimes you 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 kind of 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, although the resources are still in use. This process generates an unreturned resource temporarily. Figure 6-38 displays the dialog boxes that are shown on the HMC.

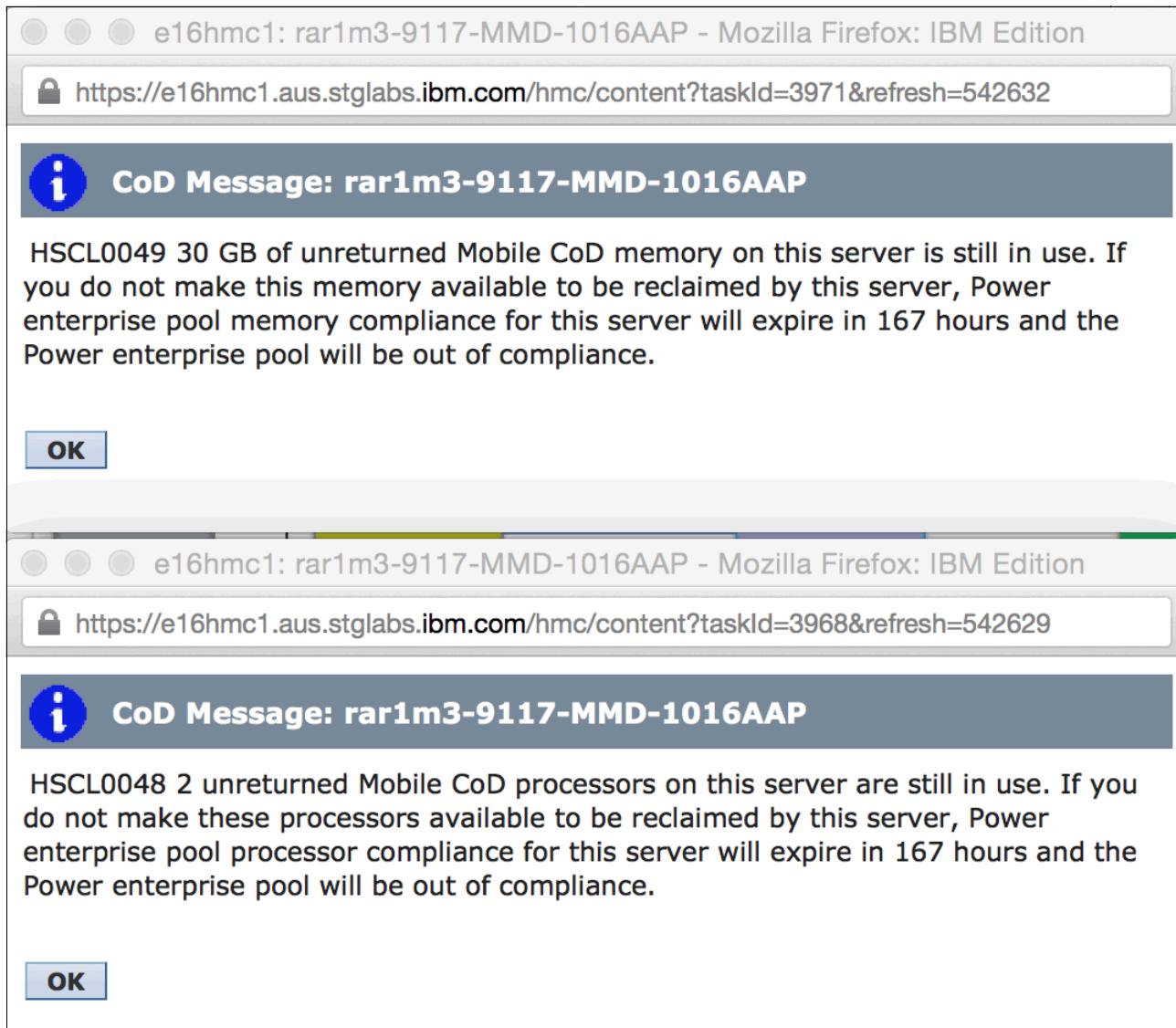


Figure 6-38 HMC message shows that there are unreturned resources that are generated

We can display the unreturned resources by using the **c1mgr view report roha** command from the AIX command line, as shown in Example 6-17.

Example 6-17 Displaying unreturned resources from the AIX command line

```
# c1mgr 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 CPU(s): '4'
        Available CPU(s): '3' --> the 2CPU has been changed to EPCoD
available status
    Unreturned CPU(s): '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 CPU(s): '1' CPU(s)
        Unreturned CPU(s): '2' CPU(s)
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
```

From the HMC command line, you can see the unreturned resources that are generated, as shown in Example 6-18.

Example 6-18 Showing the unreturned resources and the status from the HMC command line

```
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=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,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=Approaching out of compliance (within server grace
period),sequence_num=41,master_mc_name=e16hmc1,master_mc_mtms=7042-CR5*06K0040,bac
kup_master_mc_name=e16hmc3,backup_master_mc_mtms=7042-CR5*06K0036,mobile_procs=4,a
vail_mobile_procs=3,unreturned_mobile_procs=2,mobile_mem=102400,avail_mobile_mem=9
1136,unreturned_mobile_mem=30720
```

Meanwhile, PowerHA SystemMirror triggers one asynchronous process to do the DLPAR remove operation, and it removes 2C and 30 GB 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 resource is reclaimed immediately, and some messages are shown on the HMC (Figure 6-39). The Enterprise Pool's status is changed back to In compliance.

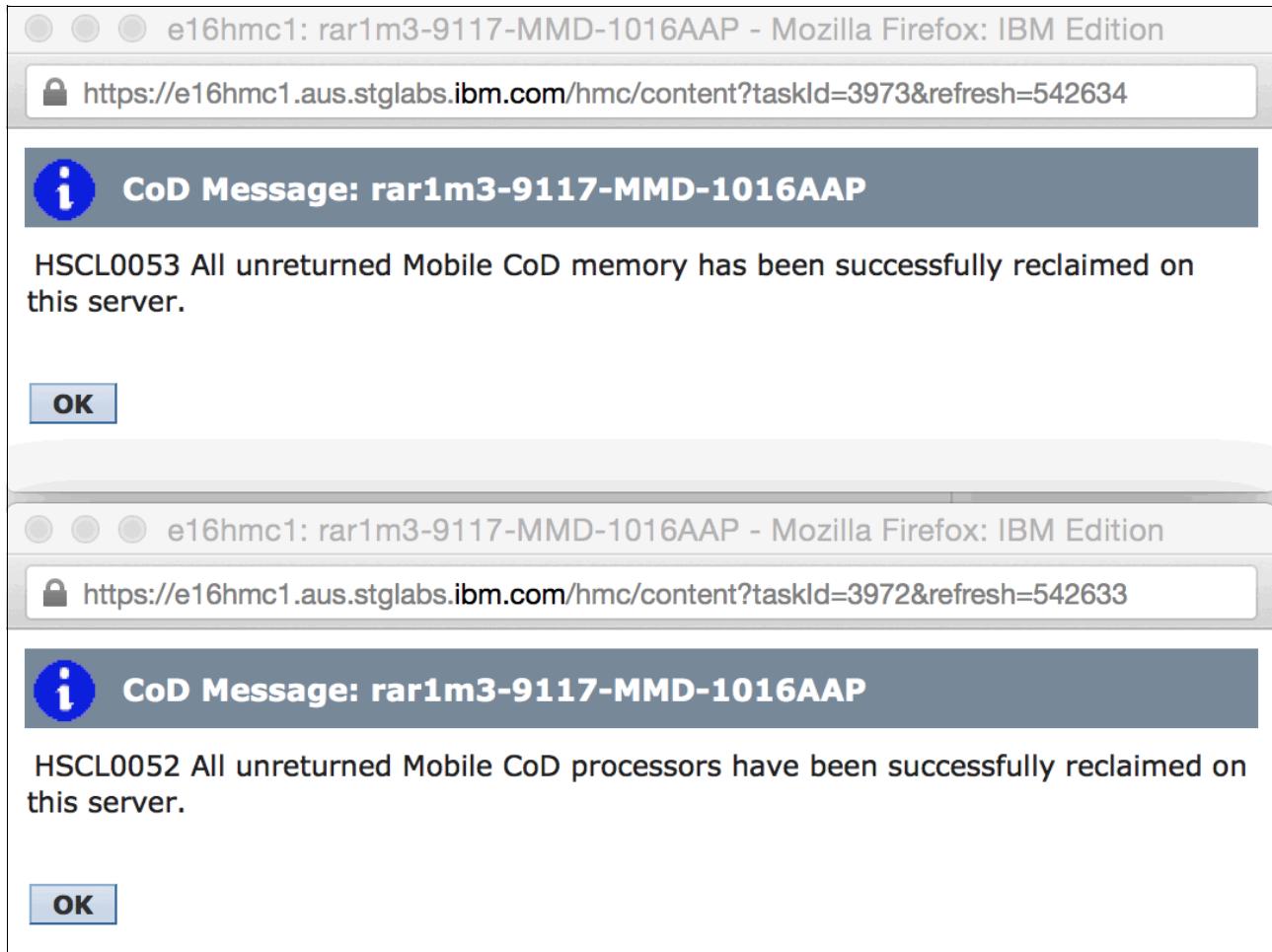


Figure 6-39 The unreturned resource is reclaimed after the DLPAR operation

You can see the changes from HMC command line, as shown in Example 6-19.

Example 6-19 Showing the unreturned resource that is reclaimed from the HMC command line

```
hsroot@e16hmc1:~> lscodpool -p DEC_2CEC --level sys
name=rar1m3-9117-MMD-1016AAP,mtms=9117-MMD*1016AAP,mobile_procs=1,non_mobile_procs=8,unretrn
ned_mobile_procs=0,inactive_procs=3,installed_procs=12,mobile_mem=11264,non_mobile_mem=532
48,unreturned_mobile_mem=0,inactive_mem=132096,installed_mem=196608
name=r1r9m1-9117-MMD-1038B9P,mtms=9117-MMD*1038B9P,mobile_procs=0,non_mobile_procs=16,unretr
ned_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=In compliance,sequence_num=41,master_mc_name=e16hmc1,
master_mc_mtms=7042-CR5*06K0040,backup_master_mc_name=e16hmc3,backup_master_mc_mtms=7042-CR
5*06K0036,mobile_procs=4,avail_mobile_procs=3,unreturned_mobile_procs=0,mobile_mem=102400,a
vail_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 the cases.

Log information in the hacmp.out file

The hacmp.out log file records the process of the RG offlineing, as shown in Example 6-20.

Example 6-20 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 CPU(s) =====
minimal + running = total <=> current <=> optimal <=> saved
1 + 6 = 7 <=> 9 <=> 2 <=> 7 : => 2 CPU(s)
===== 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 CPU(s) to return back: 2.00 CPU(s)
Total On/Off CoD CPU(s) to de-activate: 0.00 CPU(s)
Total DLPAR CPU(s) to release: 2.00 CPU(s)
===== End =====
clhmccmd: 30.00 GB of Enterprise Pool CoD have been returned.
clhmccmd: 2 CPU(s) 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 CPU(s) On/Off CoD processor: 0.00 CPU(s)
Enterprise Pool processor: 2.00 CPU(s)
Close session 22937664 at Sun Nov 8
09:12:32 CST 2015
..
```

During the releasing process, the de-allocation order is EPCoD and then the local server's free pool. Because EPCoD is shared between different servers, the standby node 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 online on 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 onlining RG.

Note: Before acquiring the process start, the 2C and 30 GB resources were available in the Enterprise Pool, so this kind of resource can also be used by standby node.

Figure 6-40 describes the resource acquisition process on the standby node (ITSO_S2Node1).

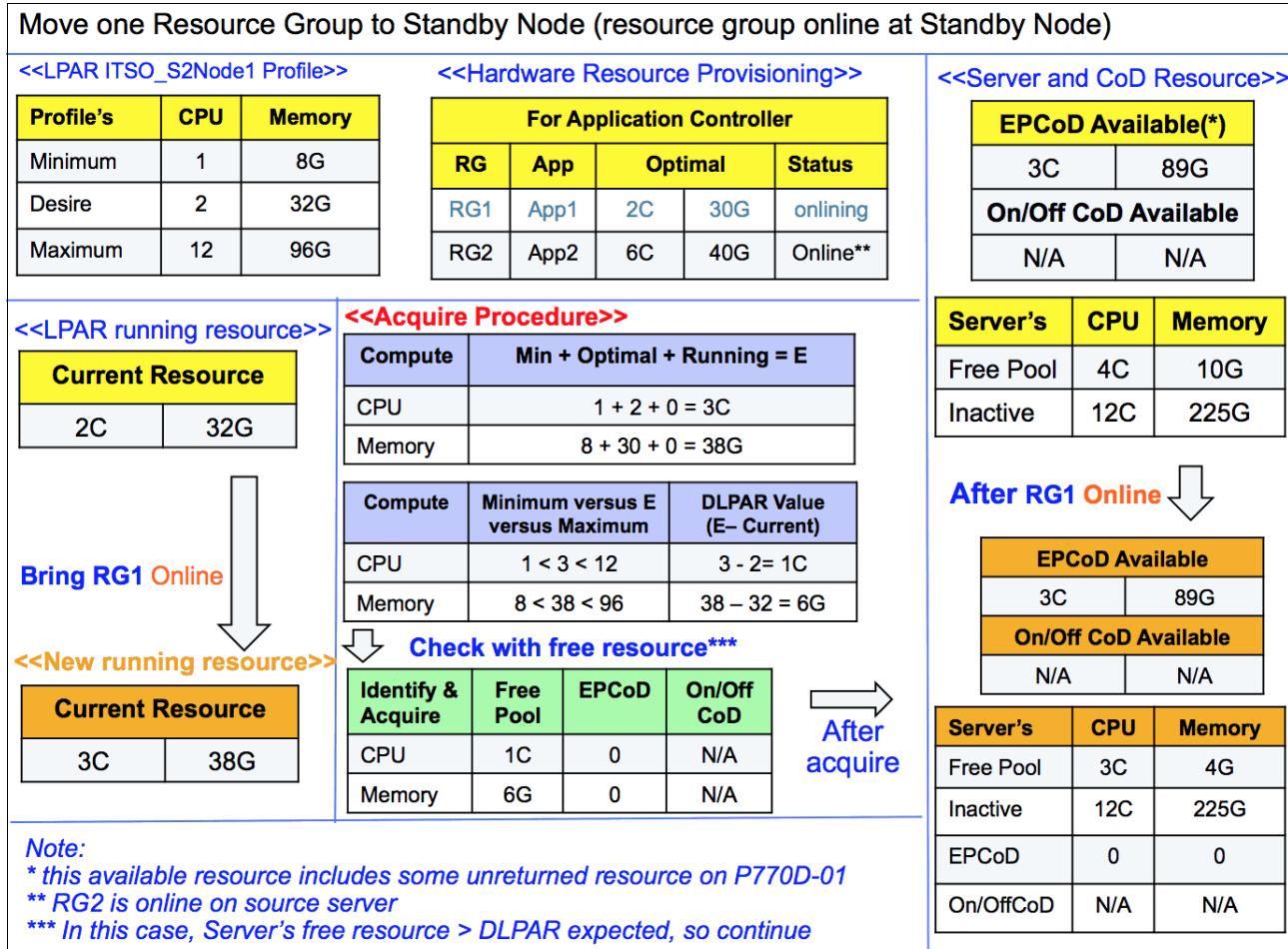


Figure 6-40 The acquisition process on standby node

This acquisition process differs from the scenario that is described in 6.9.1, “Bringing two resource groups online” on page 198. The expected resources to add to the LPAR is 1C and 6 GB and the system’s free pool can satisfy it, so it does not need to acquire resources from EPCoD.

Testing scenario summary

The total time of this RG moving is 80 seconds, from 10:53:15 to 10:53:43.

Removing the resource (2C and 30 GB) from the LPAR to a free pool on the primary node costs 257 seconds, from 10:52:51 to 10:57:08, but we are not concerned with this time because it is an asynchronous process.

Example 6-21 shows the hacmp.out information about ITSO_S1Node1.

Example 6-21 The key time stamp 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 6-22 shows the async_release.log file on ITSO_S2Node1.

Example 6-22 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[c1hmccexec:3624] : Start ssh command at Sun Nov 8 10:52:56 CST 2015
+RG1:c1hmccmd[c1hmccexec: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 '$hscroot@9.3.207.130 \'lssyscfg -r sys -m
9117-MMD*1016AAP -F name 2>&1\''
+RG1:c1hmccmd[c1hmccexec:3627] : Return from ssh command at Sun Nov 8 10:52:56 CST
2015
+RG1:c1hmccmd[c1hmccexec:3624] : Start ssh command at Sun Nov 8 10:52:56 CST 2015
+RG1:c1hmccmd[c1hmccexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no '$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[c1hmccexec:3627] : Return from ssh command at Sun Nov 8 10:54:19 CST
2015
+RG1:c1hmccmd[c1hmccexec:3624] : Start ssh command at Sun Nov 8 10:54:19 CST 2015
+RG1:c1hmccmd[c1hmccexec: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[c1hmccexec:3627] : Return from ssh command at Sun Nov 8 10:55:32 CST
2015
```

```

+RG1:c1hmccmd[c1hmccexec:3624] : Start ssh command at Sun Nov 8 10:55:32 CST 2015
+RG1:c1hmccmd[c1hmccexec: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 '$hscroot@9.3.207.130 \'chhwres -m
rar1m3-9117-MMD-1016AAP -p ITSO_S1Node1 -r mem -o r -q 10240 -w 30 2>&1\''
+RG1:c1hmccmd[c1hmccexec:3627] : Return from ssh command at Sun Nov 8 10:56:40 CST
2015
+RG1:c1hmccmd[c1hmccexec:3624] : Start ssh command at Sun Nov 8 10:56:40 CST 2015
+RG1:c1hmccmd[c1hmccexec:3625] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o
ConnectionAttempts=3 -o TCPKeepAlive=no '$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[c1hmccexec:3627] : Return from ssh command at Sun Nov 8 10:57:08 CST
2015
Sun Nov 8 10:57:08 CST 2015

```

Example 6-23 shows the hacmp.out information about ITSO_S2Node1.

Example 6-23 The key time stamp 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

```

6.9.3 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 crashed 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 still can hold the same amount of resources as before the crash.

As described in 6.7.3, “Automatic resource release process after an operating system crash” on page 191, 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 6-41.

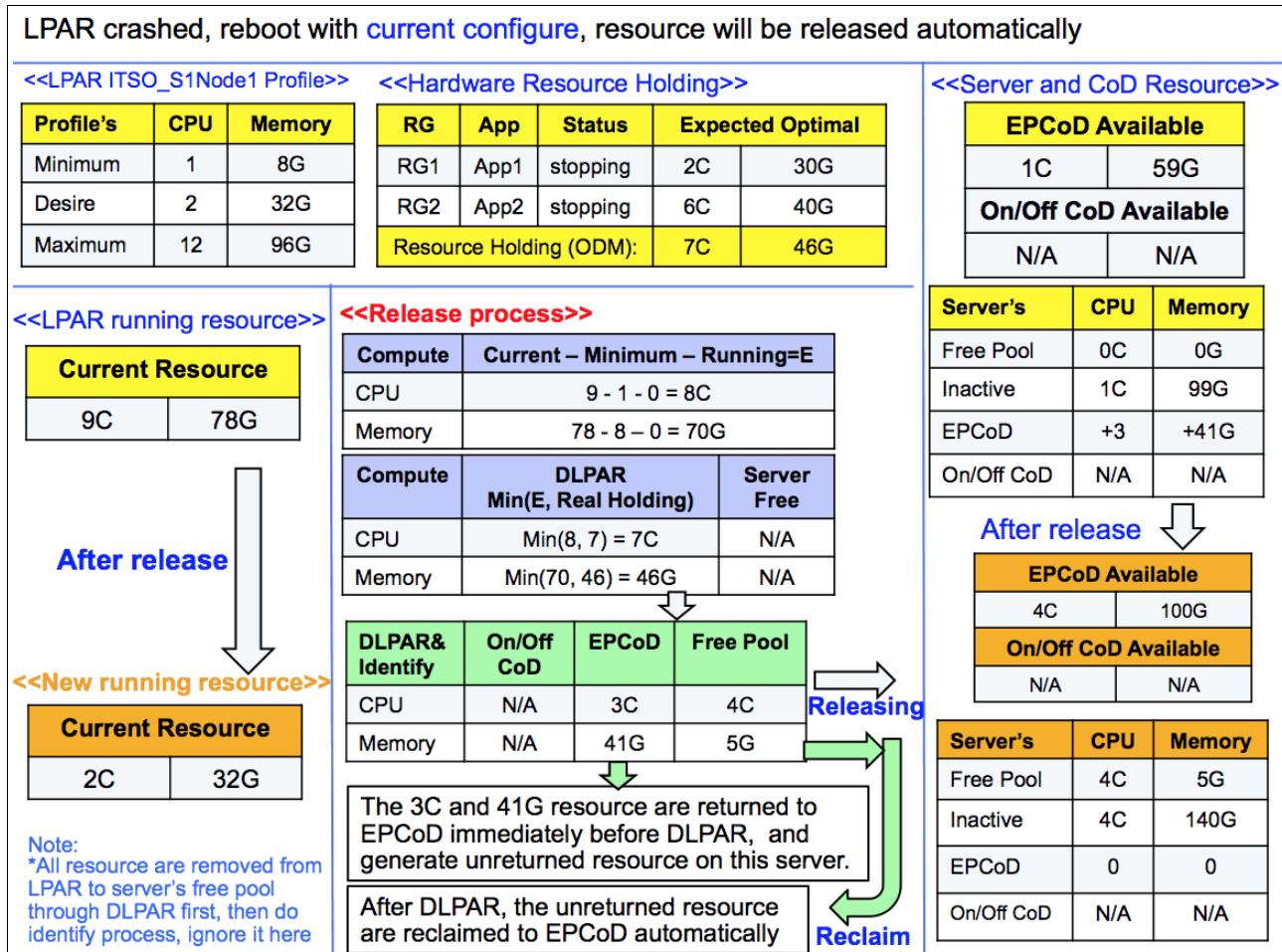


Figure 6-41 Resource release process in the ARAF process

The process is similar to “Resource group offline at the primary node (ITSO_S1Node1)” on page 206. 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 do the release operation automatically after this node starts. This operation occurs before you start the PowerHA SystemMirror service through the `smitty clstart` or the `clmgr start cluster` commands.

6.10 Example 2: Setting up one Resource Optimized High Availability cluster (with On/Off CoD)

This section describes the setup of one ROHA cluster example.

6.10.1 Requirements

We have two Power 770 D model servers in one Power Enterprise Pool, and each server has an On/Off CoD license. We want to deploy one PowerHA SystemMirror cluster, and include two nodes that are in different servers. We want the PowerHA SystemMirror cluster to manage the server's free resources, EPCoD mobile resources, and On/Off CoD resources automatically to satisfy the application's hardware requirement before starting it.

6.10.2 Hardware topology

Figure 6-42 shows the server and LPAR information of example 2.

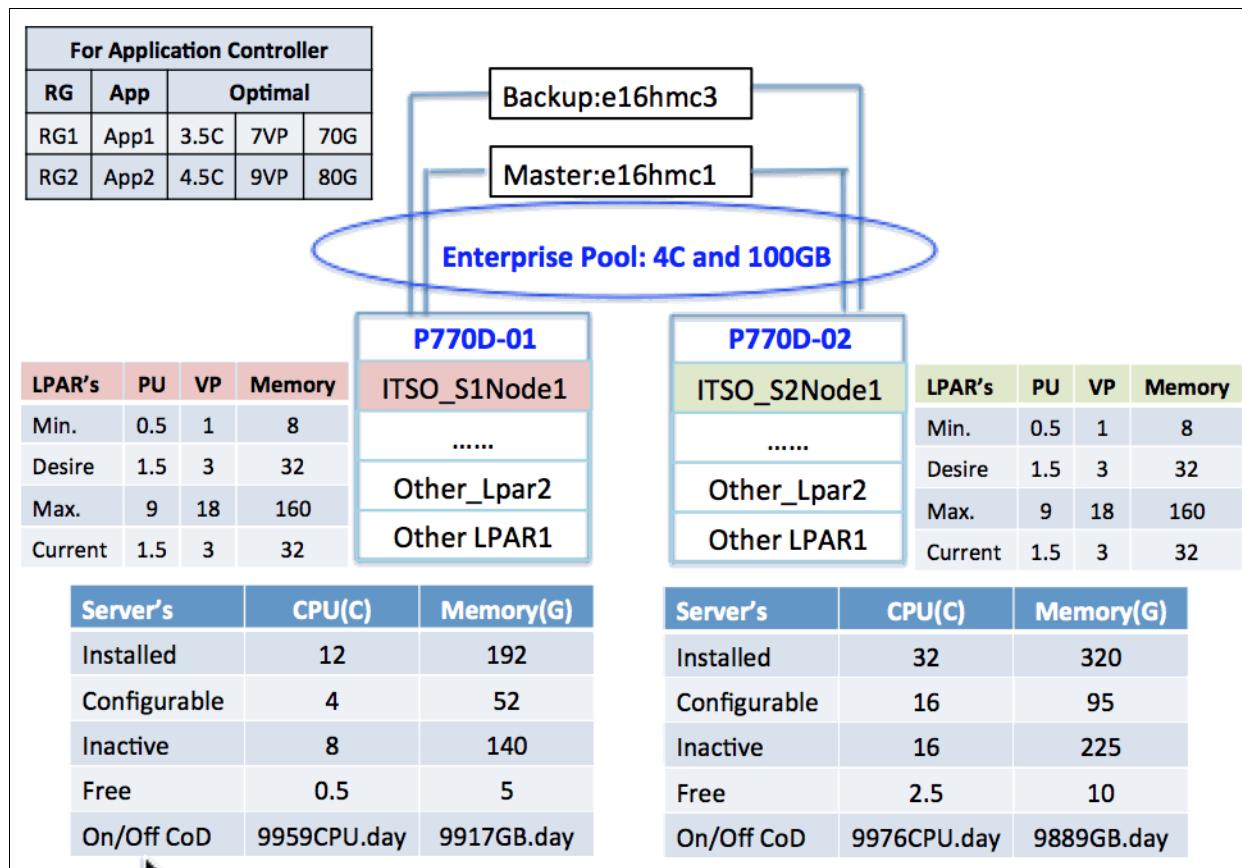


Figure 6-42 Server and LPAR information

The topology includes the following components for configuration:

- ▶ Two Power 770 D model servers that are named P770D-01 and P770D-02.
- ▶ One Power Enterprise Pool with four mobile processors and 100 GB of mobile memory.
- ▶ Each server has enabled the On/Off CoD feature.

- ▶ PowerHA SystemMirror cluster includes two nodes, ITSO_S1Node1 and ITSO_S2Node1.
- ▶ 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.
- ▶ This topology also includes the profile configuration for each LPAR.

There are two HMCs to manage the EPCoD that 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 examples, 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: Number of activating days for On/Off CoD requests (for more information, see 6.2.5, “Change>Show Default Cluster Tunable” on page 160).

If 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 only, and so we have 20 GB On/Off CoD only, even if we have 600 GB.Days available.

6.10.3 Cluster configuration

The Topology and RG configuration and HMC configuration is the same as shown in 6.8.3, “Cluster configuration” on page 193.

Hardware resource provisioning for application controller

There are two application controllers to be added, as shown in Table 6-31 and Table 6-32.

Table 6-31 Configure HMC1

Items	Value
I agree to use On/Off CoD and be billed for extra costs	Yes
Application Controller Name	AppController1
Use 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 6-32 Configure HMC1

Items	Value
I agree to use On/Off CoD and be billed for extra costs	Yes
Application Controller Name	AppController2
Use wanted level from the LPAR profile	No
Optimal number of gigabytes of memory	80

Items	Value
Optimal number of processing units	4.5
Optimal number of virtual processors	9

Cluster-wide tunables

All the tunables are at the default value, as shown in Table 6-33.

Table 6-33 Configuration of HMC1

Items	Value
DLPAR Always Start RGs	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 after changing the previous configuration.

6.10.4 Showing the Resource Optimized High Availability configuration

The **clmgr view report roha** command shows the current ROHA data, as shown in Example 6-24.

Example 6-24 Shows Resource Optimized High Availability data with the **clmgr view report roha** command

```
# clmgr view report roha
Cluster: ITSO_ROHA_cluster of NSC type --> NSC means No Site Cluster
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
    HMC(s): 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'  desired '32'  current '32'  maximum
        '160'
        Processing mode: Shared
        Shared processor pool: 'DefaultPool'
        Processing units:   minimum '0.5'  desired '1.5'  current '1.5'  maximum
        '9.0'
        Virtual processors: minimum '1'  desired '3'  current '3'  maximum '18'
    ROHA provisioning for resource groups
        No ROHA provisioning.
```

```

Node: ITSO_S2Node1 --> Information of ITSO_S2Node1 node
  HMC(s): 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'  desired '32'  current '32'  maximum
'160'
    Processing mode: Shared
    Shared processor pool: 'DefaultPool'
    Processing units:  minimum '0.5'  desired '1.5'  current '1.5'  maximum
'9.0'
    Virtual processors: minimum '1'  desired '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 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 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, not change during
resource moving
        Available memory: '100' GB -->Available for assign, will change during resource
moving
        Unreturned memory: '0' GB
    Enterprise pool processor
        Activated CPU(s): '4'
        Available CPU(s): '4'
        Unreturned CPU(s): '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 reclaim within a period time
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)

```

6.11 Test scenarios for Example 2 (with On/Off CoD)

Based on the configuration in 6.10, “Example 2: Setting up one Resource Optimized High Availability cluster (with On/Off CoD)” on page 215, this section introduces two testing scenarios:

- ▶ Bringing two resource groups online
- ▶ Bringing one resource group offline

6.11.1 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 6-43.

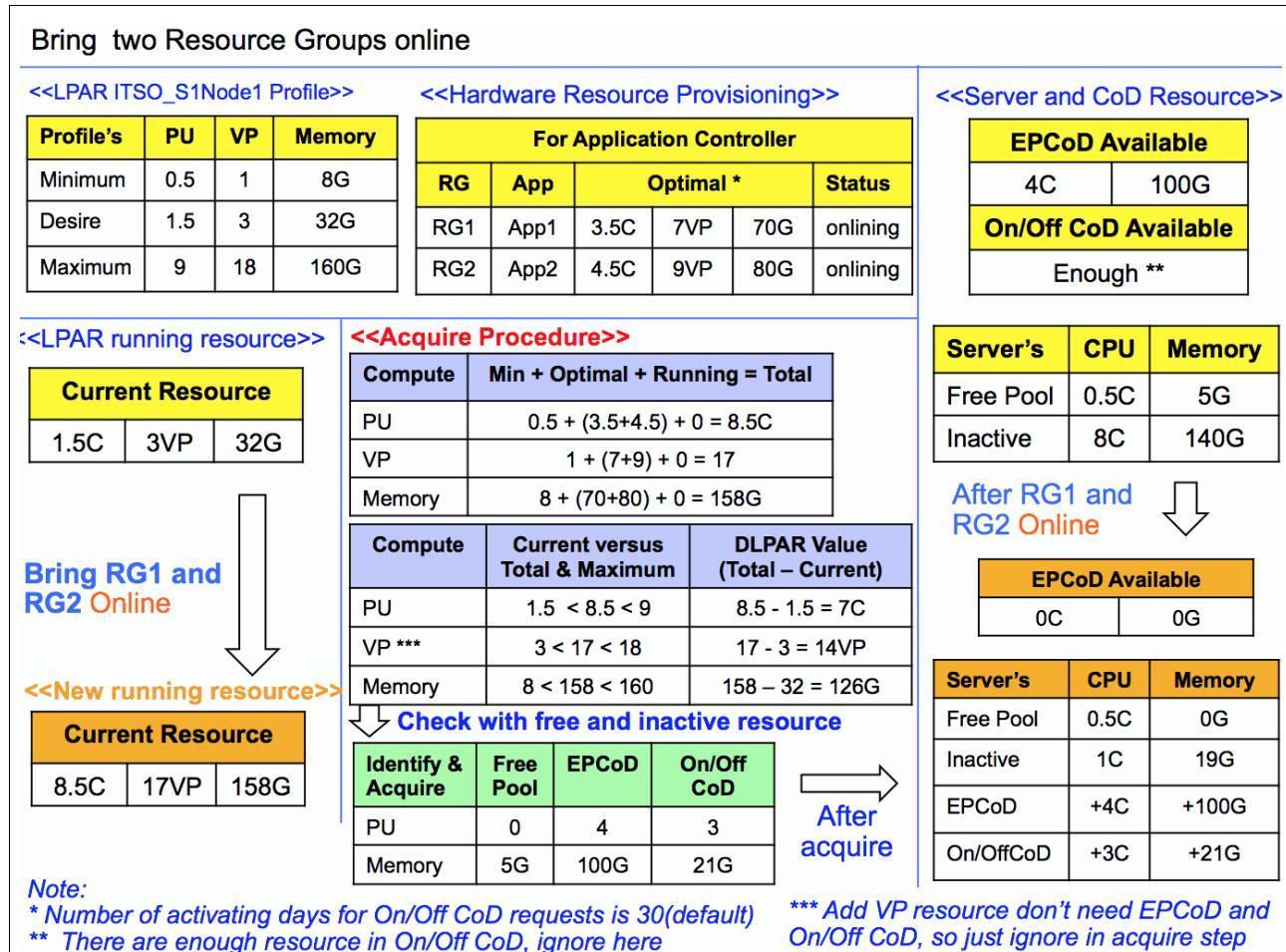


Figure 6-43 Acquire resource process of example 2

Section 6.6, “Introduction to resource acquisition” on page 175 introduces four steps for PowerHA SystemMirror to acquire the resources. In this case, the following sections are the detailed descriptions of the four steps.

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 tables in Figure 6-43.

For the On/Off CoD resources, we do not display the available resources because there are enough resources in our testing environment:

- ▶ P770D-01 has 9959 CPU.days and 9917 GB.days.
- ▶ 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 tables show this process (Figure 6-43 on page 220). 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 and more than or equal to the Minimum value.
- ▶ If this configuration satisfies the requirement, then take this value minus the current running CPU ($8.5 - 1.5 = 7$), and this 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, and even if there is 0.5 CPU in the server's free pool, the requirement is 7, 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 table in Figure 6-43 on page 220.

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. To describe this clearly, the free pool means the available resources of only one server before adding the EPCoD's resources to it.

The orange table shows (Figure 6-43 on page 220) 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 to know what is happening

From hacmp.out, you know that all the resources (sevenCPUs and 126 memory) costs 117 seconds as a synchronous process, as shown in Example 6-25:

22:44:40 → 22:46:37

Example 6-25 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 PU(s)/VP(s) ==
minimal + optimal + running = total <=> current <=> maximum
1 + 16 + 0 = 17 <=> 3 <=> 18 : => 14 Virtual
Processor(s)
minimal + optimal + running = total <=> current <=> maximum
0.50 + 8.00 + 0.00 = 8.50 <=> 1.50 <=> 9.00 : => 7.00 Processing
Unit(s)
===== 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 PU(s) for partition:      0.50 Processing Unit(s)
Total Enterprise Pool CPU(s) to allocate:    4.00 CPU(s)
Total Enterprise Pool CPU(s) to yank:         0.00 CPU(s)
Total On/Off CoD CPU(s) to activate:          3.00 CPU(s) for 30 days
Total DLPAR PU(s)/VP(s) to acquire:           7.00 Processing Unit(s) and
14.00 Virtual Processor(s)
===== End =====
c1hmccmd: 100.00 GB of Enterprise Pool CoD have been allocated.
c1hmccmd: 4 CPU(s) of Enterprise Pool CoD have been allocated.
c1hmccmd: 21.00 GB of On/Off CoD resources have been activated for 30 days.
c1hmccmd: 3 CPU(s) of On/Off CoD resources have been activated for 30 days.
c1hmccmd: 126.00 GB of DLPAR resources have been acquired.
c1hmccmd: 14 VP(s) or CPU(s) and 7.00 PU(s) 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 CPU(s)
Enterprise Pool processor: 4.00 CPU(s)

```

Resource Optimized High Availability report update

The **clmgr view report roha** command reports the ROHA data, as shown in Example 6-26.

Example 6-26 Resource Optimized High Availability data after acquiring resources in example 2

```

# clmgr 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
    HMC(s): 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' desired '32' current
'158' maximum '160'
      Processing mode: Shared
      Shared processor pool: 'DefaultPool'
      Processing units:  minimum '0.5' desired '1.5' current
'8.5' maximum '9.0'
      Virtual processors: minimum '1' desired '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
      HMC(s): 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'  desired '32'  current
              '32'  maximum '160'
              Processing mode: Shared
              Shared processor pool: 'DefaultPool'
              Processing units:   minimum '0.5'  desired '1.5'  current
              '1.5'  maximum '9.0'
              Virtual processors: minimum '1'  desired '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 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' CPU(s)
      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 CPU(s): '4'
        Available CPU(s): '0'
        Unreturned CPU(s): '0'
    Used by: 'rar1m3-9117-MMD-1016AAP'
        Activated memory: '100' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '4' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
    Used by: 'r1r9m1-9117-MMD-1038B9P'
        Activated memory: '0' GB
        Unreturned memory: '0' GB
        Activated CPU(s): '0' CPU(s)
        Unreturned CPU(s): '0' CPU(s)
```

The **clmgr view report roha** command output (Example 6-26 on page 222) has some updates about the resources of P770D-01, Enterprise Pool, and On/Off CoD.

How to calculate the On/Off CoD consumption

In this case, before bringing the two RGs online, the remaining resources in On/Off CoD are shown in Example 6-27.

Example 6-27 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 is online, the status of the On/Off CoD resource is shown in Example 6-28.

Example 6-28 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' CPU(s)
    Left: '90' CPU.days
```

For processor, 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).

6.11.2 Bringing one resource group offline

This section introduces the process of RG offline. Figure 6-44 shows the overall process.

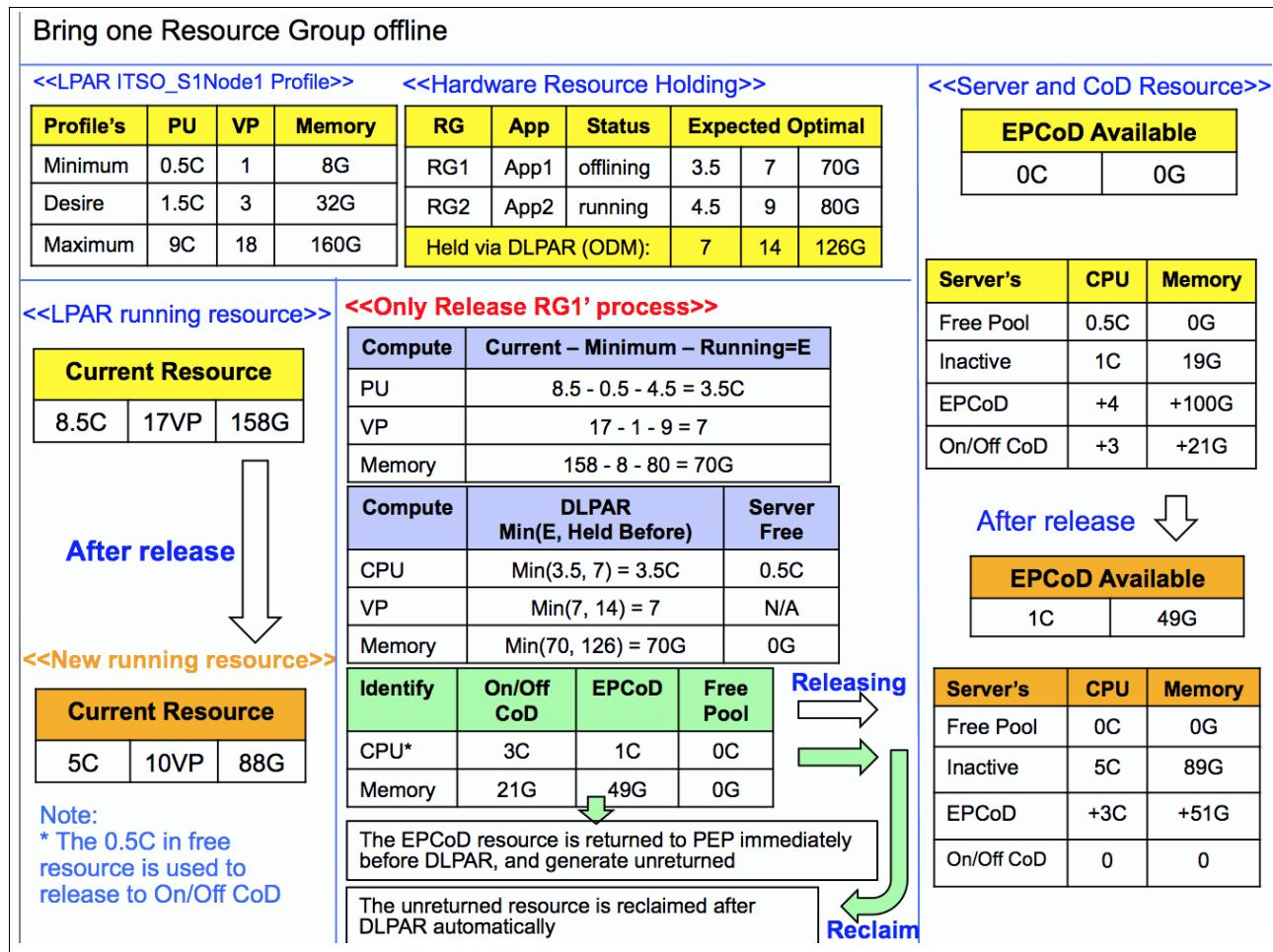


Figure 6-44 Overall release process of example 2

The process is similar to the one that is shown in 6.9.2, “Moving one resource group to another node” on page 205.

In the release process, the de-allocation order is On/Off CoD, then EPCoD, and then the server's free pool because you always need to pay an extra cost for the On/Off CoD.

After the release process completes, you can find the detailed information about compute, identify, and release processes in the hacmp.out file, as shown in Example 6-29.

Example 6-29 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 PU(s)/VP(s) ==
minimal + running = total <=> current <=> optimal <=> saved
1 + 9 = 10 <=> 17 <=> 7 <=> 14 : => 7 Virtual
Processor(s)
minimal + running = total <=> current <=> optimal <=> saved
0.50 + 4.50 = 5.00 <=> 8.50 <=> 3.50 <=> 7.00 : => 3.50
Processing Unit(s)
===== 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 CPU(s) to return back: 1.00 CPU(s)
Total On/Off CoD CPU(s) to de-activate: 3.00 CPU(s)
Total DLPAR PU(s)/VP(s) to release: 7.00 Virtual Processor(s) and
3.50 Processing Unit(s)
===== End =====
clhmccmd: 49.00 GB of Enterprise Pool CoD have been returned.
clhmccmd: 1 CPU(s) 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 CPU(s)
Enterprise Pool processor: 1.00 CPU(s)
```

6.12 Hardware Management Console high availability introduction

More than one HMC can be configured for a node, so that if one HMC fails to respond, the ROHA function can switch to the other HMC.

This section describes the mechanism that enables the HMC to switch from one HMC to another HMC.

Suppose that you have, for a given node, three HMCs in the following order: HMC1, HMC2, and HMC3. (These HMCs can be set either at the node level, at the site level, or at the cluster level. What counts at the end is that you have an ordered list of HMCs for a given node).

A given node uses the first HMC in its list, for example, HMC1, and uses it while it works.

HMC1 might fail for different reasons. For example:

1. HMC1 is not reachable by the **ping** command.

One parameter controls the **ping** command in the HMC: Timeout on ping (which is set by default to 3 seconds, and you cannot adjust it). If an HMC cannot be pinged after this timeout, you cannot use it through **ssh**, so switch immediately to another HMC, in this case the HMC following the current one in the list (for example, HMC2).

2. HMC1 is not reachable through SSH:

- SSH is not properly configured between the node and HMC1, so it is not worth trying to use HMC1, and it is best to switch to another HMC, in this case, the HMC following the current one in the list, for example, HMC2.
- SSH has temporary conditions that prevent it from responding.

Two parameters controls the **ssh** command on the HMC:

- Connect Attempts (which is set by default to 5).
- Connect Timeout (which is set by default to 5), meaning that after a 25-second delay, the HMC can be considered as not reachable through **ssh**.

If the HMC is not reachable through **ssh**, it is not worth trying to perform an **hmc** command through **ssh** on it, and the best is to switch at to another HMC. In this case, the HMC following the current one in the list, for example HMC2.

3. The HMC is repeatedly busy.

When the HMC is processing a command, it cannot perform another command at the same time. The command fails with RC=-1 with the HSCL3205 message indicating that the HMC is busy.

The PowerHA SystemMirror ROHA function has a retry mechanism that is controlled by two parameters:

- **RETRY_COUNT**, which indicates how many retries must be done.
- **RETRY_DELAY**, which indicates how long to wait between retries.

When the HMC is busy, the retry mechanism is used until declaring that the HMC is flooded.

When the HMC is considered flooded, it is not worth using it again, and the best is to switch immediately to another HMC, which is the HMC following the current one in the list, for example, HMC2.

4. The HMC returns an application error. Several cases can occur:

- One case is when you request an amount of resources that is not available, and the same request is attempted with another smaller amount.
- A second case is when the command is not understandable by the HMC, which is more like a programming bug. In these cases, the bug must be debugged at test time. In any case, this is not a reason to switch to another HMC.

If you decide to switch to another HMC, consider the next HMC of the list, and use it.

If the first HMC is not usable (HMC1), you are currently using the second HMC in the list (HMC2), which helps prevent the ROHA function from trying again and failing again by using the first HMC (HMC1). You can add (persistence) into the ODM for which HMC is being used (for example, HMC2).

This mechanism enables the ROHA function to skip the failing HMCs, and to use the HMC that works (in this case, HMC2). At the end of the session, the persistence into the ODM is cleared, meaning that the first HMC in the list is restored to its role of HMC1 or the first in the list.

6.12.1 Switching to the backup HMC for the Power Enterprise Pool

For Enterprise Pool operations, querying operations can be run on the master or backup HMC, but changing operations must run on the master HMC. If the master HMC fails, the PowerHA SystemMirror actions are as follows:

- ▶ For querying operations, PowerHA SystemMirror tries to switch to the backup HMC to continue the operation, but does not set the backup HMC as the master.
- ▶ For changing operations, PowerHA SystemMirror tries to set the backup HMC as the master, and then continues the operation. Example 6-30 shows the command that PowerHA SystemMirror performs to set the backup HMC as the master. This command is triggered by PowerHA SystemMirror automatically.

Example 6-30 Setting the backup HMC as the master

```
chcodpool -o setmaster -p <pool> --mc backup
```

There are some prerequisites in PowerHA SystemMirror before switching to the backup HMC when the master HMC fails:

- ▶ Configure the master HMC and the backup HMC for your Power Enterprise Pool.
For more information about how to configure the backup HMC for the Power Enterprise Pool, see the [IBM Knowledge Center](#) and *Power Enterprise Pools on IBM Power Systems*, REDP-5101.
- ▶ Both HMCs are configured in PowerHA SystemMirror.
- ▶ Establish password-less communication between the PowerHA SystemMirror nodes to the two HMCs.
- ▶ Ensure reachability (pingable) from PowerHA SystemMirror nodes to the master and backup HMCs.
- ▶ Ensure that all of the servers that participate in the pool are connected to the two HMCs.
- ▶ Ensure that the participating servers are in either the Standby state or the Operating state.

6.13 Test scenario for HMC failover

This section shows how PowerHA SystemMirror switches the HMC automatically when the primary HMC fails.

6.13.1 Hardware topology

Figure 6-45 shows the initial status of the hardware topology.

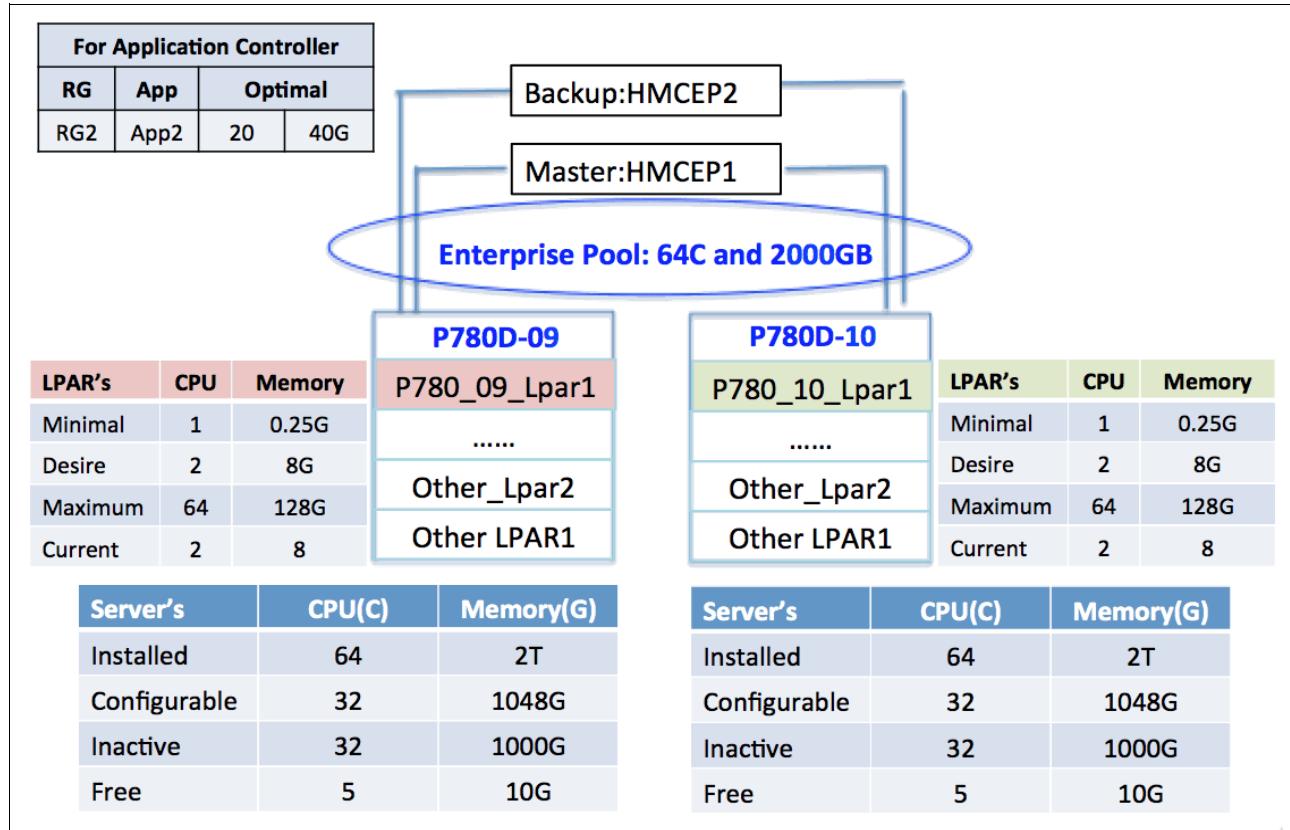


Figure 6-45 Initial status of the hardware topology

The topology includes the following components:

- ▶ Two Power 780 D model servers, named P780D_09 and P780D_10.
- ▶ One Power Enterprise Pool, which has 64 mobile processors and 2 TB of mobile memory resources.
- ▶ There are 64 CPUs and 2 TB of memory that are installed in P780D_09, 32 CPUs, 1 TB of memory that is configured, and another 32 CPUs and 1 TB of memory are in the inactive status. At this time, there are five CPUs and 10 GB of memory available for the DLPAR.
- ▶ The PowerHA SystemMirror cluster includes two nodes:
 - P780_09_Lpar1
 - P780_10_Lpar2
- ▶ The PowerHA SystemMirror cluster includes one RG (RG2); this RG has one application controller (app2) configured hardware resource provisioning.

- ▶ This application needs 20 C and 40 G when it runs.
- ▶ There is no On/Off CoD in this testing.

There are two HMCs to manage the EPCoD, named HMCEP1 and HMCEP2.

HMCEP1 is the master and HMCEP2 is the backup, as shown in Example 6-31.

Example 6-31 HMCs that are available

```
hscroot@HMCEP1:~> lscodpool -p 0019 --level pool
name=0019,id=0019,state=In
compliance,sequence_num=5,master_mc_name=HMCEP1,master_mc_mtms=V017-ffe*d33e8a1,ba
ckup_master_mc_name=HMCEP2,backup_master_mc_mtms=V017-f93*ba3e3aa,mobile_procs=64,
avail_mobile_procs=64,unreturned_mobile_procs=0,mobile_mem=2048000,avail_mobile_me
m=2048000,unreturned_mobile_mem=0
```

In the AIX /etc/hosts file, define the resolution between the HMC IP address, and the HMC's host name, as shown in Example 6-32.

Example 6-32 Define the resolution between the HMC IP and HMC name in /etc/hosts

```
172.16.50.129 P780_09_Lpar1
172.16.50.130 P780_10_Lpar1
172.16.51.129 testservice1
172.16.51.130 testservice2
172.16.50.253 HMCEP1
172.16.50.254 HMCEP2
```

Start the PowerHA SystemMirror service on P780_09_Lpar1. During the start, PowerHA SystemMirror acquires resources from the server's free pool and EPCoD (Figure 6-46).

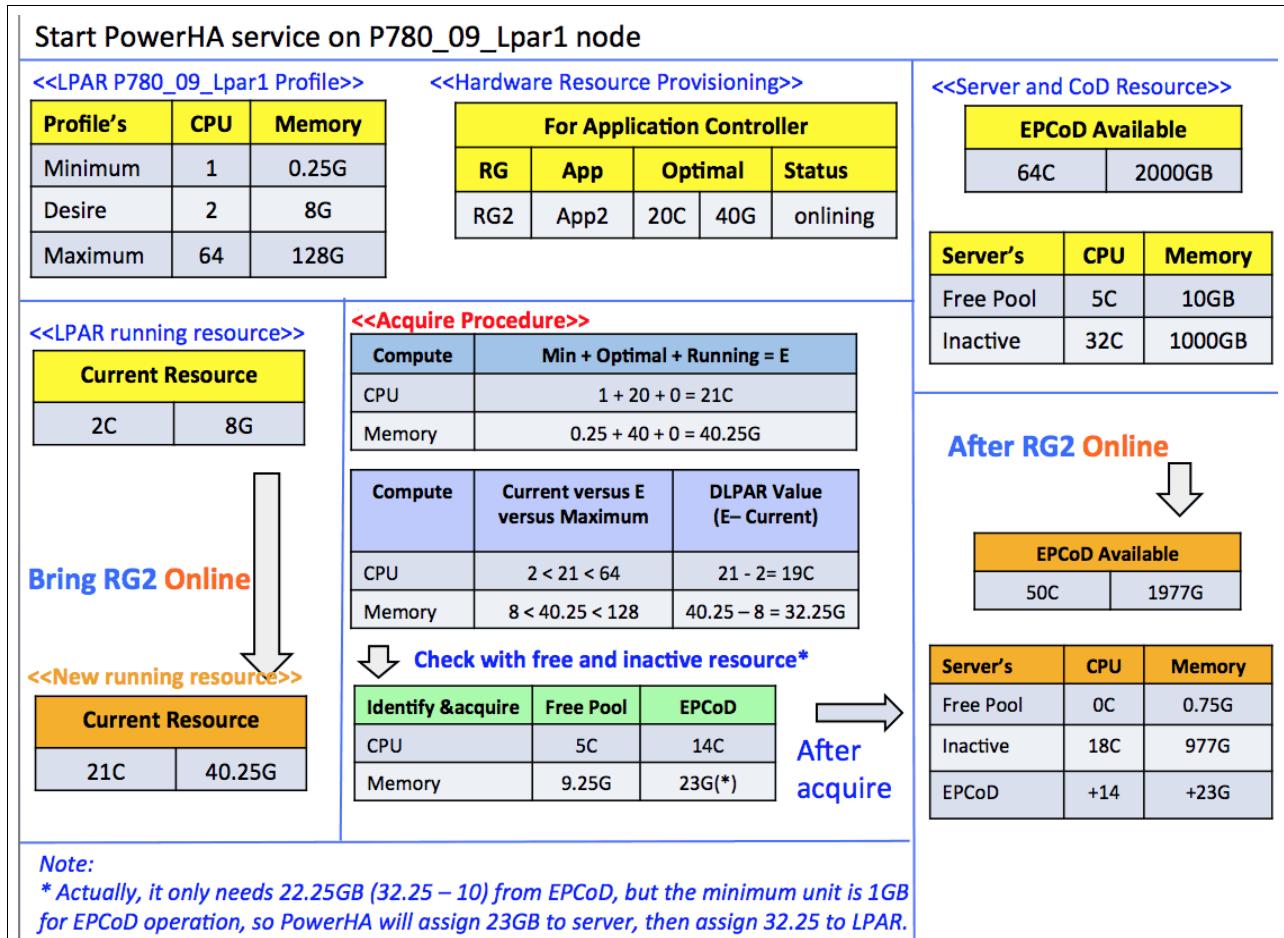


Figure 6-46 Resource Acquisition process during the start of the PowerHA SystemMirror service

In this process, HMCEP1 acts as the primary HMC and does all the query and resource acquisition operations. Example 6-33 and Example 6-34 on page 232 show the detailed commands that are used in the acquisition step.

Example 6-33 EPCoD operation during resource acquisition (hacmp.out)

```
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-o TCPKeepAlive=no hscroot@HMCEP1 'chcodpool -p 0019 -m SVRP7780-09-SN060COAT -r
mem -o add -q 23552 2>&1' -->23552 means 23 GB
...
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-o TCPKeepAlive=no hscroot@HMCEP1 'chcodpool -p 0019 -m SVRP7780-09-SN060COAT -r
proc -o add -q 14 2>&1'
```

Example 6-34 DLPAR add operation in the acquire step

```
+testRG2:c1hmccmd[c1hmccexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-o TCPKeepAlive=no hscroot@172.16.50.253 'chhwres -m SVRP7780-09-SN060COAT -p
P780_09_Lpar1 -r mem -o a -q 33024 -w 32 2>&1' -->33024 means 32.25 GB
...
+testRG2:c1hmccmd[c1hmccexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-o TCPKeepAlive=no hscroot@172.16.50.253 'chhwres -m SVRP7780-09-SN060COAT -p
P780_09_Lpar1 -r proc -o a --procs 19 -w 32 2>&1 -->172.16.50.253 is HMCEP1
```

Note: We do not display the DLPAR and EPCoD operations in the query step in the previous examples.

6.13.2 Bringing one resource group offline when the primary HMC fails

After the RG is online, we bring the RG offline. During this process, we shut down HMCEP1 to see how PowerHA SystemMirror handles this situation.

The resource releasing process is shown in Figure 6-47.

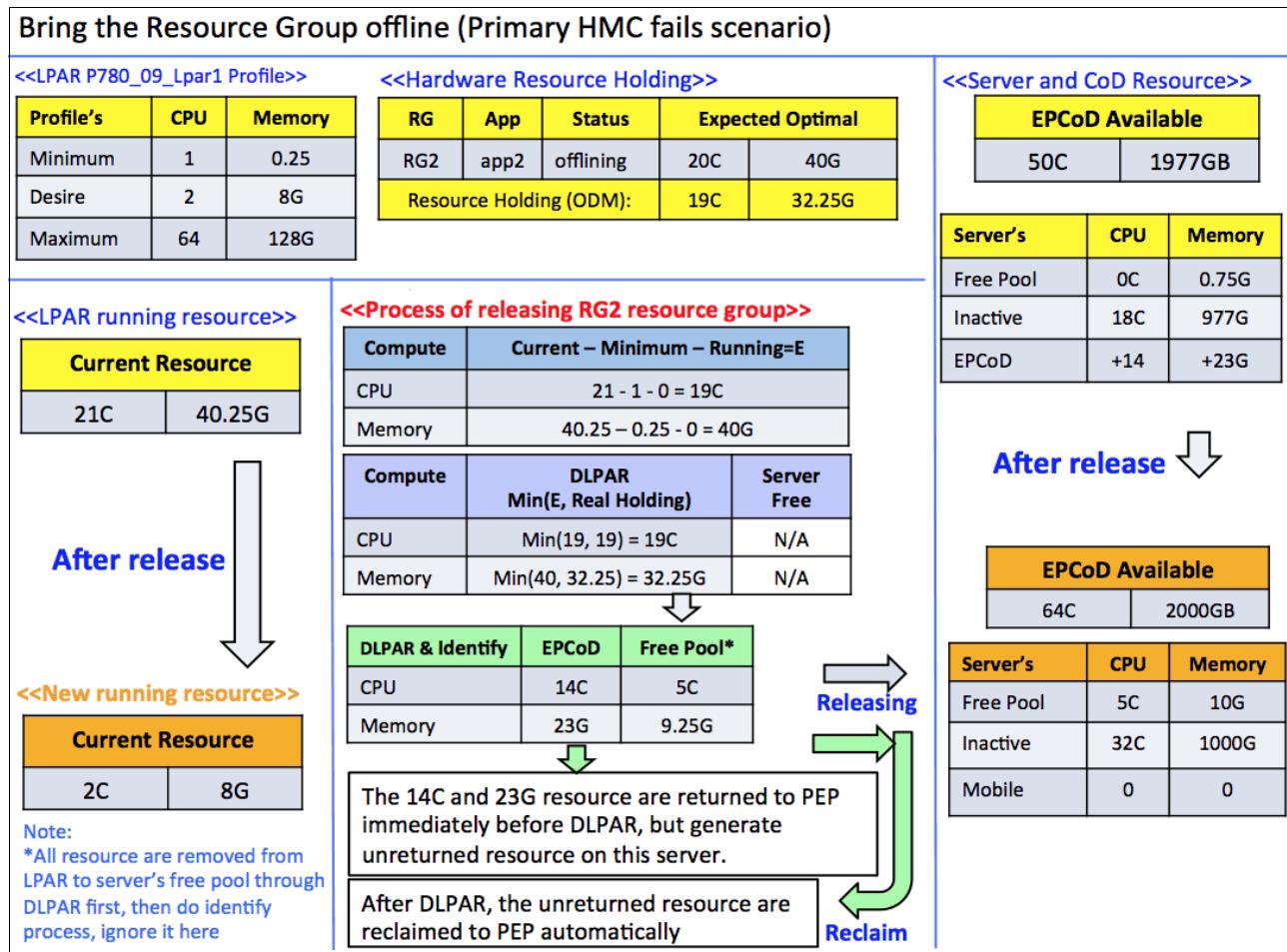


Figure 6-47 Bringing the resource group offline process

Section 6.6, “Introduction to resource acquisition” on page 175 introduces the four steps for PowerHA SystemMirror to acquire the resources. The following sections give a detailed description of the four steps.

Query step

In this step, PowerHA SystemMirror must query the server’s data and the EPCoD’s data.

To get the server’s information, PowerHA SystemMirror uses the default primary HMC (172.16.50.253, HMCEP1). At first, HMCEP1 is alive, and the operation succeeds. But after the HMCEP1 shutdown, the operation fails and PowerHA SystemMirror uses 172.16.50.254 as the primary HMC to continue. Example 6-35 shows the takeover process.

Example 6-35 HMC takeover process

```
+testRG2:c1hmccmd[get_local_hmc_list:815] g_hmc_list='172.16.50.253 172.16.50.254'  
--> default, the global HMC list is:172.16.50.253 is first, then 172.16.50.254  
...  
+testRG2:c1hmccmd[c1hmceexec:3512] ping -c 1 -w 3 172.16.50.253  
+testRG2:c1hmccmd[c1hmceexec:3512] 1> /dev/null 2>& 1  
+testRG2:c1hmccmd[c1hmceexec:3512] ping_output=''  
+testRG2:c1hmccmd[c1hmceexec:3513] ping_rc=1  
+testRG2:c1hmccmd[c1hmceexec:3514] (( 1 > 0 ))  
+testRG2:c1hmccmd[c1hmceexec:3516] : Cannot contact this HMC. Ask following HMC in list.  
--> after checking, confirm that 172.16.50.253 is unaccessible, then to find next HMC in the list  
...  
+testRG2:c1hmccmd[c1hmceexec:3510] : Try to ping the HMC at address 172.16.50.254.  
+testRG2:c1hmccmd[c1hmceexec:3512] ping -c 1 -w 3 172.16.50.254  
+testRG2:c1hmccmd[c1hmceexec:3512] 1> /dev/null 2>& 1  
+testRG2:c1hmccmd[c1hmceexec:3512] ping_output=''  
+testRG2:c1hmccmd[c1hmceexec:3513] ping_rc=0  
+testRG2:c1hmccmd[c1hmceexec:3514] (( 0 > 0 ))  
--> 172.16.50.254 is the next, so PowerHA SystemMirror check it  
...  
+testRG2:c1hmccmd[update_hmc_list:3312] g_hmc_list='172.16.50.254 172.16.50.253'  
--> it is accessible, change it as first HMC in global HMC list  
...  
+testRG2:c1hmccmd[c1hmceexec:3456] loop_hmc_list='172.16.50.254 172.16.50.253'  
--> global HMC list has been changed, following operation will use 172.16.50.254  
...  
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3 -o TCPKeepAlive=no hscroot@172.16.50.254 'lshmc -v 2>&1'  
--> start with 172.16.50.254 to do query operation  
...  
+testRG2:c1hmccmd[c1hmceexec:3618] 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@172.16.50.254 \'lscodpool -p 0019 --level sys --filter names=SVRP7780-09-SN060COAT -F inactive_mem:mobile_mem:unreturne d_mobile_mem:inactive_procs:mobile_procs:unreturned_mobile_procs 2>&1\''  
+t  
--> using 172.16.50.254 to query EPCoD information
```

Important: The process enables you to query EPCoD information from the backup HMC. However, any change operations must be done on the master HMC.

Compute step

This step does not require an HMC operation. For more information, see Figure 6-47 on page 232.

Identify and acquire step

After the identify step, there are some resources that must be released to EPCoD. Therefore, PowerHA SystemMirror returns the resource back to EPCoD immediately before the resource is removed from the LPAR. This generates an unreturned resource temporarily.

At this time, PowerHA SystemMirror checks whether the master HMC is available. If not, it switches to the backup HMC automatically. Example 6-36 shows the detailed process.

Example 6-36 The EPCoD master and backup HMC switch process

```
+testRG2:c1hmccmd[c1hmccexec:3388] cmd='chcodpool -p 0019 -m SVRP7780-09-SN060COAT
-r mem -o remove -q 23552 --force'
-->PowerHA SystemMirror try to do chcodpool operation
...
+testRG2:c1hmccmd[c1hmccexec:3401] : If working on an EPCoD Operation, we need
master
-->PowerHA SystemMirror want to check whether master HMC is accessible
...
ctionAttempts=3 -o TCPKeepAlive=no $'hscroot@172.16.50.254 \'lscodpool -p 0019
--level pool -F master_mc_name:backup_master_mc_name 2>&1\''
+testRG2:c1hmccmd[c1hmccexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-o TCPKeepAlive=no hscroot@172.16.50.254 'lscodpool -p 0019 --level pool -F
master_mc_name:backup_master_mc_name 2>&1'
+testRG2:c1hmccmd[c1hmccexec:1] LC_ALL=C
+testRG2:c1hmccmd[c1hmccexec:3415] res=HMCEP1:HMCEP2
-->Current HMC is 172.16.50.254, so PowerHA SystemMirror query current master and
backup HMC name from it. At this time, HMCEP1 is master and HMCEP2 is backup.
...
+testRG2:c1hmccmd[c1hmccexec:3512] ping -c 1 -w 3 HMCEP1
+testRG2:c1hmccmd[c1hmccexec:3512] 1> /dev/null 2>& 1
+testRG2:c1hmccmd[c1hmccexec:3512] ping_output=''
+testRG2:c1hmccmd[c1hmccexec:3513] ping_rc=1
+testRG2:c1hmccmd[c1hmccexec:3514] (( 1 > 0 ))
+testRG2:c1hmccmd[c1hmccexec:3516] : Cannot contact this HMC. Ask following HMC in
list.
+testRG2:c1hmccmd[c1hmccexec:3518] dspmsg scripts.cat -s 38 500 '%1$s: WARNING:
unable to ping HMC at address %2$s.\n' c1hmccmd HMCEP1
-->PowerHA SystemMirror try to ping HMCEP1, but fails
...
+testRG2:c1hmccmd[c1hmccexec:3510] : Try to ping the HMC at address HMCEP2.
+testRG2:c1hmccmd[c1hmccexec:3512] ping -c 1 -w 3 HMCEP2
+testRG2:c1hmccmd[c1hmccexec:3512] 1> /dev/null 2>& 1
+testRG2:c1hmccmd[c1hmccexec:3512] ping_output=''
+testRG2:c1hmccmd[c1hmccexec:3513] ping_rc=0
+testRG2:c1hmccmd[c1hmccexec:3514] (( 0 > 0 ))
-->PowerHA SystemMirror try to verify HMCEP2 and it is available
```

```

...
+testRG2:c1hmccmd[c1hmceexec:3527] : the hmc is the master_hmc
+testRG2:c1hmccmd[c1hmceexec:3529] (( g_epcod_modify_operation == 1 &&
loop_hmc_counter != 1 ))
+testRG2:c1hmccmd[c1hmceexec:3531] : If not, we need to change master_hmc, we also
try to
+testRG2:c1hmccmd[c1hmceexec:3532] : set a backup_master_hmc

+testRG2:c1hmccmd[c1hmceexec:3536] eval LC_ALL=C ssh -o StrictHostKeyChecking=no -o
LogLevel=quiet -o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o Conne
ctionAttempts=3 -o TCPKeepAlive=no $'hscroot@HMCEP2 \'chcodpool -p 0019 -o
setmaster --mc this 2>&1\''
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-
o TCPKeepAlive=no hscroot@HMCEP2 'chcodpool -p 0019 -o setmaster --mc this 2>&1'
+testRG2:c1hmccmd[c1hmceexec:1] LC_ALL=C
+testRG2:c1hmccmd[c1hmceexec:3536] out_str=''
+testRG2:c1hmccmd[c1hmceexec:3537] ssh_rc=0
-->PowerHA SystemMirror set backup HMC(HMCEP2) as master
...
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-
o TCPKeepAlive=no hscroot@HMCEP2 'chcodpool -p 0019 -o update -a
"backup_master_mc_name=HMCEP1" 2>&1'
+testRG2:c1hmccmd[c1hmceexec:1] LC_ALL=C
+testRG2:c1hmccmd[c1hmceexec:3722] out_str='HSCL90E9 Management console HMCEP1was
not found.'
-->PowerHA SystemMirror also try to set HMCEP1 as backup, but it fails because
HMCEP1 is shut down at this time
...
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-
o TCPKeepAlive=no hscroot@HMCEP2 'chcodpool -p 0019 -m SVRP7780-09-SN060COAT -r
mem -o remove -q 23552 --force 2>&1'
+testRG2:c1hmccmd[c1hmceexec:1] LC_ALL=C
-->PowerHA SystemMirror do the force release for memory resource
...
+testRG2:c1hmccmd[c1hmceexec:1] ssh -o StrictHostKeyChecking=no -o LogLevel=quiet
-o AddressFamily=any -o BatchMode=yes -o ConnectTimeout=3 -o ConnectionAttempts=3
-
o TCPKeepAlive=no hscroot@HMCEP2 'chcodpool -p 0019 -m SVRP7780-09-SN060COAT -r
proc -o remove -q 14 --force 2>&1'
-->PowerHA SystemMirror do the force release for CPU resource

```

Example 6-37 shows the update that is performed from the EPCoD view.

Example 6-37 EPCoD status change during the takeover operation

```

hscroot@HMCEP2:~> lscodpool -p 0019 --level pool
name=0019,id=0019,state=In
compliance,sequence_num=5,master_mc_name=HMCEP1,master_mc_mtms=V017-ffe*d33e8a1,ba
ckup_master_mc_name=HMCEP2,backup_master_mc_mtms=V017-f93*ba3e3aa,mobile_procs=64,

```

```

avail_mobile_procs=50,unreturned_mobile_procs=0,mobile_mem=2048000,avail_mobile_me
m=2024448,unreturned_mobile_mem=0
--> There are 14CPU(64-50) and 23 GB((2048000-2024448)/1024) has assigned to
P780D_09.
hscroot@HMCEP2:~> lscodpool -p 0019 --level sys
name=SVRP7780-10-SN061949T,mtms=9179-MHD*061949T,mobile_procs=0,non_mobile_procs=3
2,unreturned_mobile_procs=0,inactive_procs=32,installed_procs=64,mobile_mem=0,non_
mobile_mem=1073152,unreturned_mobile_mem=0,inactive_mem=1024000,installed_mem=2097
152
name=SVRP7780-09-SN060C0AT,mtms=9179-MHD*060C0AT,mobile_procs=14,non_mobile_procs=
32,unreturned_mobile_procs=0,inactive_procs=18,installed_procs=64,mobile_mem=23552
,non_mobile_mem=1073152,unreturned_mobile_mem=0,inactive_mem=1000448,installed_mem=
=2097152
--> Show the information from server level report
...
hscroot@HMCEP2:~> lscodpool -p 0019 --level pool
name=0019,id=0019,state=unavailable,sequence_num=5,master_mc_name=HMCEP1,master_mc_
_mtms=V017-ffe*d33e8a1,backup_master_mc_name=HMCEP2,backup_master_mc_mtms=V017-f93
*ba3e3aa,mobile_procs=unavailable,avail_mobile_procs=unavailable,unreturned_mobile_
procs=unavailable,mobile_mem=unavailable,avail_mobile_mem=unavailable,unreturned_
mobile_mem=unavailable
--> After HMCEP1 shutdown, the EPCoD's status is changed to 'unavailable'
...
hscroot@HMCEP2:~> lscodpool -p 0019 --level pool
name=0019,id=0019,state=In
compliance,sequence_num=5,master_mc_name=HMCEP2,master_mc_mtms=V017-f93*ba3e3aa,ba
ckup_master_mc_mtms=none,mobile_procs=64,avail_mobile_procs=50,unreturned_mobile_p
rocs=0,mobile_mem=2048000,avail_mobile_mem=2024448,unreturned_mobile_mem=0
--> After PowerHA SystemMirror run 'chcodpool -p 0019 -o setmaster --mc this' on
HMCEP2, the master HMC is changed and status is changed to 'In compliance'
....
hscroot@HMCEP2:~> lscodpool -p 0019 --level sys
name=SVRP7780-10-SN061949T,mtms=9179-MHD*061949T,mobile_procs=0,non_mobile_procs=3
2,unreturned_mobile_procs=0,inactive_procs=32,installed_procs=64,mobile_mem=0,non_
mobile_mem=1073152,unreturned_mobile_mem=0,inactive_mem=1024000,installed_mem=2097
152
name=SVRP7780-09-SN060C0AT,mtms=9179-MHD*060C0AT,mobile_procs=0,non_mobile_procs=
32,unreturned_mobile_procs=14,inactive_procs=18,installed_procs=64,mobile_mem=0,non_
_mobile_mem=1073152,unreturned_mobile_mem=23552,inactive_mem=1000448,installed_mem=
=2097152
--> After PowerHA SystemMirror forcibly releases resource, unreturned resource is
generated
...
hscroot@HMCEP2:~> lscodpool -p 0019 --level pool
name=0019,id=0019,state=Approaching out of compliance (within server grace
period),sequence_num=5,master_mc_name=HMCEP2,master_mc_mtms=V017-f93*ba3e3aa,backu
p_master_mc_mtms=none,mobile_procs=64,avail_mobile_procs=64,unreturned_mobile_pro
c=14,mobile_mem=2048000,avail_mobile_mem=2048000,unreturned_mobile_mem=23553
--> At this time, the resource has returned to EPCoD and can be used by other
servers.
...

```

When PowerHA SystemMirror completes the previous steps, it raises an asynchronous process to remove the resources from P780_09_Lpar1 by using DLPAR. The resources include 19 CPUs and 32.25 GB of memory.

After the DLPAR operation, the unreturned resource is reclaimed automatically, and the EPCoD status is changed to In compliance, as shown in Example 6-38.

Example 6-38 EPCoD status that is restored after the DLPAR operation completes

```
hscroot@HMCEP1:~> lscodpool -p 0019 --level pool
name=0019,id=0019,state=In
compliance,sequence_num=5,master_mc_name=HMCEP1,master_mc_mtms=V017-ffe*d33e8a1,ba
ckup_master_mc_name=HMCEP2,backup_master_mc_mtms=V017-f93*ba3e3aa,mobile_procs=64,
avail_mobile_procs=64,unreturned_mobile_procs=0,mobile_mem=2048000,avail_mobile_me
m=2048000,unreturned_mobile_mem=0
```

6.13.3 Testing summary

This scenario introduced how PowerHA SystemMirror performs HMC takeovers when the primary HMC fails. This is an automatic process and has no impact on your environment.

6.14 Managing, monitoring, and troubleshooting

This section introduces some tools to manage, monitor, and troubleshoot a ROHA cluster.

6.14.1 The clmgr interface to manage Resource Optimized High Availability

SMIT relies on the **clmgr** command to perform the configuration that is related to ROHA.

HMC configuration

The following examples show how to configure HMC with the **clmgr** command.

Query/Add/Modify/Delete

Example 6-39 shows how to query, add, modify, and delete HMC with the **clmgr** command.

Example 6-39 Query/Add/Modify/Delete HMC with the clmgr command

```
# clmgr query hmc -h
clmgr query hmc [<HMC>[,<HMC#2>,...]]
```



```
# clmgr -v query hmc
NAME="r1r9sdmc.austin.ibm.com"
TIMEOUT="-1"
RETRY_COUNT="8"
RETRY_DELAY="-1"
NODES=clio1,clio2
SITES=site1
```



```
# clmgr add hmc -h
clmgr add hmc <HMC> \
    [ TIMEOUT={<#>} ] \
    [ RETRY_COUNT={<#>} ] \
    [ RETRY_DELAY={<#>} ] \
    [ NODES=<node>[,<node#2>,...]> ] \
    [ SITES=<site>[,<site#2>,...]> ] \
    [ CHECK_HMC={<yes>|<no>} ]
```

```

# clmgr modify hmc -h
clmgr modify hmc <HMC> \
    [ TIMEOUT={<#>} ] \
    [ RETRY_COUNT={<#>} ] \
    [ RETRY_DELAY={<#>} ] \
    [ NODES=<node>[,<node#2>,...]> ] \
    [ SITES=<site>[,<site#2>,...]> ] \
    [ CHECK_HMC={<yes>|<no>} ]

```

```

# clmgr delete hmc -h
clmgr delete hmc {<HMC>[,<HMC#2>,...] | ALL}

```

Query/Modify a node with the list of associated HMCs

Example 6-40 shows how to query and modify a node with the list of associated HMCs.

Example 6-40 Query/Modify node with a list of associated HMCs with the clmgr command

```

# clmgr query node -h
clmgr query node {<node>|LOCAL} [,<node#2>,...]

# clmgr -v query node
NAME="rar1m31"
...
HMCS="r1r9sdmc.austin.ibm.com cuodhmc.austin.ibm.com"

# clmgr modify node -h
clmgr modify node <NODE> \
    ... \
    [ HMCS=<sorted_hmc_list> ]

```

Query/Modify a site with the list of associated HMCs

Example 6-41 shows how to query and modify the site with a list of associated HMCs with the **clmgr** command.

Example 6-41 Query/Modify a site with a list of the associated HMCs with the clmgr command

```

# clmgr query site -h
clmgr query site [<site> [,<site#2>,...]]

# clmgr -v query site
NAME="site1"
...
HMCS="r1r9sdmc.austin.ibm.com cuodhmc.austin.ibm.com"

# clmgr modify site -h
clmgr modify site <SITE> \
    ... \
    [ HMCS =<sorted_hmc_list> ]

```

Query/Modify a cluster with the default HMC tunables

Example 6-42 on page 239 shows how to query and modify the cluster with the default HMC tunables.

Example 6-42 Query/Modify a cluster with the default HMC tunables with the clmgr command

```
# clmgr query cluster -h
clmgr query cluster [ ALL | {CORE,SECURITY,SPLIT-MERGE,HMC,ROHA} ]

# clmgr query cluster hmc
DEFAULT_HMC_TIMEOUT="10"
DEFAULT_HMC_RETRY_COUNT="5"
DEFAULT_HMC_RETRY_DELAY="10"
DEFAULT_HMCS_LIST="r1r9sdmc.austin.ibm.com cuodhmc.austin.ibm.com"

# clmgr manage cluster hmc -h
clmgr manage cluster hmc \
    [ DEFAULT_HMC_TIMEOUT=# ] \
    [ DEFAULT_HMC_RETRY_COUNT=# ] \
    [ DEFAULT_HMC_RETRY_DELAY=# ] \
    [ DEFAULT_HMCS_LIST=<new_hmcs_list> ]
```

Hardware resource provisioning

SMIT relies on the **clmgr** command to list or query current values of the hardware resource provisioning and to add/modify/delete the HACMPserver ODM data structure, as shown in Example 6-43.

Example 6-43 Hardware resource provisioning configuration with the clmgr command

```
# clmgr query cod -h
clmgr query cod [<APP>[,<APP#2>,...]]]

# clmgr -v query cod
NAME="appli1_APPCON_A"
USE_DESIRED=No
OPTIMAL_MEM="4"
OPTIMAL_CPU="3"
OPTIMAL_PU="2.5"
OPTIMAL_PV="3.0"

# clmgr add cod -h
clmgr add cod <APPCTRL> \
    [ USE_DESIRED =<Yes|No> ] \
    [ OPTIMAL_MEM=# ] \
    [ OPTIMAL_CPU=# ] \
    [ OPTIMAL_PU=#.# ] \
    [ OPTIMAL_PV=#.# ]

# clmgr modify cod -h
clmgr modify cod <APPCTRL> \
    [ USE_DESIRED =<Yes|No> ] \
    [ OPTIMAL_MEM=# ] \
    [ OPTIMAL_CPU=# ] \
    [ OPTIMAL_PU=#.# ] \
    [ OPTIMAL_PV=# ]

# clmgr delete cod -h
clmgr delete cod {<APPCTRL> | ALL}
```

Cluster tunables

SMIT relies on the `clmgr` command to query or modify cluster CoD tunables, as shown in Example 6-44.

Example 6-44 Cluster-wide tunables configuration with the clmgr command

```
# clmgr query cluster -h
clmgr query cluster [ ALL | {CORE,SECURITY,SPLIT-MERGE,HMC,ROHA} ]

# clmgr query cluster roha
ALWAYS_START_RG="no"
ADJUST_SPP_SIZE="yes"
FORCE_SYNC_RELEASE="no"
AGREE_TO_COD_COSTS="no"
COD_ONOFF_DAYS="30"
RESOURCE_ALLOCATION_ORDER="free_pool_first"

# clmgr manage cluster roha -h
clmgr manage cluster roha \
    [ ALWAYS_START_RG={yes|no} ] \
    [ ADJUST_SPP_SIZE={yes|no} ] \
    [ FORCE_SYNC_RELEASE={yes|no} ] \
    [ AGREE_TO_COD_COSTS={yes|no} ] \
    [ COD_ONOFF_DAYS=<new_number_of_days> ] \
```

Important: There is a new variable that is shown in Example 6-44:

`RESOURCE_ALLOCATION_ORDER`

The resource allocation order specifies the order in which resources are allocated. The resources are released in the reverse order in which they are allocated. The default value for this field is Free Pool First.

Select Free Pool First to acquire resources from the free pool. If the amount of resources in the free pool is insufficient, PowerHA SystemMirror first requests more resources from the Enterprise pool and then from the CoD pool.

Select Enterprise Pool First to acquire the resources from the Enterprise pool. If the amount of resources in the CoD pool is insufficient, PowerHA SystemMirror first requests more resources from the free pool and then from the CoD pool.

6.14.2 Changing the DLPAR and CoD resources dynamically

You can change the DLPAR and CoD resource requirements for application controllers without stopping the cluster services. Synchronize the cluster after making the changes.

The new configuration is not reflected until the next event that causes the application (hence the RG) to be released and reacquired on another node. A change in the resource requirements for CPUs, memory, or both does not cause the recalculation of the DLPAR resources. PowerHA SystemMirror does not stop and restart the application controllers solely for the purpose of making the application provisioning changes.

If another dynamic reconfiguration change causes the RGs to be released and reacquired, the new resource requirements for DLPAR and CoD are used at the end of this dynamic reconfiguration event.

6.14.3 View the Resource Optimized High Availability report

The **c1mgr view report roha** command is intended to query all the ROHA data so that a report and a summary can be presented to the user.

The output of this command includes the following sections:

- ▶ CEC name
- ▶ LPAR name
- ▶ LPAR profile (min, desired, and max)
- ▶ LPAR processing mode
- ▶ If shared (capped or uncapped, SPP name, and SPP size)
- ▶ LPAR current level of resources (mem, cpu, and pu)
- ▶ Number and names of AC and optimal level of resources, and the sum of them
- ▶ Release mode (sync/async), which is computed at release time
- ▶ All On/Off CoD information of the CECs
- ▶ All EPCoD information of the CECs

There is an example of the report in 6.10.4, “Showing the Resource Optimized High Availability configuration” on page 217.

6.14.4 Troubleshooting DLPAR and CoD operations

This section provides a brief troubleshooting of the DLPAR and CoD operations.

Log files

There are several log files that you can use to track the ROHA operation process.

Logs for verification

In the process of Verify and Synchronize Cluster Configuration, there are some log files that are generated in the `/var/hacmp/c1verify` directory. The `c1verify.log` and the `ver_collect_dlpars.log` files are useful for debugging if the process fails. For example, after performing the process, there is some error information appearing in the console output (`/smrit.log`), as shown in Example 6-45.

Example 6-45 Error information about console or /smrit.log

```
WARNING: At the time of verification, node ITSO_S2Node1 would not have been able to acquire
         sufficient resources to run Resource Group(s) RG1 (multiple Resource Groups
         in case of node collocation). Please note that the amount of resources and
         CoD resources available at the time of verification may be different from
         the amount available at the time of an actual acquisition of resources.
         Reason : 708.00 GB of memory that is needed will exceed LPAR maximum of 160.00 GB.
12.50 Processing Unit(s) needed will exceed LPAR maximum of 9.00 Processing Unit(s).
ERROR: At the time of verification, no node (out of 2) was able to acquire
        sufficient resources to run Resource Group(s) RG1
```

You can get detailed information to help you identify the errors' root causes from the `c1verify.log` and the `ver_collect_dlpars.log` files, as shown in Example 6-46.

Example 6-46 Detailed information in ver_collect_dlpars.log

```
[ROHALOG:2490918:(19.127)] c1managerroha: ERROR: 708.00 GB of memory that is needed will
         exceed LPAR maximum of 160.00 GB.
[ROHALOG:2490918:(19.130)] ===== Compute ROHA Memory =====
[ROHALOG:2490918:(19.133)] minimal + optimal + running = total <=> current <=> maximum
```

```

[ROHALOG:2490918:(19.137)] 8.00 + 700.00 + 0.00 = 708.00 <=> 32.00 <=> 160.00 : =>
0.00 GB
[ROHALOG:2490918:(19.140)] ===== End =====
[ROHALOG:2490918:(19.207)] clmanageroha: ERROR: 12.50 Processing Unit(s) needed will exceed
LPAR maximum of 9.00 Processing Unit(s).
[ROHALOG:2490918:(19.212)] === Compute ROHA PU(s)/VP(s) ==
[ROHALOG:2490918:(19.214)] minimal + optimal + running = total <=> current <=> maximum
[ROHALOG:2490918:(19.217)] 1 + 12 + 0 = 13 <=> 3 <=> 18 : =>
0 Virtual Processor(s)
[ROHALOG:2490918:(19.220)] minimal + optimal + running = total <=> current <=> maximum
[ROHALOG:2490918:(19.223)] 0.50 + 12.00 + 0.00 = 12.50 <=> 1.50 <=> 9.00 : =>
0.00 Processing Unit(s)
[ROHALOG:2490918:(19.227)] ===== End =====
[ROHALOG:2490918:(19.231)] INFO: received error code 21.
[ROHALOG:2490918:(19.233)] No or no more reassessment.
[ROHALOG:2490918:(19.241)] An error occurred while performing acquire operation.

```

PowerHA SystemMirror simulates the resource acquisition process based on the current configuration and generates the log in the `ver_collect_d1par.log` file.

Logs for resource group online and offline

During the process of resource online or offline, the `hacmp.out` and the `async_release.log` logs are useful for monitoring or debugging. In some RG offline scenarios, the DLPAR remove operation is a synchronous process. In this case, PowerHA SystemMirror generates the DLPAR operation logs in the `async_release.log` file. In a synchronous process, only `hacmp.out` is used.

AIX errpt output

Sometimes, the DLPAR operation fails, and AIX generates some errors that are found in the `errpt` output, as shown in Example 6-47.

Example 6-47 The errpt error report

252D3145	1109140415 T S mem	DR failed by reconfig handler
47DCD753	1109140415 T S PROBEVUE	DR: memory remove failed by ProbeVue rec

You can identify the root cause of the failure by using this information.

HMC commands

You can use the following commands on the HMC to do some monitor or maintenance. For a detailed description of the commands, see the `man` page for the HMC.

The lshwres command

This command shows the LPAR minimum, LPAR maximum, the total amount of memory, and the number of CPUs that are currently allocated to the LPAR.

The issyscfg command

This command verifies that the LPAR node is DLPAR capable.

The chhwres command

This command runs the DLPAR operations on the HMC outside of PowerHA SystemMirror to manually change the LPAR minimum, LPAR maximum, and LPAR required values for the LPAR. This might be necessary if PowerHA SystemMirror issues an error or a warning, during the verification process, if you requested DLPAR and CoD resources in PowerHA SystemMirror.

The lscod command

This command views the system CoD of the current configuration.

The chcod command

This command runs the CoD operations on the HMC outside of PowerHA SystemMirror and manually changes the Trial CoD, On/Off CoD, and so on, of the activated resources. This command is necessary if PowerHA SystemMirror issues an error or a warning during the verification process, or if you want to use DLPAR and On/Off CoD resources in PowerHA SystemMirror.

The lscodpool command

This command shows the system Enterprise Pool current configuration.

The chcodpool command

This command runs the Enterprise Pool CoD operations on the HMC outside of PowerHA SystemMirror and manually changes the Enterprise pool capacity resources. This command is necessary if PowerHA SystemMirror issues an error or a warning during the verification process, or if you want to use DLPAR, On/Off CoD, or EPCoD resources in PowerHA SystemMirror.



Geographic Logical Volume Manager configuration assistant

This chapter covers the following topics:

- ▶ Introduction:
 - Geographical Logical Volume Manager
 - GLVM configuration assistant
- ▶ Prerequisites
- ▶ Using the GLVM wizard:
 - Test environment overview
 - Synchronous configuration
 - Asynchronous configuration

7.1 Introduction

The following sections give an introduction to Geographical Logical Volume Manager (GLVM) and the configuration assistant. Additional details, including planning and implementing, can be found in the base documentation available at [IBM Knowledge Center](#).

7.1.1 Geographical Logical Volume Manager

GLVM provides an IP-based data mirroring capability for the data at geographically separated sites. It protects the data against total site failure by remote mirroring, and supports unlimited distance between participating sites.

GLVM for PowerHA SystemMirror Enterprise Edition provides automated disaster recovery capability by using the AIX Logical Volume Manager (LVM) and GLVM subsystems to create volume groups (VGs) and logical volumes that span across two geographically separated sites.

You can use the GLVM technology as a stand-alone method, or use it in combination with PowerHA SystemMirror Enterprise Edition.

The software increases data availability by providing continuing service during hardware or software outages (or both), planned or unplanned, for a two-site cluster. The distance between sites can be unlimited, and both sites can access the mirrored VGs serially over IP-based networks.

Also, it enables your business application to continue running at the takeover system at a remote site while the failed system is recovering from a disaster or a planned outage.

The software takes advantage of the following software components to reduce downtime and recovery time during disaster recovery:

- ▶ AIX LVM subsystem and GLVM
- ▶ TCP/IP subsystem
- ▶ PowerHA SystemMirror for AIX cluster management

Definitions and concepts

This section defines the basic concepts of GLVM:

- ▶ Remote physical volume (RPV)

A pseudo-device driver that provides access to the remote disks as though they were locally attached. The remote system must be connected by way of the Internet Protocol network. The distance between the sites is limited by the latency and bandwidth of the connecting networks.

The RPV consists of two parts:

- RPV Client:

This is a pseudo-device driver that runs on the local machine and allows the AIX LVM to access RPVs as though they were local. The RPV clients are seen as hdisk devices, which are logical representations of the RPV.

The RPV client device driver appears as an ordinary disk device. For example, the RPV client device hdisk8 has all its I/O directed to the remote RPV server. It also has no knowledge at all about the nodes, networks, and so on.

When configuring the RPV client, the following details are defined:

- The IP address of the RPV server.
- The local IP address (defines the network to use).
- The timeout. This field is primarily for the stand-alone GLVM option, as PowerHA overwrites this field with the cluster's config_too_long time. In a PowerHA cluster, this is the worst case scenario, as PowerHA detects problems with the remote node well before then.

The SMIT fast path to configure the RPV clients is **smitty rpvclient**.

– RPV server

The RPV server runs on the remote machine, one for each physical volume that is being replicated. The RPV server can listen to a number of remote RPV clients on different hosts to handle failover.

The RPV server is an instance of the kernel extension of the RPV device driver with names such as rpvserver0, and is not an actual physical device.

When configuring the RPV server, the following items are defined:

- The PVID of the local physical volume.
- The IP addresses of the RPV clients (comma separated).

– Geographically mirrored volume group (GMVG)

A VG that consists of local PVs and RPVs. Strict rules apply to GMVGs to ensure that you have a complete copy of the mirror at each site. For this reason, the superstrict allocation policy is required for each logical volume in a GMVG.

PowerHA SystemMirror Enterprise Edition also expects each logical volume in a GMVG to be mirrored and, for asynchronous replication, requires the use of AIX mirror pools. GMVGs are managed by PowerHA and recognized as a separate class of resource (GMVG Replicated Resources), so they have their own events. PowerHA verification issued a warning if there are resource groups (RGs) that contain GMVG resources that do not have the forced varyon flag set and if quorum is not disabled.

The SMIT fast path to configure the RPV servers is **smitty rpvserver**.

PowerHA enforces the requirement that each physical volume that is part of a VG with RPV clients has the reverse relationship defined. This, at a minimum, means that every GMVG consists of two physical volumes on each site. One disk is locally attached, and the other is a logical representation of the RPV.

– GLVM utilities

GLVM provides SMIT menus to create the GMVGs and the logical volumes. Although not required because they perform the same function as the equivalent SMIT menus in the background, they do control the location of the logical volumes to ensure proper placement of mirror copies. If you use the standard commands to configure your GMVGs, use the GLVM verification utility.

– Network types:

XD_data

Network that can be used for data replication only. A maximum of four XD_data networks can be defined. Etherchannel is supported for this network type. This network supports adapter swap, but not failover to another node. Heartbeat packets are also sent over this network.

XD_ip

An IP-based network that is used for participation in heartbeating and client communication.

- Mirror pools:

Mirror pools make it possible to divide the physical volumes of a VG into separate pools.

A mirror pool is made up of one or more physical volumes. Each physical volume can belong to only one mirror pool at a time. When creating a logical volume, each copy of the logical volume being created can be assigned to a mirror pool. Logical volume copies that are assigned to a mirror pool allocates only partitions from the physical volumes in that mirror pool, which provides the ability to restrict the disks that a logical volume copy can use.

Without mirror pools, the only way to restrict which physical volume is used for allocation when creating or extending a logical volume is to use a map file. Thus, using mirror pools greatly simplify this process. Think of mirror pools as an operating system level feature similar to storage consistency groups that are used when replicating data.

Although mirror pools are an AIX and not a GLVM-specific component, it is a preferred practice to use them in all GLVM configurations. However, they are required only when configuring asynchronous mode of GLVM.

- aio_cache logical volumes

An aio_cache is a special type of logical volume that stores write requests locally while it waits for the data to be written to a remote disk. The size of this logical volume dictates how far behind the data is allowed to be between the two sites. There is one defined at each site and they are *not* mirrored. Similar to data volumes, these volumes must be protected locally, usually by some form of RAID.

GLVM example

Figure 7-1 on page 249 shows a relatively basic two-site GLVM implementation. It consists of only one node at each site, although PowerHA does support multiple nodes within a site.

The New York site is considered the primary site because its node primarily hosts RPV clients. The Texas site is the standby site because it primarily hosts RPV servers. However, each site contains both RPV servers and clients based on where the resources are running.

Each site has two data disk volumes that are physically associated with the site node. In this case, the disks are hdisk1 and hdisk2 at both sites. However, the hdisk names do not need to match across sites. These two disks are also configured as RPV servers on each node. In turn, these are logically linked to the RPV clients at the opposite site. This configuration creates two additional pseudo-device disks that are known as hdisk3 and hdisk4. Their associated disk definitions clearly state that they are RPV clients and not real physical disks.

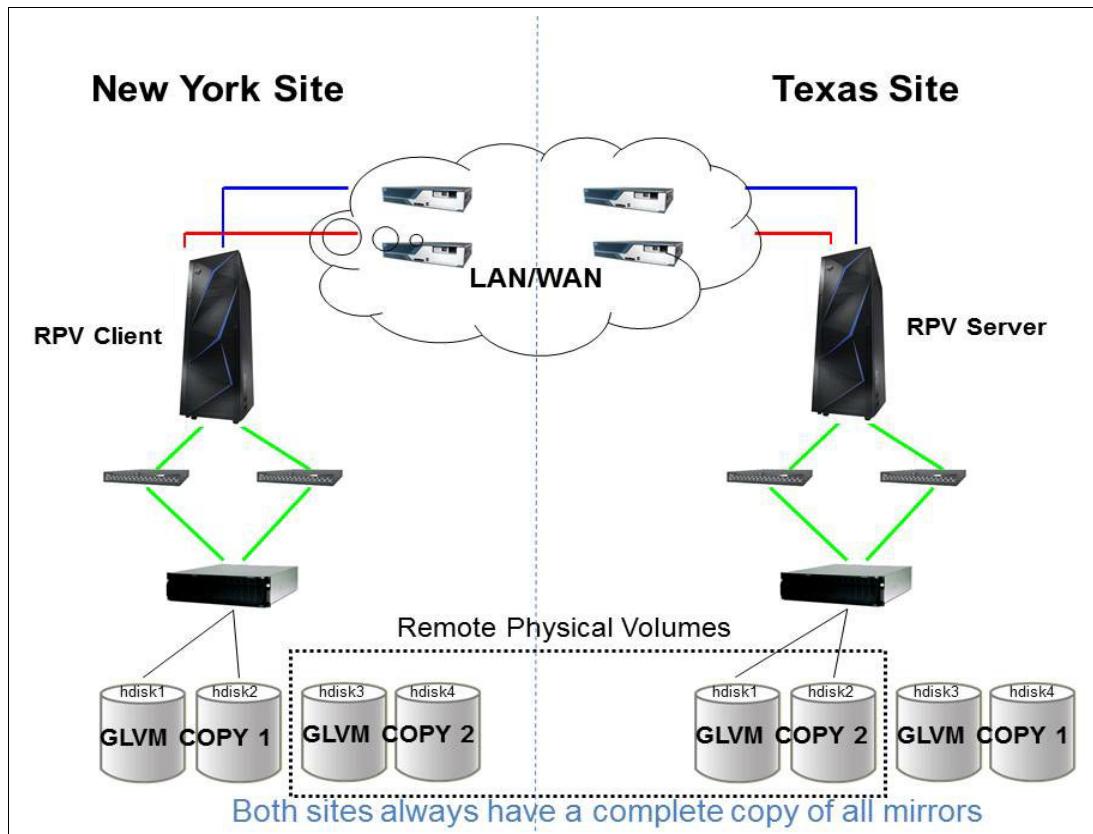


Figure 7-1 GLVM example configuration

7.1.2 GLVM configuration assistant

The GLVM configuration assistant was first introduced in PowerHA SystemMirror for AIX Enterprise Edition 6.1.0 primarily for asynchronous mode. It has been continuously enhanced over its release cycle and also includes support for synchronous mode. It is also often referred to as the *GLVM wizard*. The idea of the GLVM wizard is to streamline an otherwise cumbersome set of procedures down to minimal inputs:

- ▶ It takes the name of the nodes from both sites.
- ▶ It prompts for the selection of PVIDs to be mirrored on each site.
- ▶ When configuring async GLVM, it also prompts for the size of the aio_cache.

Given this information, the GLVM wizard configures all of the following items:

- ▶ GMVGs.
- ▶ RPV servers.
- ▶ RPV clients.
- ▶ Mirror pools.
- ▶ Resource Group.
- ▶ Synchronizes the cluster.

The GMVG is created as a scalable VG. It also activates the rpvserver at the remote site and the rpvcclient on the local site and leaves the VG active. The node upon which the GLVM wizard is run becomes the primary node, and is considered the local site. The RG is created with the key settings that are shown in Example 7-1.

Example 7-1 GLVM wizard resource group settings

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
Volume Groups	asyncglvm
Use forced varyon for volume groups, if necessary	true
GMVG Replicated Resources	asyncglvm

The GMVG does *not* do the following actions:

- ▶ Create any data-specific logical volumes or file systems within the GMVGs.
- ▶ Add any other resources into the RG (for example, service IPs and application controllers).
- ▶ Work for more than one GMVG.

This process can be used for the first GMVG, but additional GMVGs must be manually created and added into an RG.

7.2 Prerequisites

Before you use the GLVM wizard, complete the following prerequisites:

- ▶ Additional files from the PowerHA SystemMirror Enterprise Edition media:
 - cluster.xd.base
 - cluster.xd.glvm
 - cluster.xd.license
 - glvm.rpv.client
 - glvm.rpv.server
- ▶ A linked cluster is configured with sites.
- ▶ A repository disk is defined at each site.
- ▶ The verification and synchronization process completes successfully on the cluster.
- ▶ XD_data networks with persistent IP labels are defined on the cluster.
- ▶ The network communication between the local site and remote site is working.
- ▶ All PowerHA SystemMirror services are active on both nodes in the cluster.
- ▶ The /etc/hosts file on both sites contains all of the host IP, service IP, and persistent IP labels that you want to use in the GLVM configuration.
- ▶ The remote site must have enough free disks and enough free space on those disks to support all of the local site VGs that are created for geographical mirroring.

7.3 Using the GLVM wizard

This section goes through an example on our test cluster of using the GLVM wizard for both synchronous and asynchronous configurations.

7.3.1 Test environment overview

For the following example shown in Example 7-2 on page 252, we are using a two-node cluster with nodes *Jess* and *Ellie*. Our test configuration consists of two sites, New York and Chicago, along with the following hardware and software (Figure 7-2):

- ▶ Two POWER8 S814 with firmware 850
- ▶ HMC 850
- ▶ AIX 7.2.0 SP2
- ▶ PowerHA SystemMirror for AIX Enterprise Edition 7.2.1
- ▶ Two Storwize V7000 v7.6.1.1, one at each site for each S814
- ▶ Two IP networks, one public Ethernet, and one xd_data network for GLVM traffic

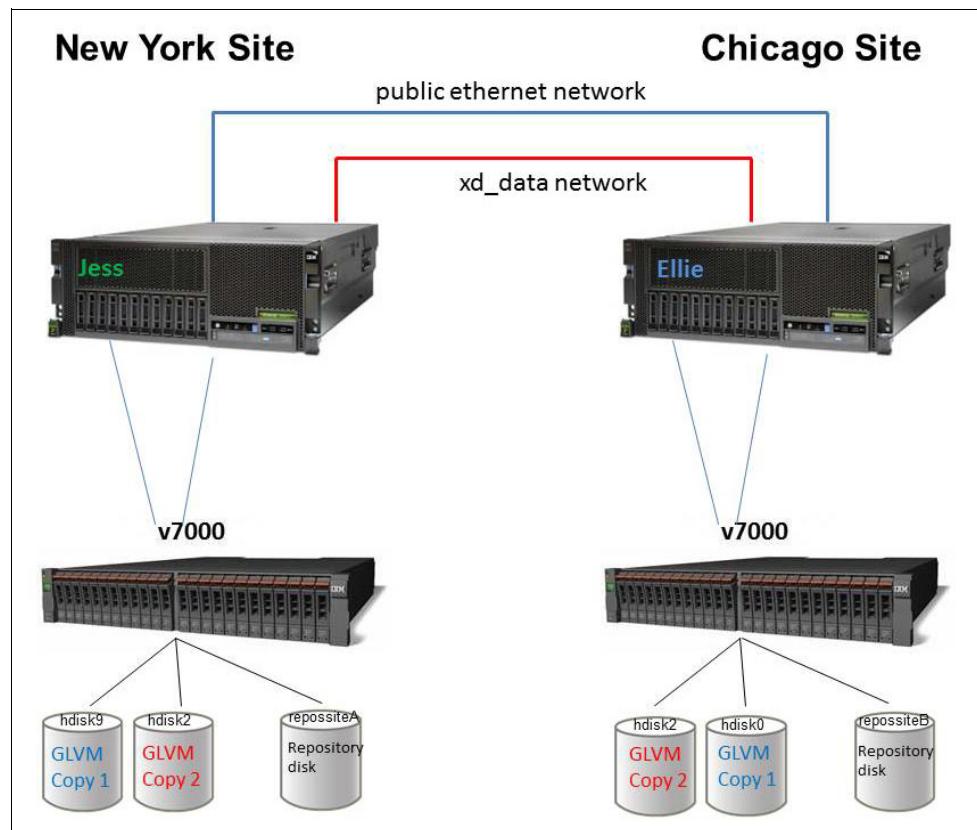


Figure 7-2 GLVM test cluster

7.3.2 Synchronous configuration

Before attempting to use the GLVM wizard, all the prerequisites that are listed in 7.2, “Prerequisites” on page 250 must be complete. Our scenario is a basic two-site configuration, with one node at each site, and an XD_data network with persistent alias defined in the configuration, as shown in Example 7-2.

Example 7-2 Base GLVM cluster topology

```
# cltopinfo
Cluster Name: GLVMdemocluster
Cluster Type: Linked
Heartbeat Type: Unicast
Repository Disks:
    Site 1 (NewYork@Jess): repositeA
    Site 2 (Chicago@Ellie): repositeB
Cluster Nodes:
    Site 1 (NewYork):
        Jess
    Site 2 (Chicago):
        Ellie

#cllsif
Adapter      Type      Network     Net Type   Attribute  Node
Jess         boot      net_ether_01 ether      public    Jess
Jess_glvm   boot      net_ether_02 XD_data   public    Jess
Jess_glvm   boot      net_ether_02 XD_data   public    Jess
Ellie        boot      net_ether_01 ether      public    Ellie
Ellie_glvm  boot      net_ether_02 XD_data   public    Ellie
Ellie_glvm_pers  persistent net_ether_02 XD_data   public    Ellie
```

Run **smitty sysmirror** and select **Cluster Applications and Resources → Make Applications Highly Available (Use Smart Assists) → GLVM Configuration Assistant → Configure Asynchronous GMVG**.

The menu that is shown in Figure 7-3 opens. If not, then the previously mentioned prerequisites have not been met, and you see a similar message as shown in Figure 7-4 on page 253.

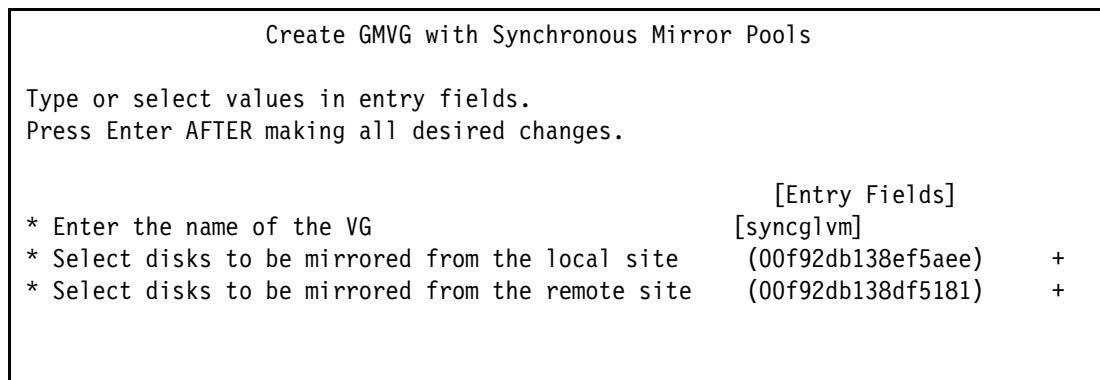


Figure 7-3 Synchronous GLVM wizard menu

COMMAND STATUS		
Command: OK	stdout: yes	stderr: no
Before command completion, additional instructions may appear below.		
No nodes are currently defined for the cluster.		
Define at least one node, and ideally all nodes, prior to defining the repository disk/disks and cluster IP address. It is important that all nodes in the cluster have access to the repository disk or respective repository disks(in case of a linked cluster) and can be reached via the cluster IP addresses, therefore you should define the nodes in the cluster first		

Figure 7-4 Synchronous GLVM prerequisites not met

Enter the field values as follows:

Enter the Name of the VG

Enter the name of the VG that you want to create as a geographically mirrored VG. If the RG is created by using the GLVM Configuration Assistant, the VG name is appended with _RG. For example, if the VG name is syncglvmvg, the RG name is syncglvmvg_RG.

Select disks to be mirrored from the local site

Press F4 to display a list of available disks. Press F7 to select the disks that you want to geographically mirror from the local site. After all disks are selected, press Enter.

Select disks to be mirrored from the remote site

Press F4 to display a list of available disks. Press F7 to select the disks that you want to geographically mirror from the remote site. After all disks are selected, press Enter.

Node Jess uses local disk hdisk9, and node Ellie uses local disk hdisk3 for the GMVG. Each one is associated with a rpvservers, which in turn is linked to their respective rpvcclients. The rpvcclients become hdisk1 on Jess and hdisk0 on Ellie, as shown in Figure 7-2 on page 251. The rpvcclients acquire these disk names because they are the first hdisk names that are available on each node. The output from running the synchronous GLVM wizard is shown in Example 7-3.

Example 7-3 Synchronous GLVM wizard output

Extracting the names for sites.

Extracting the name for nodes from both local and remote sites.

Creating RPVServers on all nodes of local site.

Creating RPVServers on node rpvserv0 Available
Creating RPVServers on all nodes of remote site.

Creating RPVServers on node rpvserv0 Available

Creating RPVServers on node rpvserv0 Available
Creating RPVClients on all nodes of local site.

Creating RPVClients on node hdisk1 Available

Creating RPVClients on all nodes of remote site.

Creating RPVClients on node hdisk0 Available

Changing RPVServers and RPVClients to defined and available state accordingly to facilitate the creation of VG.

Changing RPVServer rpvserver0 Defined

Changing RPVClient hdisk0 Defined

Generating Unique Names for Mirror pools and Resource Group.

Generating resource group (RG) name.

Unique names generated.

Creating VG syncglvmvg

Creating first mirror pool

Extending the VG to RPVClient disks and creating second mirror pool

Creating SYNC Mirror Pools

Varying on volume group:

Setting attributes for 0516-1804 chvg: The quorum change takes effect immediately.

Varying off volume group:

Changing RPVClient hdisk1 Defined

Changing RPVServer rpvserver0 Defined

Changing RPVServer rpvserver0 Available

Importing the VG

Changing RPVClient hdisk0 Available

Importing the VG synclovdm: No logical volumes in volume group syncglvmvg.
syncglvmvg

Varying on volume group:

Setting attributes for 0516-1804 chvg: The quorum change takes effect immediately.

Varying off volume group:

Changing RPVClient hdisk0 Defined

Definition of VG is available on all the nodes of the cluster.

Changing RPVServer rpvserver0 Defined

Creating a resource group.

Adding VG Verifying and synchronising the cluster configuration ...

Verification to be performed on the following:

Cluster Topology Cluster Resources

Retrieving data from available cluster nodes. This could take a few minutes.

```
Start data collection on node Jess
Start data collection on node Ellie
Collector on node Jess completed
Collector on node Ellie completed
Data collection complete
```

WARNING: No backup repository disk is UP and not already part of a VG for nodes:
- Jess
- Ellie

Completed 10 percent of the verification checks

WARNING: There are IP labels known to PowerHA SystemMirror and not listed in file /usr/es/sbin/cluster/etc/clhosts.client on node: Jess. Clverify can automatically populate this file to be used on a client node, if executed in auto-corRECTive mode.

WARNING: There are IP labels known to PowerHA SystemMirror and not listed in file /usr/es/sbin/cluster/etc/clhosts.client on node: Ellie. Clverify can automatically populate this file to be used on a client node, if executed in auto-corRECTive mode.

WARNING: An XD_data network has been defined, but no additional XD heartbeat network is defined. It is strongly recommended that an XD_ip network be configured in order to help prevent cluster partitioning if the XD_data network fails. Cluster partitioning may lead to data corruption for your replicated resources.

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
Verifying additional prerequisites for Dynamic Reconfiguration...
...completed.
```

Committing any changes, as required, to all available nodes...
Adding any necessary PowerHA SystemMirror for AIX entries to /etc/inittab and /etc/rc.net for IP address Takeover on node Jess.
Checking for any added or removed nodes
1 tunable updated on cluster GLVMdemocluster.
Adding any necessary PowerHA SystemMirror for AIX entries to /etc/inittab and /etc/rc.net for IP address Takeover on node Ellie.
Updating Split Merge policies

Verification has completed normally.

clsnapshot: Creating file /usr/es/sbin/cluster/snapshots/active.0.odm.

```

clsnapshot: Succeeded creating Cluster Snapshot: active.0
Attempting to sync user mirror groups (if any)...
Attempting to refresh user mirror groups (if any)...

cldare: Requesting a refresh of the Cluster Manager...
00026|NODE|Jess|VERIFY|PASSED|Fri Nov 18 11:20:38|A cluster configuration ver-
ification operation PASSED on node "Jess". Detailed output can be found in "/
var/hacmp/clverify/clverify.log" on that node.

PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_RP_RUNNING
PowerHA SystemMirror Cluster Manager current state is: ST_BARRIER
PowerHA SystemMirror Cluster Manager current state is: ST_RP_RUNNING.
PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_BARRIER..
PowerHA SystemMirror Cluster Manager current state is: ST_RP_RUNNING
PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_BARRIER..
PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_STABLE.....completed.

```

Synchronous cluster configuration

After the successful running of the GLVM wizard, the cluster RG is shown in Example 7-4.

Example 7-4 Synchronous GLVM resource group

Resource Group Name	syncglvmvg_RG
Participating Node Name(s)	Jess Ellie
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
Filesystems/Directories to be exported (NFSv3)	
Filesystems/Directories to be exported (NFSv4)	
Filesystems to be NFS mounted	
Network For NFS Mount	
Filesystem/Directory for NFSv4 Stable Storage	
Volume Groups	syncglvmvg
Concurrent Volume Groups	
Use forced varyon for volume groups, if necessary	true
Disks	
Raw Disks	
Disk Error Management?	no
GMVG Replicated Resources	syncglvmvg
GMD Replicated Resources	
PPRC Replicated Resources	
SVC PPRC Replicated Resources	
EMC SRDF? Replicated Resources	

Hitachi TrueCopy? Replicated Resources	
Generic XD Replicated Resources	
AIX Connections Services	
AIX Fast Connect Services	
Shared Tape Resources	
Application Servers	
Highly Available Communication Links	
Primary Workload Manager Class	
Secondary Workload Manager Class	
Delayed Fallback Timer	
Miscellaneous Data	
Automatically Import Volume Groups	false
Inactive Takeover	
SSA Disk Fencing	false
Filesystems mounted before IP configured	false
WPAR Name	

Primary node and site configuration

The primary node, Jess, has both a rpvserver, rpvserver0, and rpvcclient, hdisk1, created, and the scalable GMVG, syncglvmvg, is active. Also, there are two mirror pools, glvmMP01 and glvmMP02. Considering that there are no logical volumes or file systems that are created, the GMVG is also in sync. All of this is shown in Example 7-5.

Example 7-5 Synchronous GLVM primary site configuration

```
Jess# lspv
hdisk0      00f92db16aa2703a          rootvg      active
repositoryA 00f92db10031b9e9          caavg_private active
hdisk9      00f92db138ef5aee          syncglvmvg  active
hdisk10     00f92db17835e777         None        -
hdisk1      00f92db138df5181          syncglvmvg  active

Jess# lsvg syncglvmvg
VOLUME GROUP: syncglvmvg           VG IDENTIFIER:
00f92db100004c00000001587873d400
VG STATE:    active               PP SIZE:      8 megabyte(s)
VG PERMISSION: read/write          TOTAL PPs:   2542 (20336
megabytes)
MAX LVs:     256                 FREE PPs:    2542 (20336
megabytes)
LVs:         0                   USED PPs:   0 (0 megabytes)
OPEN LVs:    0                   QUORUM:     1 (Disabled)
TOTAL PVs:   2
STALE PVs:   0
ACTIVE PVs:  2
MAX PPs per VG: 32768
LTG size (Dynamic): 512 kilobyte(s)
HOT SPARE:   no
MIRROR POOL STRICT: super
PV RESTRICTION: none
DISK BLOCK SIZE: 512
FS SYNC OPTION: no

INFINITE RETRY: no
CRITICAL VG:   yes

Jess# lsmp -A syncglvmvg
```

```

VOLUME GROUP: syncglvmvg Mirror Pool Super Strict: yes

MIRROR POOL: glvMP01 Mirroring Mode: SYNC
MIRROR POOL: glvMP02 Mirroring Mode: SYNC

Jess# lsrvpclient -H
# RPV Client Physical Volume Identifier Remote Site
#
# hdisk1 00f92db138df5181 Chicago

Jess# lsrvpserver -H
# RPV Server Physical Volume Identifier Physical Volume
#
# rpvserver0 00f92db138ef5aee hdisk9

Jess# gmvgstat
GMVG Name PVs RPVs Tot Vols St Vols Total PPs Stale PPs Sync
----- ----- ----- ----- ----- ----- ----- -----
syncglvmvg 1 1 2 0 2542 0 100%

```

Secondary node and site configuration

The secondary node, Ellie, has both a rpvserver, rpvserver0, and rpvclient, hdisk0, created, and the scalable GMVG, syncglvmvg, is offline. Although the GMVG and mirror pools exist, they are not active on the secondary node, so their status is not known. All of this is shown in Example 7-6.

Example 7-6 Synchronous GLVM secondary site configuration

```

Ellie# lspv
repositoryB 00f92db1002568b2 caavg_private active
hdisk12 00f92db16aa2703a rootvg active
hdisk3 00f92db138df5181 syncglvmvg
hdisk2 00f92db17837528a None

Ellie# lsvg syncglvmvg
0516-010 : Volume group must be varied on; use varyonvg command.
#
Ellie# lsmp -A syncglvmvg
0516-010 lsmp: Volume group must be varied on; use varyonvg command.

Ellie# lsrvpserver -H
# RPV Server Physical Volume Identifier Physical Volumer
#
# rpvserver0 00f92db138df5181 hdisk3

# lsrvpclient -H
# RPV Client Physical Volume Identifier Remote Site
#
# hdisk0 00f92db138ef5aee Unknown

# gmvgstat
GMVG Name PVs RPVs Tot Vols St Vols Total PPs Stale PPs Sync
----- ----- ----- ----- ----- ----- ----- -----

```

```
gmvgstat: Failed to obtain geographically mirrored volume group information using  
lsglvm -v.
```

Completing the cluster configuration

To complete the configuration, perform the following steps.

- ▶ Create any additional resources that are required. The most common ones are as follows:
 - Site-specific service IPs
 - Application controllers
- ▶ Add the additional resources to the RG.
- ▶ Create all logical volumes and file systems that are required in the GMVG syncglvmvg.
- ▶ Synchronize the cluster.

Important: This procedure does *not* configure the GMVG on the remote node; that action must be done manually.

When creating logical volumes, ensure that two copies are created with the *superstrict* allocation policy and the mirror pools. This should be completed on the node in which the GMVG is active. In our case, it is node Jess. An example of creating a mirrored logical volume by running **smitty mklv** is shown in Example 7-7. Repeat as needed for every logical volume, and add any file systems that use the logical volumes, if applicable.

Example 7-7 Creating a mirrored logical volume

Add a Logical Volume

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

[Entry Fields]		
Logical volume NAME	[synclv]	
* VOLUME GROUP name	syncglvmvg	#
* Number of LOGICAL PARTITIONS	[10]	#
PHYSICAL VOLUME names	[]	+
Logical volume TYPE	[jfs2]	+
POSITION on physical volume	middle	+
RANGE of physical volumes	minimum	+
MAXIMUM NUMBER of PHYSICAL VOLUMES to use for allocation	[2]	#
Number of COPIES of each logical partition	2	+
Mirror Write Consistency?	active	+
Allocate each logical partition copy on a SEPARATE physical volume?	superstrict	+
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/reading logical partition copies	parallel write/sequen>	+
Enable WRITE VERIFY?	no	+
File containing ALLOCATION MAP	[]	
Stripe Size?	[Not Striped]	+

Serialize IO?	no	+
Mirror Pool for First Copy	glvmMP01	+
Mirror Pool for Second Copy	glvmMP02	+
Mirror Pool for Third Copy		+
Infinite Retry Option	no	

After all logical volumes are created, it is necessary to take the VG offline on the primary node, and then reimport the VG on the standby node by performing the following steps:

- ▶ On primary node Jess:
 - a. Deactivate the GMVG by running **varyoffvg syncglvmsg**.
 - b. Deactivate the rpvcclient, hdisk1 by running **rmdev -l hdisk1**.
 - c. Activate the rpvserv, rpvserv0 by running **mkdev -l rpvserv0**.
- ▶ On standby node Ellie:
 - a. Deactivate the rpvserv, rpvserv0 by running **rmdev -l rpvserv0**.
 - b. Activate rpvcclient, hdisk0 by running **mkdev -l hdisk0**.
 - c. Import the new VG information by running **importvg -L syncglvmsg hdisk0**.
 - d. Activate the VG by running **varyonvg syncglvmsg**.
 - e. Verify the GMVG information by running **1svg -l syncglvmsg**.

After you are satisfied that the GMVG information is correct, reverse these procedures to return the GMVG back to the primary node as follows:

- ▶ On standby node Ellie:
 - a. Deactivate the VG by running **varyoffvg syncglvmsg**.
 - b. Deactivate the rpvcclient, hdisk0 by running **rmdev -l hdisk0**.
 - c. Activate the rpvserv by running **mkdev -l rpvserv0**.
- ▶ On primary node Jess:
 - a. Deactivate the rpvserv, rpvserv0 by running **rmdev -l rpvserv0**.
 - b. Activate the rpvcclient, hdisk1 by running **mkdev -l hdisk1**.
 - c. Activate the GMVG by running **varyonvg syncglvmsg**.

Run a cluster verification, and if there are no errors, then the cluster can be tested.

7.3.3 Asynchronous configuration

Before attempting to use the GLVM wizard, you must complete all the prerequisites that are described in 7.2, “Prerequisites” on page 250. Our scenario consists of a basic two-site configuration, with one node at each site, and an XD_data network with a persistent alias defined in the configuration, as shown in Example 7-2 on page 252.

To begin, run **smitty sysmirror** and select **Cluster Applications and Resources → Make Applications Highly Available (Use Smart Assists) → GLVM Configuration Assistant → Configure Synchronous GMVG**.

The menu that is shown in Figure 7-5 on page 261 opens. If not, then the previously mentioned prerequisites have not been met, and you see a similar message as shown in Figure 7-6 on page 261.

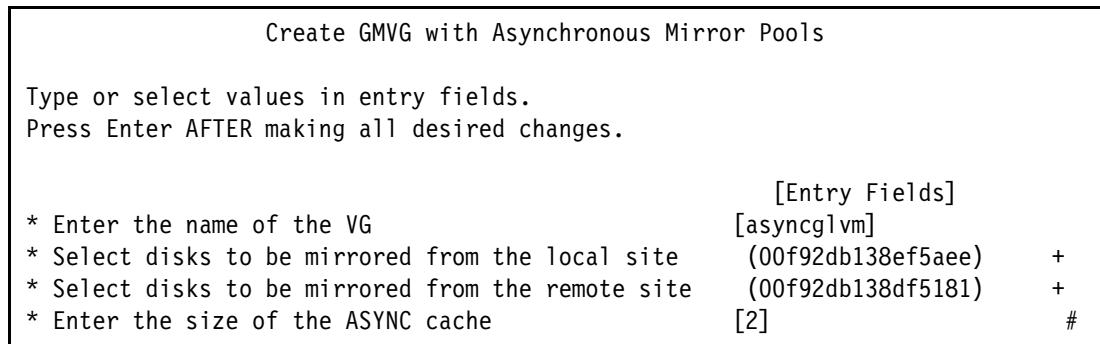


Figure 7-5 Asynchronous GLVM wizard menu

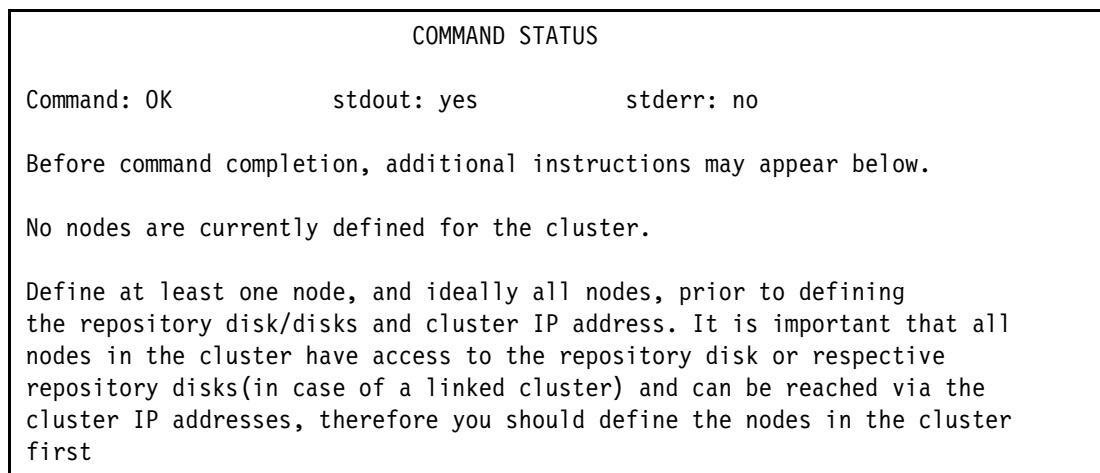


Figure 7-6 Async GLVM prerequisites not met

Enter the field values as follows:

Enter the Name of the VG

Enter the name of the VG that you want to create as a geographically mirrored VG. If the RG is created by using the GLVM Configuration Assistant, the VG name is appended with _RG. For example, if the VG name is syncglvmvg, the RG name is syncglvmvg_RG.

Select disks to be mirrored from the local site

Press F4 to display a list of available disks. Press F7 to select the disks that you want to geographically mirror from the local site. After all disks are selected, press Enter.

Select disks to be mirrored from the remote site

Press F4 to display a list of available disks. Press F7 to select the disks that you want to geographically mirror from the remote site. After all disks are selected, press Enter.

Enter the size of the ASYNCH cache

This is the aio_cache_lv, and one is created at each site. Enter the number of physical partitions (PPs) on the VG. The number that you enter depends on the load of the applications and bandwidth that is available in the network. You might need to enter different values for peak workload optimization.

Node Jess is using local disk hdisk9 and node Ellie is using hdisk3 for the GMVG. Each one is associated with a rpvserver, which in turn is linked to their respective rpvcclients. The rpvcclients become hdisk1 on Jess and hdisk0 on Ellie, as shown in Figure 7-2 on page 251. The rpvcclients acquire those disk names because they are the first hdisk names that are available on each node. The output from running the synchronous GLVM wizard is shown in Example 7-8.

Example 7-8 Asynchronous GLVM wizard output

Extracting the names for sites.

Extracting the name for nodes from both local and remote sites.

Creating RPVServers on all nodes of local site.

Creating RPVServers on node rpvserver0 Available

Creating RPVServers on all nodes of remote site.

Creating RPVServers on node rpvserver0 Available

Creating RPVClients on node rpvserver0 Available

Creating RPVClients on all nodes of local site.

Creating RPVClients on node hdisk1 Available

Creating RPVClients on all nodes of remote site.

Creating RPVClients on node hdisk0 Available

Changing RPVServers and RPVClients to defined and available state accordingly to facilitate the creation of VG.

Changing RPVServer rpvserver0 Defined

Changing RPVClient hdisk0 Defined

Generating Unique Names for Mirror pools and Resource Group.

Generating resource group (RG) name.

Unique names generated.

Creating VG asyncglvmvg

Creating first mirror pool

Extending the VG to RPVCClient disks and creating second mirror pool

Creating first ASYNC cache LV glvm_cache_LV01

Creating second ASYNC cache LV glvm_cache_LV02

Varying on volume group:

Setting attributes for 0516-1804 chvg: The quorum change takes effect immediately.

Varying off volume group:

Changing RPVClient hdisk1 Defined

Changing RPVServer rpvserver0 Defined

Changing RPVServer rpvserver0 Available

Importing the VG

Changing RPVClient hdisk0 Available

Importing the VG syncldvdm: No logical volumes in volume group asyncglvmvg.
asyncglvmvg

Varying on volume group:

Setting attributes for 0516-1804 chvg: The quorum change takes effect immediately.

Varying off volume group:

Changing RPVClient hdisk0 Defined

Definition of VG is available on all the nodes of the cluster.

Changing RPVServer rpvserver0 Defined

Creating a resource group.

Adding VG Verifying and synchronising the cluster configuration ...

Verification to be performed on the following:

Cluster Topology

Cluster Resources

Retrieving data from available cluster nodes. This could take a few minutes.

Start data collection on node Jess

Start data collection on node Ellie

Collector on node Jess completed

Collector on node Ellie completed

Data collection complete

WARNING: No backup repository disk is UP and not already part of a VG for nodes:

- Jess

- Ellie

Completed 10 percent of the verification checks

WARNING: There are IP labels known to PowerHA SystemMirror and not listed in file /usr/es/sbin/cluster/etc/clhosts.client on node: Jess. Clverify can automatically populate this file to be used on a client node, if executed in auto-correction mode.

WARNING: There are IP labels known to PowerHA SystemMirror and not listed in file /usr/es/sbin/cluster/etc/clhosts.client on node: Ellie. Clverify can automatically populate this file to be used on a client node, if executed in auto-correction mode.

WARNING: An XD_data network has been defined, but no additional XD heartbeat network is defined. It is strongly recommended that an XD_ip network be configured in order to help prevent cluster partitioning if the XD_data network fails. Cluster partitioning may lead to data corruption for your replicated resources.

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
Verifying additional prerequisites for Dynamic Reconfiguration...
...completed.

Committing any changes, as required, to all available nodes...
Adding any necessary PowerHA SystemMirror for AIX entries to /etc/inittab and /etc/rc.net for IP address Takeover on node Jess.
Checking for any added or removed nodes
1 tunable updated on cluster GLVMDemocluster.
Adding any necessary PowerHA SystemMirror for AIX entries to /etc/inittab and /etc/rc.net for IP address Takeover on node Ellie.
Updating Split Merge policies

Verification has completed normally.

clsnapshot: Creating file /usr/es/sbin/cluster/snapshots/active.0.odm.

clsnapshot: Succeeded creating Cluster Snapshot: active.0
Attempting to sync user mirror groups (if any)...
Attempting to refresh user mirror groups (if any)...

cldare: Requesting a refresh of the Cluster Manager...
00026|NODE|Jess|VERIFY|PASSED|Fri Nov 18 11:20:38|A cluster configuration verification operation PASSED on node "Jess". Detailed output can be found in "/var/hacmp/clverify/clverify.log" on that node.

PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_RP_RUNNING
PowerHA SystemMirror Cluster Manager current state is: ST_BARRIER
PowerHA SystemMirror Cluster Manager current state is: ST_RP_RUNNING.
PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_BARRIER..
PowerHA SystemMirror Cluster Manager current state is: ST_RP_RUNNING
PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_BARRIER..
PowerHA SystemMirror Cluster Manager current state is: ST_UNSTABLE.
PowerHA SystemMirror Cluster Manager current state is: ST_STABLE....completed.

Asynchronous cluster configuration

After the successful running of the GLVM wizard, the cluster RG is shown in Example 7-9.

Example 7-9 Synchronous GLVM resource group

Resource Group Name	asyncglvmvg_RG
Participating Node Name(s)	Jess Ellie
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
Filesystems/Directories to be exported (NFSv3)	
Filesystems/Directories to be exported (NFSv4)	
Filesystems to be NFS mounted	
Network For NFS Mount	
Filesystem/Directory for NFSv4 Stable Storage	
Volume Groups	asyncglvmvg
Concurrent Volume Groups	
Use forced varyon for volume groups, if necessary	true
Disks	
Raw Disks	
Disk Error Management?	no
GMVG Replicated Resources	asyncglvmvg
GMD Replicated Resources	
PPRC Replicated Resources	
SVC PPRC Replicated Resources	
EMC SRDF? Replicated Resources	
Hitachi TrueCopy? Replicated Resources	
Generic XD Replicated Resources	
AIX Connections Services	
AIX Fast Connect Services	
Shared Tape Resources	
Application Servers	
Highly Available Communication Links	
Primary Workload Manager Class	
Secondary Workload Manager Class	
Delayed Fallback Timer	
Miscellaneous Data	
Automatically Import Volume Groups	false
Inactive Takeover	
SSA Disk Fencing	false
Filesystems mounted before IP configured	false
WPAR Name	

Primary node and site configuration

The primary node, Jess, has both a rpvserver, rpvserver0, and rpvcclient, hdisk1, created, and the scalable GMVG, asyncglvmvg, is active. Also, the node has two mirror pools, glvmMP01 and glvmMP02. Two aio_cache_lv logical volumes, glvm_cache_LV01 and glvm_cache_LV02, are also created. All of this is shown in Example 7-10.

Example 7-10 Asynchronous GLVM primary site configuration

```
Jess# lspv
hdisk0      00f92db16aa2703a          rootvg      active
repositoryA 00f92db10031b9e9          caavg_private active
hdisk9      00f92db138ef5aee        asyncglvmvg   active
hdisk10     00f92db17835e777         None        active
hdisk1      00f92db138df5181        syncglvmvg   active

Jess# lsvg asyncglvmvg
```

```

VOLUME GROUP: syncglvmvg VG IDENTIFIER:
00f92db100004c00000001587873d400
VG STATE: active PP SIZE: 8 megabyte(s)
VG PERMISSION: read/write TOTAL PPs: 2542 (20336 megabytes)
MAX LVs: 256 FREE PPs: 2538 (20304 megabytes)
LVs: 0 USED PPs: 0 (0 megabytes)
OPEN LVs: 0 QUORUM: 1 (Disabled)
TOTAL PVs: 2 VG DESCRIPTORS: 3
STALE PVs: 0 STALE PPs: 0
ACTIVE PVs: 2 AUTO ON: no
MAX PPs per VG: 32768 MAX PVs: 1024
LTG size (Dynamic): 512 kilobyte(s) AUTO SYNC: no
HOT SPARE: no BB POLICY: non-relocatable
MIRROR POOL STRICT: super
PV RESTRICTION: none INFINITE RETRY: no
DISK BLOCK SIZE: 512 CRITICAL VG: yes
FS SYNC OPTION: no

Jess# lsvg -l asyncglvm
asyncglvm:
LV NAME TYPE LPs PPs PVs LV STATE MOUNT POINT
glvm_cache_LV01 aio_cache 2 2 1 open/syncd N/A
glvm_cache_LV02 aio_cache 2 2 1 closed/syncd N/A

Jess# lsmp -A asycnglvm
VOLUME GROUP: asycnglvm Mirror Pool Super Strict: yes

MIRROR POOL: glvmMP01 Mirroring Mode: ASYNC
ASYNC MIRROR STATE: inactive ASYNC CACHE LV: glvm_cache_LV02
ASYNC CACHE VALID: yes ASYNC CACHE EMPTY: yes
ASYNC CACHE HWM: 80 ASYNC DATA DIVERGED: no

MIRROR POOL: glvmMP02 Mirroring Mode: ASYNC
ASYNC MIRROR STATE: active ASYNC CACHE LV: glvm_cache_LV01
ASYNC CACHE VALID: yes ASYNC CACHE EMPTY: no
ASYNC CACHE HWM: 80 ASYNC DATA DIVERGED: no

Jess# lsrpvclient -H
# RPV Client Physical Volume Identifier Remote Site
# -----
hdisk1 00f92db138df5181 Chicago

Jess# lsrpvserver -H
# RPV Server Physical Volume Identifier Physical Volume
# -----
rpvserver0 00f92db138ef5aee hdisk9

Jess# gmvgstat
GMVG Name PVs RPVs Tot Vols St Vols Total PPs Stale PPs Sync
----- ----- ----- ----- ----- -----
syncglvmvg 1 1 2 0 2542 0 100%

```

Secondary node and site configuration

The secondary node, Ellie, has both a rpvserver, rpvserver0, and rpvclient, hdisk0, created, and the scalable GMVG, syncglvmvg, is offline. Although the GMVG and mirror pools exist, they are not active on the secondary node, and their status is not known. All of this is shown in Example 7-11 on page 267.

Example 7-11 Asynchronous GLVM secondary site configuration

```
Ellie# lspv

repositoryB      00f92db1002568b2          caavg_private   active
hdisk12         00f92db16aa2703a          rootvg           active
hdisk3          00f92db138df5181        asyncglvmvg
hdisk2          00f92db17837528a          None

Ellie# lsvg syncglvmvg
0516-010 : Volume group must be varied on; use varyonvg command.
#
Ellie# lsmp -A syncglvmvg
0516-010 lsmp: Volume group must be varied on; use varyonvg command.

Ellie# lsrvpserver -H
# RPV Server      Physical Volume Identifier      Physical Volumer
# -----
    rpvserver0     00f92db138df5181                hdisk3

# lsrvpclient -H
# RPV Client      Physical Volume Identifier      Remote Site
# -----
    hdisk0        00f92db138ef5aee                Unknown

# gmvgstat
GMVG Name      PVs  RPVs  Tot Vols  St Vols  Total PPs  Stale PPs  Sync
-----  -----  -----  -----  -----  -----  -----  -----
gmvgstat: Failed to obtain geographically mirrored volume group information using
lsglvm -v.
```

Completing the cluster configuration

To complete the configuration, complete the following steps.

- ▶ Create any additional resources that are required. The most common ones are the following ones:
 - Site-specific service IPs
 - Application controllers
- ▶ Add the additional resources to the RG.
- ▶ Create all the logical volumes and file systems that are required in the GMVG syncglvmvg.
- ▶ Synchronize the cluster.

Important: This procedure does *not* configure the GMVG on the remote node. That procedure must be done manually.

When creating logical volumes, ensure that two copies are created with a *superstrict* allocation policy and the mirror pools, which should be completed on the node in which the GMVG is active. In our case, it is node Jess. An example of creating a mirrored logical volume by running **smitty mklv** is shown in Example 7-12. Repeat as needed for every logical volume, and add any file systems that use the logical volumes, if applicable.

Example 7-12 Creating a mirrored logical volume

Add a Logical Volume

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

[Entry Fields]		
Logical volume NAME	[asynclv]	
* VOLUME GROUP name	asyncglvmvg	#
* Number of LOGICAL PARTITIONS	[20]	#
PHYSICAL VOLUME names	[]	+
Logical volume TYPE	[jfs2]	+
POSITION on physical volume	middle	+
RANGE of physical volumes	minimum	+
MAXIMUM NUMBER of PHYSICAL VOLUMES to use for allocation	[2]	#
Number of COPIES of each logical partition	2	+
Mirror Write Consistency?	active	+
Allocate each logical partition copy on a SEPARATE physical volume?	superstrict	+
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/reading logical partition copies	parallel write/sequen>	+
Enable WRITE VERIFY?	no	+
File containing ALLOCATION MAP	[]	
Stripe Size?	[Not Striped]	+
Serialize IO?	no	+
Mirror Pool for First Copy	glvmMP01	+
Mirror Pool for Second Copy	glvmMP02	+
Mirror Pool for Third Copy		+
Infinite Retry Option	no	

After all logical volumes are created, it is necessary to take the VG offline on the primary node and then reimport the VG on the standby node by completing the following steps:

- ▶ On primary node Jess:
 - a. Deactivate the GMVG by running **varyoffvg asyncglvmvg**.
 - b. Deactivate the rpvclient, hdisk1 by running **rmdev -l hdisk1**.
 - c. Activate the rpvserver, rpvserver0 by running **mkdev -l rpvserver0**.
- ▶ On standby node Ellie:
 - a. Deactivate the rpvserver, rpvserver0 by running **rmdev -l rpvserver0**.
 - b. Activate rpvclient, hdisk0 by running **mkdev -l hdisk0**.

- c. Import new VG information by running **importvg -L asynclvmvg hdisk0**.
- d. Activate the VG by running **varyonvg syncg1vmg**.
- e. Verify the GMVG information by running **lsvg -l syncg1vmg**.

After you are satisfied that the GMVG information is correct, reverse these procedures to return the GMVG back to the primary node:

- ▶ On standby node Ellie:
 - a. Deactivate the VG by running **varyoffvg asynclvmg**.
 - b. Deactivate the rpvclient, hdisk0 by running **rmdev -l hdisk0**.
 - c. Activate the rpvserver by running **mkdev -l rpvserver0**.
- ▶ On primary node Jess:
 - a. Deactivate the rpvserver, rpvserver0 by running **rmdev -l rpvserver0**.
 - b. Activate the rpvclient, hdisk1 by running **mkdev -l hdisk1**.
 - c. Activate the GMVG by running **varyonvg asynclvmvg**.

Run a cluster verification, and if there are no errors, then the cluster can be tested.



Automation adaptation for Live Partition Mobility

This chapter covers a feature that was originally introduced in PowerHA V7.2.0: Automation adaptation for Live Partition Mobility (LPM).

Before PowerHA SystemMirror V7.2, if customers wanted to implement the LPM operation for one AIX LPAR that is running the PowerHA service, they had to perform a manual operation, which is illustrated at [IBM Knowledge Center](#).

This feature plugs into the LPM infrastructure to maintain awareness of LPM events and adjusts the clustering related monitoring as needed for the LPM operation to succeed without disruption. This feature reduces the burden on the administrator to perform manual operations on the cluster node during LPM operations. For more information about this feature, see [IBM Knowledge Center](#).

This chapter introduces the necessary operations to ensure that the LPM operation for the PowerHA node completes successfully. This chapter uses both PowerHA V7.1 and PowerHA V7.2 cluster environments to illustrate the scenarios.

This chapter covers the following topics:

- ▶ Concept
- ▶ Prerequisites for PowerHA node support of Live Partition Mobility
- ▶ Operation flow to support Live Partition Mobility on a PowerHA node
- ▶ Example: Live Partition Mobility scenario for PowerHA V7.1
- ▶ Live Partition Mobility SMIT panel
- ▶ PowerHA V7.2 scenario and troubleshooting

8.1 Concept

This section provides an introduction to the LPM concepts.

Live Partition Mobility

LPM enables you to migrate LPARs running the AIX operating system and their hosted applications from one physical server to another without disrupting the infrastructure services. The migration operation maintains system transactional integrity and transfers the entire system environment, including processor state, memory, attached virtual devices, and connected users.

LPM provides the facility for no downtime for planned hardware maintenance. However, LPM does not offer the same facility for software maintenance or unplanned downtime. You can use PowerHA SystemMirror within a partition that is capable of LPM. However, this does not mean PowerHA SystemMirror uses LPM, and PowerHA treats LPM as another application within the partition.

Live Partition Mobility operation and freeze times

The amount of operational time that an LPM migration requires on an LPAR is determined by multiple factors, such as the LPAR's memory size, workload activity (more memory pages require more memory updates across the system), and network performance.

LPAR freeze time is a part of LPM operational time, and it occurs when the LPM tries to reestablish the memory state. During this time, no other processes can operate in the LPAR. As part of this memory reestablishment process, memory pages from the source system can be copied to the target system over the network connection. If the network connection is congested, this process of copying over the memory pages can increase the overall LPAR freeze time.

Cluster software in a PowerHA cluster environment

In a PowerHA solution, although PowerHA is one cluster software, there are two other kinds of cluster software running behind the PowerHA cluster:

- ▶ Reliable Scalable Cluster Technology (RSCT)
- ▶ Cluster Aware AIX (CAA)

For a description of their relationship, see 4.5, “PowerHA, Reliable Scalable Clustering Technology, and Cluster Aware AIX” on page 103.

PowerHA cluster heartbeating and the Dead Man Switch

PowerHA SystemMirror uses constant communication between the nodes to track the health of the cluster, nodes, and so on. One of the key components of communication is the heartbeating between the nodes. Lack of heartbeats forms a critical part of the decision-making process to declare a node to be dead.

The PowerHA V7.2 default node failure detection time is 40 seconds, and 30 seconds for node communication timeout plus a 10-second grace period. These values can be altered as wanted.

Node A declares partner Node B to be dead if Node A did not receive any communication or heartbeats for more than 40 seconds. This process works well when Node B is dead (crashed, powered off, and so on). However, there are scenarios where Node B is not dead, but cannot communicate for long periods.

Here are two examples of such scenarios:

1. There is one communication link between the nodes and it is broken (multiple communication links should be deployed between the nodes to avoid this scenario).
2. Due to a rare situation, the operating system freezes the cluster processes and kernel threads such that the node cannot send any I/O (disk or network) for more than 40 seconds. This situation results in the same situation where Node A cannot receive any communication from Node B for more than 40 seconds, and therefore declares Node B to be dead, even though it is alive. This leads to a *split-brain* condition, which can result in data corruption if the disks are shared across nodes.

Some scenarios can be handled in the cluster. For example, in scenario 2, when Node B is allowed to run after the unfreeze, it recognizes the fact that it has not been able to communicate to other nodes for a long time and takes evasive actions. Those types of action are called Dead Man Switch (DMS) protection.

DMS involves timers that monitor various activities, such as I/O traffic and process health, to recognize stray cases where there is potential for it (Node B) to be considered dead by its peers in the cluster. In these cases, the DMS timers trigger just before the node failure detection time and evasive action is initiated. A typical evasive action involves fencing the node.

PowerHA SystemMirror consists of different DMS protections:

- ▶ CAA DMS protection

When CAA detects that a node is isolated in a multiple node environment, a DMS is triggered. This timeout occurs when the node cannot communicate with other nodes during the delay that is specified by the `node_timeout` cluster tunable. The system crashes with an errlog Deadman timer triggered if the `deadman_mode` cluster tunable (`c1ctr1 -tune`) is set to `a` (assert mode, which is the default), or only log an event if `deadman_mode` is set to `e` (event mode).

This protection can occur on the node performing LPM, or on both nodes in a two-node cluster. To prevent a system crash due to this timeout, increase `node_timeout` to its maximum value, which is 600 seconds before LPM and restore it after LPM.

Note: This operation is done manually with a PowerHA SystemMirror V7.1 node. Section 8.3, “Example: Live Partition Mobility scenario for PowerHA V7.1” on page 279 introduces the operation. This operation is done automatically with a PowerHA System V7.2 node, as described in 8.4, “Live Partition Mobility SMIT panel” on page 296.

- ▶ Group Services DMS

Group services is a critical component that allows for cluster-wide membership and group management. This daemon's health is monitored continuously. If this process exits or becomes inactive for long periods, then the node is brought down.

- ▶ RSCT RMC, ConfigRMC, `c1strmgr`, and IBM.StorageRM daemons

Group Services monitor the health of these daemons. If they are inactive for a long time or exit, then the node is brought down.

Note: The Group Service (`cthags`) DMS timeout with AIX 7.2.1, at the time of writing, is 60 seconds. For now, it is hardcoded, and cannot be changed.

Therefore, if the LPM freeze time is longer than the Group Service DMS timeout, Group Service (`cthags`) reacts and halts the node.

Because we cannot tune the parameter to increase its timeout, you must disable RSCT critical process monitoring before LPM, and enable it after LPM, by using the following commands:

- Disable RSCT critical process monitoring

To disable RSCT monitoring process, use the following commands:

```
/usr/sbin/rsct/bin/hags_disable_client_kill -s cthags  
/usr/sbin/rsct/bin/dms/stopdms -s cthags
```

- Enable RSCT critical process monitoring

To enable RSCT monitoring process, use the following commands:

```
/usr/sbin/rsct/bin/dms/startdms -s cthags  
/usr/sbin/rsct/bin/hags_enable_client_kill -s cthags
```

Note: This operation is done manually in a PowerHA SystemMirror V7.1 node, as described in 8.3, “Example: Live Partition Mobility scenario for PowerHA V7.1” on page 279. This operation is done automatically in a PowerHA System V7.2 node, as described in 8.4, “Live Partition Mobility SMIT panel” on page 296.

8.1.1 Prerequisites for PowerHA node support of Live Partition Mobility

This section describes the prerequisites for PowerHA node support for LPM.

8.1.2 PowerHA fix requirement

For PowerHA SystemMirror V7.1 to support changing CAA’s node_time variable online through the PowerHA `c1mgr` command, the following APARs are required:

- ▶ PowerHA SystemMirror Version 7.1.2 - IV79502 (in SP8)
- ▶ PowerHA SystemMirror Version 7.1.3 - IV79497 (in SP5)

Without these APARs in PowerHA V7.1.1, the change requires two steps to change the CAA node_timeout variable. For more information, see “Increasing the Cluster Aware AIX node_timeout parameter” on page 286.

8.1.3 Reducing the Live Partition Mobility freeze time

To reduce the freeze time during the LPM operation, use 10-Gb network adapters and a dedicated network with enough bandwidth available, and reduce memory activity during LPM.

8.2 Operation flow to support Live Partition Mobility on a PowerHA node

The operation flow includes pre-migration and post-migration.

If the PowerHA version is earlier than Version 7.2, then you must perform the operations manually. If PowerHA version is Version 7.2 or later, the PowerHA performs the operations automatically.

This section introduces pre-migration and post-migration operation flow during LPM.

8.2.1 Pre-migration operation flow

Figure 8-1 describes the operation flow in a pre-migration stage.

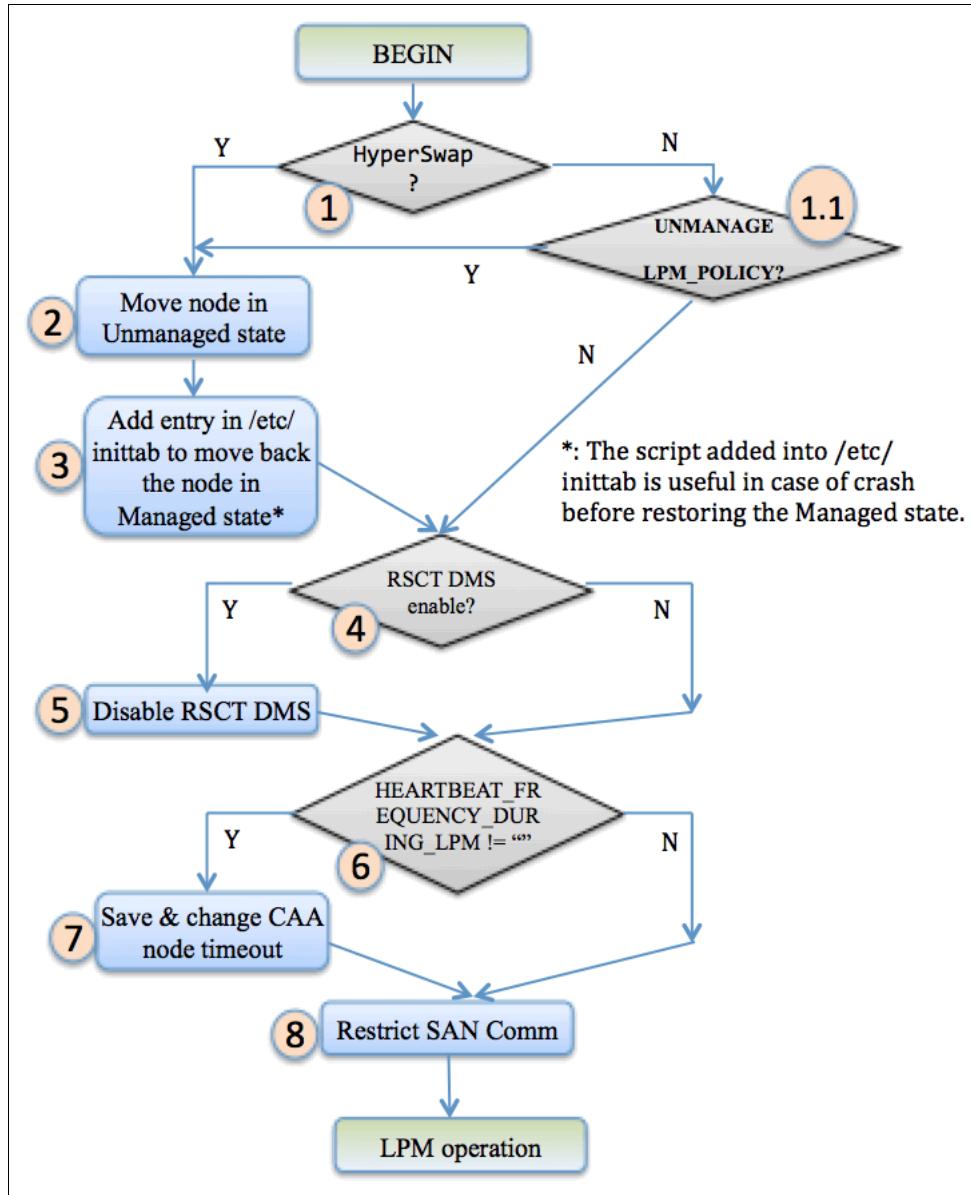


Figure 8-1 Pre-migration operation flow

Table 8-1 shows the detailed information for each step in the pre-migration stage.

Table 8-1 Description of the pre-migration operation flow

Step	Description
1	Check if HyperSwap is used. If YES, go to 2; otherwise, go to 1.1.
1.1	Check whether LPM_POLICY=unmanage is set. If YES, go to 2; otherwise, go to 4 by running the following command: <code>clodmget -n -f lpm_policy HACMPcluster</code>
2	Change the node to unmanage resource group status by running the following command: <code>clmgr stop node <node_name> WHEN=now MANAGE=unmanage</code>
3	Add an entry to the /etc/inittab file, which is useful in a node crash before restoring the managed state, by running the following command: <code>mkitab hacmp_lpm:2:once:/usr/es/sbin/cluster/utilities/cl_dr undopremigrate > /dev/null 2>&1</code>
4	Check whether RSCT DMS critical resource monitoring is enabled by running the following command: <code>/usr/sbin/rsct/bin/dms/listdms -s cthags grep -qw Enabled</code>
5	Disable RSCT DMS critical resource monitoring by running the following commands: <code>/usr/sbin/rsct/bin/hags_disable_client_kill -s cthags</code> <code>/usr/sbin/rsct/bin/dms/stopdms -s cthags</code>
6	Check whether the current node_timeout value is equal to the value that you set by running the following commands: <code>clodmget -n -f lpm_node_timeout HACMPcluster</code> <code>clctrl -tune -x node_timeout</code>
7	Change the CAA node_timeout value by running the following command: <code>clmgr -f modify cluster HEARTBEAT_FREQUENCY="600"</code>
8	If SAN-based heartbeating is enabled, then disable this function by running the following commands: <code>echo 'sfwcom' >> /etc/cluster/ifrestrict</code> <code>clusterconf</code>

8.2.2 Post-migration operation flow

Figure 8-2 describes the operation flow in the post-migration stage.

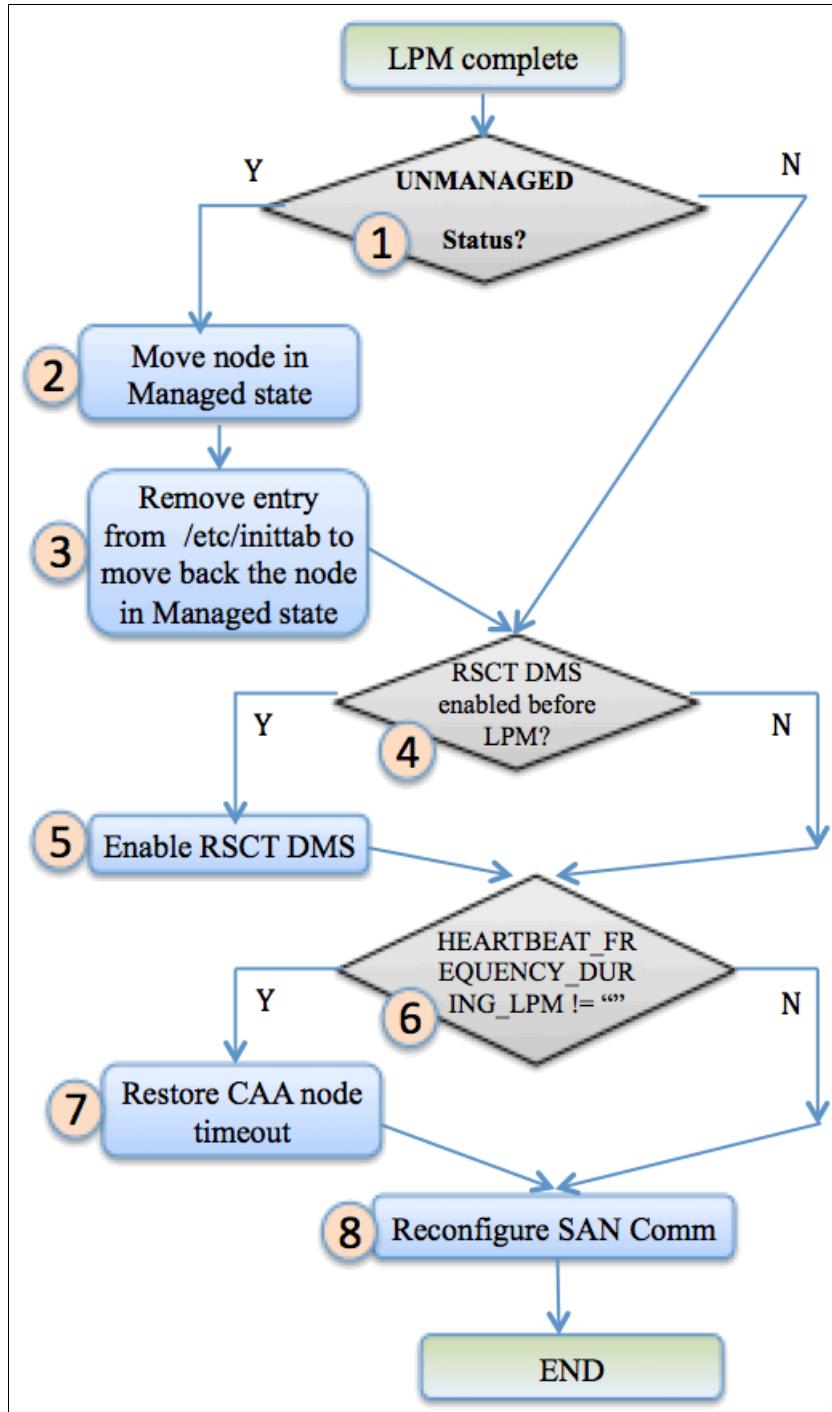


Figure 8-2 Post-migration operation flow

Table 8-2 shows the detailed information for each step in the post-migration stage.

Table 8-2 Description of the post-migration operation flow

Step	Description
1	Check whether the current resource group status is unmanaged. If Yes, go to 2; otherwise, go to 4.
2	Change the node back to the manage resource group status by running the following command: <code>clmgr start node <node_name> WHEN=now MANAGE=auto</code>
3	Remove the entry from the /etc/inittab file that was added in the pre-migration process by running the following command: <code>rmitab hacmp_1pm</code>
4	Check whether the RSCT DMS critical resource monitoring function is enabled before the LPM operation.
5	Enable RSCT DMS critical resource monitoring by running the following commands: <code>/usr/sbin/rsct/bin/dms/startdms -s cthags</code> <code>/usr/sbin/rsct/bin/hags_enable_client_kill -s cthags</code>
6	Check whether the current node_timeout value is equal to the value that you set before by running the following commands: <code>clctrl -tune -x node_timeout</code> <code>clodmget -n -f 1pm_node_timeout HACMPcluster</code>
7	Restore the CAA node_timeout value by running the following command: <code>clmgr -f modify cluster HEARTBEAT_FREQUENCY="30"</code>
8	If SAN-based heartbeating is enabled, then enable this function by running the following commands: <code>rm -f /etc/cluster/ifrestrict</code> <code>clusterconf</code> <code>rmdev -l sfwcomm*</code> <code>mkdev -l sfwcomm*</code>

8.3 Example: Live Partition Mobility scenario for PowerHA V7.1

This section introduces detailed operations for performing LPM for one node with PowerHA SystemMirror V7.1.

8.3.1 Topology introduction

Figure 8-3 describes the topology of the testing environment.

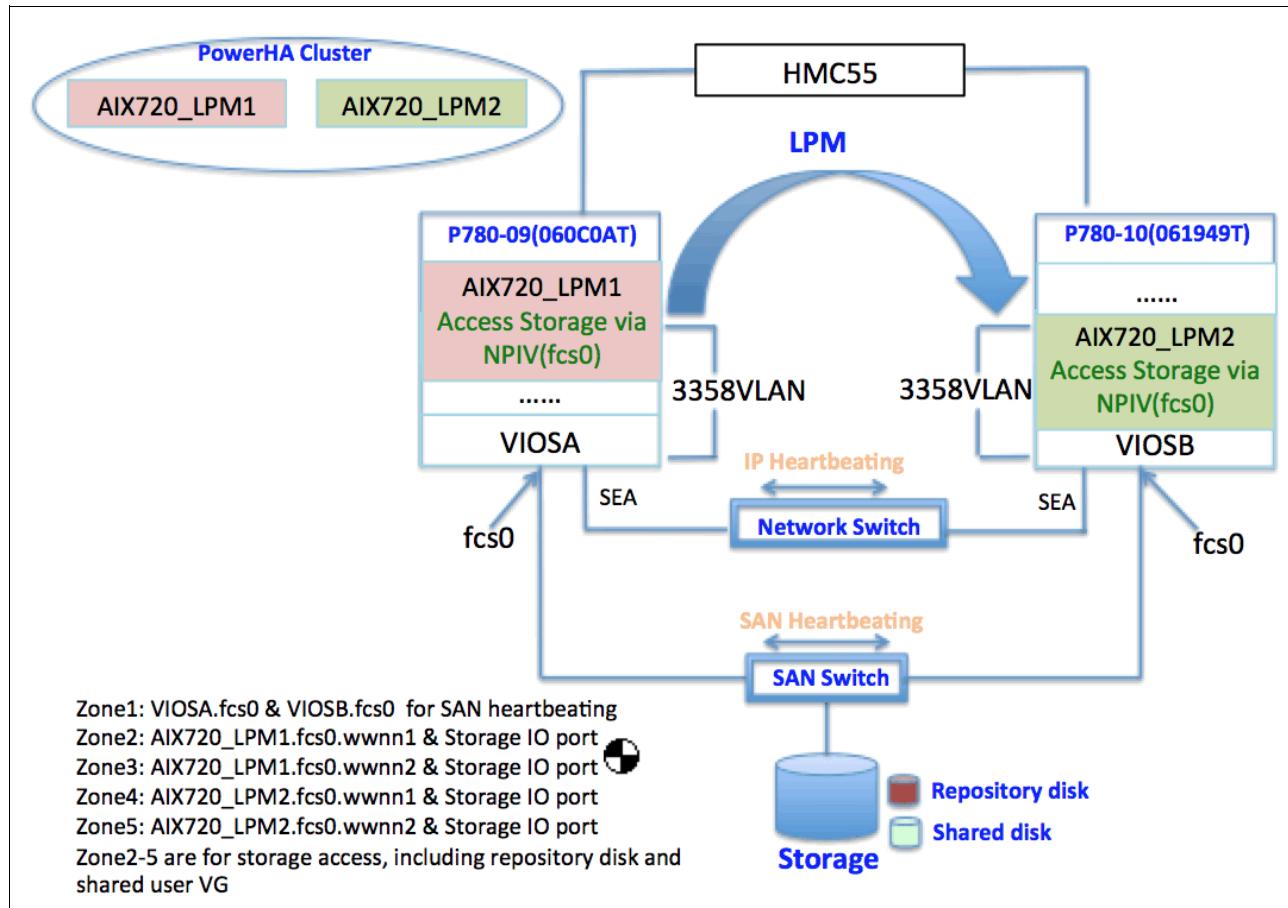


Figure 8-3 Testing environment topology

There are two Power Systems 780 servers. The first server is P780_09 and its serial number is 060C0AT, and the second server is P780_10 and its machine serial number is 061949T.

The following list provides additional details about the testing environment:

- Each server has one Virtual I/O Server (VIOS) partition and one AIX partition.
- The P780_09 server has VIOSA and AIX720_LPM1 partitions.
- The P780_10 server has VIOSB and AIX720_LPM2 partitions.
- There is one storage that can be accessed by the two VIOSs.
- The two AIX partitions access storage by way of the NPIV protocol.
- The heartbeating method includes IP, SAN, and dpcm.
- The AIX version is AIX 7.2 SP1.
- The PowerHA SystemMirror version is Version 7.1.3 SP4.

8.3.2 Initial status

This section describes the initial cluster status.

PowerHA and AIX version

Example 8-1 shows the PowerHA and the AIX version information.

Example 8-1 PowerHA and AIX version information

```
AIX720_LPM1:/usr/es/sbin/cluster # clhaver
Node AIX720_LPM2 has HACMP version 7134 installed
Node AIX720_LPM1 has HACMP version 7134 installed

AIX720_LPM1:/usr/es/sbin/cluster # clcmd oslevel -s
-----
NODE AIX720_LPM2
-----
7200-00-01-1543
-----
NODE AIX720_LPM1
-----
7200-00-01-1543
```

PowerHA configuration

Table 8-3 shows the cluster's configuration.

Table 8-3 Cluster configuration

Item	AIX720_LPM1	AIX720_LPM2
Cluster name	LPMCluster Cluster type: No Site Cluster (NSC)	
Network interface	en1: 172.16.50.21 Netmask: 255.255.255.0 Gateway: 172.16.50.1	en0: 172.16.50.22 Netmask: 255.255.255.0 Gateway: 172.16.50.1
Network	net_ether_01 (172.16.50.0/24)	
CAA	Unicast Primary disk: hdisk1	
Shared VG	testVG: hdisk2	
Service IP	172.16.50.23 AIX720_LPM_Service	
Resource Group (RG)	testRG includes testVG, AIX720_LPM_Service. The node order is AIX720_LPM1, AIX720_LPM2. Startup Policy: Online On Home Node Only Failover Policy: Failover To Next Priority Node In The List Fallback Policy: Never Fallback	

PowerHA and resource group status

Example 8-2 shows the status of PowerHA and the RG.

Example 8-2 PowerHA and resource group status

```
AIX720_LPM1:/ # clcmd -n LPMCluster lssrc -ls clstrmgrES|egrep "NODE|state"|grep -v "Last"
NODE AIX720_LPM2
Current state: ST_STABLE
NODE AIX720_LPM1
Current state: ST_STABLE

AIX720_LPM1:/ # clcmd -n LPMCluster clRGinfo
-----
NODE AIX720_LPM2
-----
-----

| Group Name | State   | Node        |
|------------|---------|-------------|
| testRG     | ONLINE  | AIX720_LPM1 |
|            | OFFLINE | AIX720_LPM2 |


-----
NODE AIX720_LPM1
-----
-----

| Group Name | State   | Node        |
|------------|---------|-------------|
| testRG     | ONLINE  | AIX720_LPM1 |
|            | OFFLINE | AIX720_LPM2 |


```

Cluster Aware AIX heartbeating status

Example 8-3 shows the current CAA heartbeating status and node_timeout parameter.

Example 8-3 Cluster Aware AIX heartbeating status and the value of the node_timeout parameter

```
AIX720_LPM1:/ # clcmd lscluster -m
-----
NODE AIX720_LPM2
-----
Calling node query for all nodes...
Node query number of nodes examined: 2

    Node name: AIX720_LPM1
    Cluster shorthand id for node: 1
    UUID for node: 112552f0-c4b7-11e5-8014-56c6a3855d04
    State of node: UP
    Smoothed rtt to node: 7
    Mean Deviation in network rtt to node: 3
    Number of clusters node is a member in: 1
    CLUSTER NAME      SHID      UUID
    LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
    SITE NAME         SHID      UUID
```

LOCAL 1 51735173-5173-5173-5173-517351735173

Points of contact for node: 2

Interface	State	Protocol	Status	SRC_IP->DST_IP
sfwcom	UP	none	none	none
tcpsock->01	UP	IPv4	none	172.16.50.22->172.16.50.21

Node name: AIX720_LPM2

Cluster shorthand id for node: 2

UUID for node: 11255336-c4b7-11e5-8014-56c6a3855d04

State of node: UP NODE_LOCAL

Smoothed rtt to node: 0

Mean Deviation in network rtt to node: 0

Number of clusters node is a member in: 1

CLUSTER NAME	SHID	UUID
LPMCluster	0	11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME	SHID	UUID
LOCAL	1	51735173-5173-5173-5173-517351735173

Points of contact for node: 0

NODE AIX720_LPM1

Calling node query for all nodes...

Node query number of nodes examined: 2

Node name: AIX720_LPM1

Cluster shorthand id for node: 1

UUID for node: 112552f0-c4b7-11e5-8014-56c6a3855d04

State of node: UP NODE_LOCAL

Smoothed rtt to node: 0

Mean Deviation in network rtt to node: 0

Number of clusters node is a member in: 1

CLUSTER NAME	SHID	UUID
LPMCluster	0	11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME	SHID	UUID
LOCAL	1	51735173-5173-5173-5173-517351735173

Points of contact for node: 0

Node name: AIX720_LPM2

Cluster shorthand id for node: 2

UUID for node: 11255336-c4b7-11e5-8014-56c6a3855d04

State of node: UP

Smoothed rtt to node: 17

Mean Deviation in network rtt to node: 13

Number of clusters node is a member in: 1

CLUSTER NAME	SHID	UUID
--------------	------	------

```

LPMCluster          0           11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME          SHID        UUID
LOCAL              1           51735173-5173-5173-5173-517351735173

Points of contact for node: 2
-----
Interface  State  Protocol  Status  SRC_IP->DST_IP
-----
sfwcom     UP     none      none    none
tcpsock->02 UP     IPv4      none    172.16.50.21->172.16.50.22

AIX720_LPM2:/ # clctrl -tune -L
NAME          DEF   MIN   MAX   UNIT      SCOPE
...
node_timeout   20000 10000 600000 milliseconds c n
    LPMCluster(11403f34-c4b7-11e5-8014-56c6a3855d04)           30000
...
--> Current node_timeout is 30s
-----
```

RSCT cthags status

Example 8-4 shows the current RSCT cthags service's status.

Example 8-4 RSCT cthags service's status

```

AIX720_LPM1:/ # lssrc -ls cthags
Subsystem      Group          PID      Status
  cthags       cthags        13173166  active
5 locally-connected clients. Their PIDs:
9175342(IBM.ConfigRMd) 6619600(rmcld) 14549496(IBM.StorageRMd) 7995658(clstrmgr)
10355040(gsclvmd)
HA Group Services domain information:
Domain established by node 1
Number of groups known locally: 8
                                         Number of      Number of local
Group name      providers      providers/subscribers
rmc_peers       2             1             0
s00VOCKI0009G000001A9UHPVQ4      2             1             0
IBM.ConfigRM    2             1             0
IBM.StorageRM.v1 2             1             0
CLRESMGRD_1495882547      2             1             0
CLRESMGRDNPD_1495882547      2             1             0
CLSTRMGR_1495882547      2             1             0
d00VOCKI0009G000001A9UHPVQ4      2             1             0
Critical clients will be terminated if unresponsive
```

Dead Man Switch Enabled

```

AIX720_LPM1:/usr/sbin/rsct/bin/dms # ./listdms -s cthags
Dead Man Switch Enabled:
  reset interval = 3 seconds
  trip  interval = 30 seconds
```

LPAR and server location information

Example 8-5 shows the current LPAR's location information.

Example 8-5 LPAR and server location information

```
AIX720_LPM1:/ # prtconf
System Model: IBM,9179-MHD
Machine Serial Number: 060COAT --> this server is P780_09

AIX720_LPM2:/ # prtconf
System Model: IBM,9179-MHD
Machine Serial Number: 061949T --> this server is P780_10
```

8.3.3 Manual pre-Live Partition Mobility operations

Before performing the LPM operation, there are several manual operations that are required.

Unmanaging resource groups

There are two methods to change the PowerHA service to the Unmanage Resource Group status. The first method is through the SMIT menu (accessed by running **smit s1stop**), as shown in Example 8-6.

Example 8-6 Change the cluster service to unmanage resource groups through the SMIT menu

```
Stop Cluster Services

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

[Entry Fields]
* Stop now, on system restart or both
  Stop Cluster Services on these nodes
    BROADCAST cluster shutdown?
* Select an Action on Resource Groups
  now
  [AIX720_LPM1]
  true
  Unmanage Resource Groups
```

The second method is through the **clmgr** command, as shown in Example 8-7.

Example 8-7 Change the cluster service to unmanage a resource group through the clmgr command

```
AIX720_LPM1:/ # clmgr stop node AIX720_LPM1 WHEN=now MANAGE=unmanage
Broadcast message from root@AIX720_LPM1 (tty) at 23:52:44 ...
PowerHA SystemMirror on AIX720_LPM1 shutting down. Please exit any cluster
applications...
AIX720_LPM1: 0513-044 The clevmgrdES Subsystem was requested to stop.

.
"AIX720_LPM1" is now unmanaged.
AIX720_LPM1: Jan 26 2016 23:52:43 /usr/es/sbin/cluster/utilities/clstop: called
with flags -N -f

AIX720_LPM1:/ # clcmd -n LPMCluster clRGinfo
-----
NODE AIX720_LPM2
-----
-----
```

Group Name	State	Node

testRG	UNMANAGED	AIX720_LPM1
	UNMANAGED	AIX720_LPM2
<hr/>		
NODE AIX720_LPM1		
<hr/>		
Group Name	State	Node
<hr/>		
testRG	UNMANAGED	AIX720_LPM1
	UNMANAGED	AIX720_LPM2

Disabling cthags monitoring

Example 8-8 shows how to disable the RSCT **cthags** critical resource monitoring function to prevent a DMS trigger if the LPM freeze time is longer than its timeout.

Note: In this case, there are *only* two nodes in this cluster, so you must disable this function on both nodes. Only one node is shown in the example, but the command is run on both nodes.

Example 8-8 Disabling the RSCT cthags critical resource monitoring function

```
AIX720_LPM1:/ # /usr/sbin/rsct/bin/hags_disable_client_kill -s cthags
AIX720_LPM1:/ # /usr/sbin/rsct/bin/dms/stopdms -s cthags

Dead Man Switch Disabled
DMS Re-arm Thread cancelled

AIX720_LPM1:/ # lssrc -ls cthags
Subsystem      Group          PID      Status
  cthags        cthags       13173166    active
5 locally-connected clients. Their PIDs:
9175342(IBM.ConfigRMd) 6619600(rmcd) 14549496(IBM.StorageRMd) 19792370(clstrmgr)
19268008(gsclvmd)
HA Group Services domain information:
Domain established by node 1
Number of groups known locally: 8
          Number of   Number of local
Group name     providers   providers/subscribers
rmc_peers      2           1           0
s00VOCKI0009G000001A9UHPVQ4      2           1           0
IBM.ConfigRM    2           1           0
IBM.StorageRM.v1  2           1           0
CLRESMGRD_1495882547    2           1           0
CLRESMGRDNPD_1495882547    2           1           0
CLSTRMGR_1495882547    2           1           0
d00VOCKI0009G000001A9UHPVQ4      2           1           0
```

Critical clients will not be terminated even if unresponsive

Dead Man Switch Disabled

```
AIX720_LPM1:/usr/sbin/rsct/bin/dms # ./listdms -s cthags
```

Dead Man Switch Disabled

Increasing the Cluster Aware AIX node_timeout parameter

Example 8-9 shows how to increase the CAA node_timeout parameter to prevent a CAA DMS trigger if the LPM freeze time is longer than its timeout. You must run this command on only one node because it is cluster aware.

Example 8-9 Increasing the CAA node_timeout parameter

```
AIX720_LPM1:/ # clmgr -f modify cluster HEARTBEAT_FREQUENCY="600"
1 tunable updated on cluster LPMCluster.

AIX720_LPM1:/ # clctr1 -tune -L
NAME           DEF    MIN    MAX    UNIT      SCOPE
ENTITY_NAME(UUID)                               CUR
...
node_timeout      20000 10000 600000 milliseconds c n
LPMCluster(11403f34-c4b7-11e5-8014-56c6a3855d04)          600000
```

Note: With the previous configuration, if LPM's freeze time is longer than 600 seconds, CAA DMS is triggered because of the CAA's `deadman_mode=a` (assert) parameter. The node crashes and its RG is moved to another node.

Note: The `-f` option of the `clmgr` command means not to update the HACMPcluster ODM because it updates the CAA parameter (`node_timeout`) directly with the `clctr1` command. This function is included with the following interim fixes:

- ▶ PowerHA SystemMirror Version 7.1.2 - IV79502 (SP8)
- ▶ PowerHA SystemMirror Version 7.1.3 - IV79497 (SP5)

If you do not apply one of these interim fixes, then you must perform four steps to increase the CAA `node_timeout` variable (Example 8-10):

1. Change the PowerHA service to `online` status (because cluster sync needs this status).
2. Change the HACMPcluster ODM.
3. Perform cluster verification and synchronization.
4. Change the PowerHA service to `unmanage` resource group status.

Example 8-10 Detailed steps to change the CAA node_timeout parameter without PowerHA interim fix

--> Step 1

```
AIX720_LPM1:/ # clmgr start node AIX720_LPM1 WHEN=now MANAGE=auto
Adding any necessary PowerHA SystemMirror entries to /etc/inittab and /etc/rc.net
for IPAT on node AIX720_LPM1.
AIX720_LPM1: start_cluster: Starting PowerHA SystemMirror
...
"AIX720_LPM1" is now online.
Starting Cluster Services on node: AIX720_LPM1
This may take a few minutes. Please wait...
AIX720_LPM1: Jan 27 2016 06:17:04 Starting execution of
/usr/es/sbin/cluster/etc/rc.cluster
AIX720_LPM1: with parameters: -boot -N -A -b -P cl_rc_cluster
AIX720_LPM1:
AIX720_LPM1: Jan 27 2016 06:17:04 Checking for srcmstr active...
AIX720_LPM1: Jan 27 2016 06:17:04 complete.
```

--> Step 2

```
AIX720_LPM1:/ # clmgr modify cluster HEARTBEAT_FREQUENCY="600"
```

--> Step 3

```
AIX720_LPM1:/ # clmgr sync cluster
Verifying additional prerequisites for Dynamic Reconfiguration...
...completed.

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 AIX720_LPM1.
Checking for added nodes
Updating Split Merge Policies
1 tunable updated on cluster LPMCluster.
Adding any necessary PowerHA SystemMirror entries to /etc/inittab and /etc/rc.net
for IPAT on node AIX720_LPM2.
```

Verification has completed normally.

--> Step 4

```
AIX720_LPM1:/ # clmgr stop node AIX720_LPM1 WHEN=now MANAGE=unmanage
Broadcast message from root@AIX720_LPM1 (tty) at 06:15:02 ...
PowerHA SystemMirror on AIX720_LPM1 shutting down. Please exit any cluster
applications...
AIX720_LPM1: 0513-044 The clevmgrdES Subsystem was requested to stop.
.
"AIX720_LPM1" is now unmanaged.
```

--> Check the result

```
AIX720_LPM1:/ # clctrl -tune -L
NAME           DEF    MIN    MAX    UNIT      SCOPE
ENTITY_NAME(UUID)                               CUR
...
node_timeout      20000  10000  600000 milliseconds  c n
          LPMCluster(11403f34-c4b7-11e5-8014-56c6a3855d04) 600000
```

Note: When you stop the cluster with unmanage and when you start it with auto, it tries to bring the RG online, which does not cause any problem with the VGs, file systems, and IPs. However, it runs the application controller one more time. If you do not predict the appropriate *checks* in its application controller before running the commands, it can cause problems with the application. Therefore, the application controller start script checks if the application is already online before starting it.

Disabling the SAN heartbeating function

Note: In our scenario, SAN-based heartbeating is configured, so this step is required. You do not need to do this step if SAN-based heartbeating is not configured.

Example 8-11 shows how to disable SAN heartbeating function.

Example 8-11 Disabling the SAN heartbeating function

```
AIX720_LPM1:/ # echo "sfwcom" >> /etc/cluster/ifrestrict
AIX720_LPM1:/ # clusterconf

AIX720_LPM2:/ # echo "sfwcom" >> /etc/cluster/ifrestrict
```

```

AIX720_LPM2:/ # clusterconf

AIX720_LPM1:/ # clcmd lscluster -m

-----
NODE AIX720_LPM2
-----
Calling node query for all nodes...
Node query number of nodes examined: 2

    Node name: AIX720_LPM1
    Cluster shorthand id for node: 1
    UUID for node: 112552f0-c4b7-11e5-8014-56c6a3855d04
    State of node: UP
    Smoothed rtt to node: 7
    Mean Deviation in network rtt to node: 3
    Number of clusters node is a member in: 1
    CLUSTER NAME      SHID      UUID
    LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
    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->01   UP     IPv4       none    172.16.50.22->172.16.50.21
-----

    Node name: AIX720_LPM2
    Cluster shorthand id for node: 2
    UUID for node: 11255336-c4b7-11e5-8014-56c6a3855d04
    State of node: UP NODE_LOCAL
    Smoothed rtt to node: 0
    Mean Deviation in network rtt to node: 0
    Number of clusters node is a member in: 1
    CLUSTER NAME      SHID      UUID
    LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
    SITE NAME         SHID      UUID
    LOCAL             1         51735173-5173-5173-5173-517351735173

    Points of contact for node: 0
-----

NODE AIX720_LPM1
-----
Calling node query for all nodes...
Node query number of nodes examined: 2

    Node name: AIX720_LPM1
    Cluster shorthand id for node: 1
    UUID for node: 112552f0-c4b7-11e5-8014-56c6a3855d04
    State of node: UP NODE_LOCAL
    Smoothed rtt to node: 0
    Mean Deviation in network rtt to node: 0

```

```

Number of clusters node is a member in: 1
CLUSTER NAME      SHID      UUID
LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME        SHID      UUID
LOCAL            1         51735173-5173-5173-5173-517351735173

```

Points of contact for node: 0

```

Node name: AIX720_LPM2
Cluster shorthand id for node: 2
UUID for node: 11255336-c4b7-11e5-8014-56c6a3855d04
State of node: UP
Smoothed rtt to node: 18
Mean Deviation in network rtt to node: 14
Number of clusters node is a member in: 1
CLUSTER NAME      SHID      UUID
LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
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	172.16.50.21->172.16.50.22

```

AIX720_LPM1:/ # lscluster -i
Network/Storage Interface Query

Cluster Name: LPMCluster
Cluster UUID: 11403f34-c4b7-11e5-8014-56c6a3855d04
Number of nodes reporting = 2
Number of nodes stale = 0
Number of nodes expected = 2

Node AIX720_LPM1
Node UUID = 112552f0-c4b7-11e5-8014-56c6a3855d04
Number of interfaces discovered = 3
    Interface number 1, en1
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = FA:97:6D:97:2A:20
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 2
        IPv4 ADDRESS: 172.16.50.21 broadcast 172.16.50.255 netmask
255.255.255.0

```

```

IPv4 ADDRESS: 172.16.50.23 broadcast 172.16.50.255 netmask
255.255.255.0
Number of cluster multicast addresses configured on interface = 1
IPv4 MULTICAST ADDRESS: 228.16.50.21
Interface number 2, sfwcom
IFNET type = 0 (none)
NDD type = 304 (NDD_SANCOMM)
Smoothed RTT across interface = 7
Mean deviation in network RTT across interface = 3
Probe interval for interface = 990 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = DOWN RESTRICTED SOURCE HARDWARE RECEIVE SOURCE
HARDWARE TRANSMIT
Interface number 3, dpcom
IFNET type = 0 (none)
NDD type = 305 (NDD_PINGCOMM)
Smoothed RTT across interface = 750
Mean deviation in network RTT across interface = 1500
Probe interval for interface = 22500 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = UP RESTRICTED AIX_CONTROLLED

Node AIX720_LPM2
Node UUID = 11255336-c4b7-11e5-8014-56c6a3855d04
Number of interfaces discovered = 3
Interface number 1, en1
IFNET type = 6 (IFT_ETHER)
NDD type = 7 (NDD_IS088023)
MAC address length = 6
MAC address = FA:F2:D3:29:50:20
Smoothed RTT across interface = 0
Mean deviation in network RTT across interface = 0
Probe interval for interface = 990 ms
IFNET flags for interface = 0x1E084863
NDD flags for interface = 0x0021081B
Interface state = UP
Number of regular addresses configured on interface = 1
IPv4 ADDRESS: 172.16.50.22 broadcast 172.16.50.255 netmask
255.255.255.0
Number of cluster multicast addresses configured on interface = 1
IPv4 MULTICAST ADDRESS: 228.16.50.21
Interface number 2, sfwcom
IFNET type = 0 (none)
NDD type = 304 (NDD_SANCOMM)
Smoothed RTT across interface = 7
Mean deviation in network RTT across interface = 3
Probe interval for interface = 990 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = DOWN RESTRICTED SOURCE HARDWARE RECEIVE SOURCE
HARDWARE TRANSMIT
Interface number 3, dpcom
IFNET type = 0 (none)

```

```
NDD type = 305 (NDD_PINGCOMM)
Smoothed RTT across interface = 750
Mean deviation in network RTT across interface = 1500
Probe interval for interface = 22500 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = UP RESTRICTED AIX_CONTROLLED
```

8.3.4 Performing Live Partition Mobility

Example 8-12 shows how to perform the LPM operation for the AIX720_LPM1 node. This operation migrates this LPAR from P780_09 to P780_10.

Example 8-12 Performing the Live Partition Mobility operation

```
hscroot@hmc55:~> time migrlpar -o m -m SVRP7780-09-SN060C0AT -t
SVRP7780-10-SN061949T -p AIX720_LPM1
```

```
real    1m6.269s
user    0m0.001s
sys     0m0.000s
```

PowerHA service and resource group status

After LPM completes, Example 8-13 shows that the PowerHA services are still stable, and AIX720_LPM1 is moved to the P780_10 server.

Example 8-13 PowerHA services stable

```
AIX720_LPM1:/ # clcmd -n LPMCluster lssrc -ls clstrmgrES|egrep "NODE|state"|grep
-v "Last"
NODE AIX720_LPM2
Current state: ST_STABLE
NODE AIX720_LPM1
Current state: ST_STABLE

AIX720_LPM1:/ # prtconf
System Model: IBM,9179-MHD
Machine Serial Number: 061949T --> this server is P780_10

AIX720_LPM2:/ # prtconf|more
System Model: IBM,9179-MHD
Machine Serial Number: 061949T --> this server is P780_10
```

8.3.5 Manual post-Live Partition Mobility operations

Upon LPM completion, there are several manual operations that are required.

Enable SAN heartbeating function

Example 8-14 shows how to enable the SAN heartbeating function.

Example 8-14 Enabling the SAN heartbeating function

```
AIX720_LPM1:/ # rm /etc/cluster/ifrestrict
AIX720_LPM1:/ # clusterconf

AIX720_LPM2:/ # rm /etc/cluster/ifrestrict
AIX720_LPM2:/ # clusterconf

AIX720_LPM1:/ # clcmd lscluster -m

-----
NODE AIX720_LPM2
-----
Calling node query for all nodes...
Node query number of nodes examined: 2

    Node name: AIX720_LPM1
    Cluster shorthand id for node: 1
    UUID for node: 112552f0-c4b7-11e5-8014-56c6a3855d04
    State of node: UP
    Smoothed rtt to node: 7
    Mean Deviation in network rtt to node: 3
    Number of clusters node is a member in: 1
    CLUSTER NAME      SHID      UUID
    LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
    SITE NAME         SHID      UUID
    LOCAL             1         51735173-5173-5173-5173-517351735173

    Points of contact for node: 2
-----
Interface      State   Protocol     Status      SRC_IP->DST_IP
-----
sfwcom          UP      none        none        none
tcpsock->01    UP      IPv4        none        172.16.50.22->172.16.50.21
-----

Node name: AIX720_LPM2
Cluster shorthand id for node: 2
UUID for node: 11255336-c4b7-11e5-8014-56c6a3855d04
State of node: UP NODE_LOCAL
Smoothed rtt to node: 0
Mean Deviation in network rtt to node: 0
Number of clusters node is a member in: 1
CLUSTER NAME      SHID      UUID
LPMCluster        0         11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME         SHID      UUID
LOCAL             1         51735173-5173-5173-5173-517351735173
```

Points of contact for node: 0

NODE AIX720_LPM1

Calling node query for all nodes...
Node query number of nodes examined: 2

Node name: AIX720_LPM1
Cluster shorthand id for node: 1
UUID for node: 112552f0-c4b7-11e5-8014-56c6a3855d04
State of node: UP NODE_LOCAL
Smoothed rtt to node: 0
Mean Deviation in network rtt to node: 0
Number of clusters node is a member in: 1
CLUSTER NAME SHID UUID
LPMCluster 0 11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME SHID UUID
LOCAL 1 51735173-5173-5173-5173-517351735173

Points of contact for node: 0

Node name: AIX720_LPM2
Cluster shorthand id for node: 2
UUID for node: 11255336-c4b7-11e5-8014-56c6a3855d04
State of node: UP
Smoothed rtt to node: 16
Mean Deviation in network rtt to node: 14
Number of clusters node is a member in: 1
CLUSTER NAME SHID UUID
LPMCluster 0 11403f34-c4b7-11e5-8014-56c6a3855d04
SITE NAME SHID UUID
LOCAL 1 51735173-5173-5173-5173-517351735173

Points of contact for node: 2

Interface	State	Protocol	Status	SRC_IP->DST_IP
-----------	-------	----------	--------	----------------

sfwcom	UP	none	none	none
tcpsock->02	UP	IPv4	none	172.16.50.21->172.16.50.22

Note: After this step, if the sfwcom interface is still not UP, check the VLAN storage framework communication device's status. If it is in defined status, you must reconfigure it by running the following command:

```
AIX720_LPM1:/ # lsdev -C|grep vLAN
sfwcomm1      Defined vLAN Storage Framework Comm
AIX720_LPM1:/ # rmdev -l sfwcomm1; sleep 2; mkdev -l sfwcomm1
sfwcomm1 Defined
sfwcomm1 Available
```

Then, you can check the sfwcom interface's status again by running the **1scluster** command.

Restoring the Cluster Aware AIX node_timeout variable

Example 8-15 shows how to restore the CAA node_timeout variable.

Note: In a PowerHA cluster environment, the default value of node_timeout is 30 seconds.

Example 8-15 Restoring the Cluster Aware AIX node_timeout parameter

```
AIX720_LPM1:/ # clmgr -f modify cluster HEARTBEAT_FREQUENCY="30"
1 tunable updated on cluster LPMCluster.
```

```
AIX720_LPM1:/ # clctr1 -tune -L
NAME           DEF    MIN    MAX    UNIT      SCOPE
  ENTITY_NAME(UUID)                                CUR
...
node_timeout        20000 10000 600000 milliseconds  c n
  LPMCluster(11403f34-c4b7-11e5-8014-56c6a3855d04)          30000
```

Enabling cthags monitoring

Example 8-16 shows how to enable the RSCT cthags critical resource monitoring function.

Note: In this case, there are *only* two nodes in this cluster, so you disable the function on both nodes before LPM. Only one node is shown in this example, but the command is run on both nodes.

Example 8-16 Enabling RSCT cthags resource monitoring

```
AIX720_LPM1:/ # /usr/sbin/rsct/bin/dms/startdms -s cthags
Dead Man Switch Enabled
DMS Re-arm Thread created

AIX720_LPM1:/ # /usr/sbin/rsct/bin/hags_enable_client_kill -s cthags
AIX720_LPM1:/ # lssrc -ls cthags
Subsystem      Group      PID      Status
  cthags       cthags    13173166   active
5 locally-connected clients. Their PIDs:
9175342(IBM.ConfigRMd) 6619600(rmcd) 14549496(IBM.StorageRMd) 19792370(clstrmgr)
19268008(gsclvmd)
HA Group Services domain information:
Domain established by node 1
```

```

Number of groups known locally: 8
          Number of      Number of local
Group name       providers   providers/subscribers
rmc_peers        2           1           0
s00VOCKI0009G000001A9UHPVQ4    2           1           0
IBM.ConfigRM     2           1           0
IBM.StorageRM.v1 2           1           0
CLRESMGRD_1495882547 2           1           0
CLRESMGRDNPD_1495882547 2           1           0
CLSTRMGR_1495882547 2           1           0
d00VOCKI0009G000001A9UHPVQ4    2           1           0

```

Critical clients will be terminated if unresponsive

```

Dead Man Switch Enabled
AIX720_LPM1:/ # /usr/sbin/rsct/bin/dms/listdms -s cthags
  Dead Man Switch Enabled:
    reset interval = 3 seconds
    trip interval = 30 seconds

```

Changing the PowerHA service back to the normal status

Example 8-17 shows how to change the PowerHA service back to the normal status. There are two methods to achieve this task. One is through the SMIT menu (run **smit clstart**), as shown in Example 8-17.

Example 8-17 Changing the PowerHA service back to the normal status

Start Cluster Services

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

[Entry Fields]	
* Start now, on system restart or both	now
Start Cluster Services on these nodes	[AIX720_LPM1]
* Manage Resource Groups	Automatically
BROADCAST message at startup?	true
Startup Cluster Information Daemon?	false
Ignore verification errors?	false
Automatically correct errors found during cluster start?	Interactively

Another is through 'clmgr' command:

```

AIX720_LPM1:/ # clmgr start node AIX720_LPM1 WHEN=now MANAGE=auto
AIX720_LPM1: start_cluster: Starting PowerHA SystemMirror
...
"AIX720_LPM1" is now online.

```

```

Starting Cluster Services on node: AIX720_LPM1
This may take a few minutes. Please wait...
AIX720_LPM1: Jan 27 2016 01:04:43 Starting execution of
/usr/es/sbin/cluster/etc/rc.cluster
AIX720_LPM1: with parameters: -boot -N -A -b -P cl_rc_cluster
AIX720_LPM1:

```

```
AIX720_LPM1: Jan 27 2016 01:04:43 Checking for srcmstr active...
AIX720_LPM1: Jan 27 2016 01:04:43 complete.
```

Note: When you stop the cluster with **unmanage** and when you start it with **auto**, it tries to bring the RG online, which does not cause any problem with the VGs, file systems, and IPs. However, it runs the application controller one more time. If you do not predict the appropriate checks in its application controller before running the commands, it can cause problems with the application. Therefore, the application controller start script checks whether the application is already online before starting it.

Example 8-18 shows that the RG's status changed to normal.

Example 8-18 Resource group's status

```
AIX720_LPM1:/ # clcmd c1RGinfo

-----
NODE AIX720_LPM2
-----

Group Name          State      Node
-----
testRG             ONLINE    AIX720_LPM1
                  OFFLINE   AIX720_LPM2

-----
NODE AIX720_LPM1
-----

Group Name          State      Node
-----
testRG             ONLINE    AIX720_LPM1
                  OFFLINE   AIX720_LPM2
```

8.4 Live Partition Mobility SMIT panel

Starting with Version 7.2, PowerHA SystemMirror automates some of the LPM steps by registering a script with the LPM framework.

PowerHA SystemMirror listens to LPM events and automates steps in PowerHA SystemMirror to handle the LPAR freeze that can occur during the LPM process. As part of the automation, PowerHA SystemMirror provides a few variables that can be changed based on the requirements for your environment.

You can change the following LPM variables in PowerHA SystemMirror that provide LPM automation:

- ▶ Node Failure Detection Timeout during LPM
- ▶ LPM Node Policy

Start **smit sysmirror**. Select **Custom Cluster Configuration → Cluster Nodes and Networks → Manage the Cluster → Cluster heartbeat settings**. The next panel is a menu window with a title menu option and seven item menu options.

Its fast path is **cm_chng_tunables** (Figure 8-4). This menu is not new, but two items were added to it to make LPM easier in a PowerHA environment (the last two items are new).

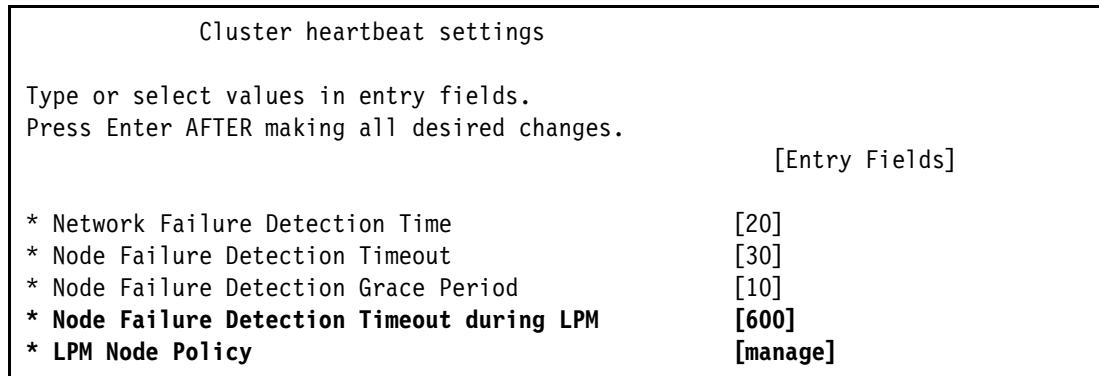


Figure 8-4 Cluster heartbeat setting

Table 8-4 describes the context-sensitive help information for the cluster heartbeating setting.

Table 8-4 Context-sensitive help for the Cluster heartbeat setting

Name and fast path	Context-sensitive help (F1)
Node Failure Detection Timeout during LPM	If specified, this timeout value (in seconds) is used during an 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. The unit is second. For PowerHA V7.2 GA Edition, the customer can enter a value 10 - 600. For PowerHA V7.2 SP1 or later, the default is 600 and is unchangeable.
LPM Node Policy	Specifies the action to be taken on the node during an LPM operation. If unmanage is selected, the cluster services are stopped with the Unmanage Resource Groups option during the duration of the LPM operation. Otherwise, PowerHA SystemMirror continues to monitor the RGs and application availability. The default is manage.

8.5 PowerHA V7.2 scenario and troubleshooting

This scenario uses the same test cluster as shown in 8.3, “Example: Live Partition Mobility scenario for PowerHA V7.1” on page 279. This scenario replaces only the PowerHA *software* with Version 7.2.

Example 8-19 shows the PowerHA version.

Example 8-19 PowerHA version

```
AIX720_LPM1:/ #clhaver
Node AIX720_LPM1 has HACMP version 7200 installed
Node AIX720_LPM2 has HACMP version 7200 installed
```

Table 8-5 shows the variables of LPM.

Table 8-5 Cluster heartbeating setting

Items	Value
Node Failure Detection Timeout during LPM	600
LPM Node Policy	unmanage

8.5.1 Troubleshooting

The PowerHA log that is related to LPM operation is in /var/hacmp/log/clutils.log. Example 8-20 and Example 8-21 on page 299 show the information in this log file, and includes pre-migration and post-migration.

Note: During the operation, PowerHA SystemMirror stops the cluster with the **unmanage** option in the pre-migration stage, and starts it with the **auto** option in the post-migration stage automatically. PowerHA SystemMirror tries to bring the RG online in the post-migration stage, which does not cause any problem with the VGs, file systems, and IPs. However, it runs the application controller one more time.

If you do not perform the appropriate checks in the application controller before running the commands, it can cause problems with the application. Therefore, the application controller start script checks whether the application is already online before starting it.

Example 8-20 Log file of the pre-migration operation

```
...
--> Check whether need to change PowerHA service to 'unmanage resource group'
status
Tue Jan 26 10:57:08 UTC 2016 cl_dr: clodmget -n -f lpm_policy HACMPcluster
Tue Jan 26 10:57:08 UTC 2016 cl_dr: lpm_policy='UNMANAGE'
...
Tue Jan 26 10:57:09 UTC 2016 cl_dr: Node = AIX720_LPM1, state = NORMAL
Tue Jan 26 10:57:09 UTC 2016 cl_dr: Stop cluster services
Tue Jan 26 10:57:09 UTC 2016 cl_dr: LC_ALL=C clmgr stop node AIX720_LPM1 WHEN=now
MANAGE=unmanage
...
"AIX720_LPM1" is now unmanaged.
...
--> Add an entry in /etc/inittab to ensure PowerHA to be in 'manage resource
group' status after crash unexpectedly
Tue Jan 26 10:57:23 UTC 2016 cl_dr: Adding a temporary entry in /etc/inittab
Tue Jan 26 10:57:23 UTC 2016 cl_dr: lsitab hacmp_lpm
Tue Jan 26 10:57:23 UTC 2016 cl_dr: mkitab
hacmp_lpm:2:once:/usr/es/sbin/cluster/utilities/cl_dr undopremigrate > /dev/null
2>&1
Tue Jan 26 10:57:23 UTC 2016 cl_dr: mkitab RC: 0
...
--> Stop RSCT cthags critical resource monitoring function (for two nodes)
Tue Jan 26 10:57:30 UTC 2016 cl_dr: Stopping RSCT Dead Man Switch on node
'AIX720_LPM1'
Tue Jan 26 10:57:30 UTC 2016 cl_dr: /usr/sbin/rsct/bin/dms/stopdms -s cthags

Dead Man Switch Disabled
```

DMS Re-arming Thread cancelled

```
Tue Jan 26 10:57:30 UTC 2016 cl_dr: stopdms RC: 0
Tue Jan 26 10:57:30 UTC 2016 cl_dr: Stopping RSCT Dead Man Switch on node
'AIX720_LPM2'
Tue Jan 26 10:57:30 UTC 2016 cl_dr: cl_rsh AIX720_LPM2 "LC_ALL=C lssrc -s cthags |
grep -qw active"
Tue Jan 26 10:57:31 UTC 2016 cl_dr: cl_rsh AIX720_LPM2 lssrc RC: 0
Tue Jan 26 10:57:31 UTC 2016 cl_dr: cl_rsh AIX720_LPM2 "LC_ALL=C
/usr/sbin/rsct/bin/dms/listdms -s cthags | grep -qw Enabled"
Tue Jan 26 10:57:31 UTC 2016 cl_dr: cl_rsh AIX720_LPM2 listdms RC: 0
Tue Jan 26 10:57:31 UTC 2016 cl_dr: cl_rsh AIX720_LPM2
"/usr/sbin/rsct/bin/dms/stopdms -s cthags"
```

Dead Man Switch Disabled

DMS Re-arming Thread cancelled

```
...
--> Change CAA node_time parameter to 600s
Tue Jan 26 10:57:31 UTC 2016 cl_dr: clodmget -n -f lpm_node_timeout HACMPcluster
Tue Jan 26 10:57:31 UTC 2016 cl_dr: clodmget LPM node_timeout: 600
Tue Jan 26 10:57:31 UTC 2016 cl_dr: clctrl -tune -x node_timeout
Tue Jan 26 10:57:31 UTC 2016 cl_dr: clctrl CAA node_timeout: 30000
Tue Jan 26 10:57:31 UTC 2016 cl_dr: Changing CAA node_timeout to '600000'
Tue Jan 26 10:57:31 UTC 2016 cl_dr: clctrl -tune -o node_timeout=600000
...
--> Disable CAA SAN heartbeating (for two nodes)
Tue Jan 26 10:57:32 UTC 2016 cl_dr: cl_rsh AIX720_LPM1 "LC_ALL=C echo $fwcom >>
/etc/cluster/ifrestrict"
Tue Jan 26 10:57:32 UTC 2016 cl_dr: cl_rsh to node AIX720_LPM1 completed, RC: 0
Tue Jan 26 10:57:32 UTC 2016 cl_dr: clusterconf
Tue Jan 26 10:57:32 UTC 2016 cl_dr: clusterconf completed, RC: 0
...
Tue Jan 26 10:57:32 UTC 2016 cl_dr: cl_rsh AIX720_LPM2 "LC_ALL=C echo $fwcom >>
/etc/cluster/ifrestrict"
Tue Jan 26 10:57:33 UTC 2016 cl_dr: cl_rsh to node AIX720_LPM2 completed, RC: 0
Tue Jan 26 10:57:33 UTC 2016 cl_dr: clusterconf
Tue Jan 26 10:57:33 UTC 2016 cl_dr: clusterconf completed, RC: 0
...

```

Example 8-21 shows the log file of the post-migration operation.

Example 8-21 Log file of the post-migration operation

```
--> Change PowerHA service back to normal status
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: POST_MIGRATE entered
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: clodmget -n -f lpm_policy HACMPcluster
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: lpm_policy='UNMANAGE'
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: grep -w node_state /var/hacmp/cl_dr.state |
cut -d '=' -f2
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: Previous state = NORMAL
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: Restarting cluster services
Tue Jan 26 10:57:52 UTC 2016 cl_2dr: LC_ALL=C clmgr start node AIX720_LPM1
WHEN=now MANAGE=auto
AIX720_LPM1: start_cluster: Starting PowerHA SystemMirror
...
"AIX720_LPM1" is now online.
```

```

...
--> Remove the entry from /etc/inittab, this entry was written in pre-migration
operation
Tue Jan 26 11:00:27 UTC 2016 cl_2dr: lsitab hacmp_lpm
Tue Jan 26 11:00:27 UTC 2016 cl_2dr: Removing the temporary entry from
/etc/inittab
Tue Jan 26 11:00:27 UTC 2016 cl_2dr: rmitab hacmp_lpm
...
--> Enable RSCT cthags critical resource monitoring function (for two nodes)
Tue Jan 26 10:58:21 UTC 2016 cl_2dr: LC_ALL=C lssrc -s cthags | grep -qw active
Tue Jan 26 10:58:21 UTC 2016 cl_2dr: lssrc RC: 0
Tue Jan 26 10:58:21 UTC 2016 cl_2dr: grep -w RSCT_local_DMS_state
/var/hacmp/cl_dr.state | cut -d '=' -f2
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: previous RSCT DMS state = Enabled
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: Restarting RSCT Dead Man Switch on node
'AIX720_LPM1'
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: /usr/sbin/rsct/bin/dms/startdms -s cthags

Dead Man Switch Enabled
DMS Re-arm Thread created
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: startdms RC: 0
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: cl_rsh AIX720_LPM2 lssrc RC: 0
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: grep -w RSCT_peer_DMS_state
/var/hacmp/cl_dr.state | cut -d '=' -f2
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: previous RSCT Dead Man Switch on node
'AIX720_LPM2' = Enabled
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: Restarting RSCT Dead Man Switch on node
'AIX720_LPM2'
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: cl_rsh AIX720_LPM2
"/usr/sbin/rsct/bin/dms/startdms -s cthags"

Dead Man Switch Enabled
DMS Re-arm Thread created
...
--> Restore CAA node_timeout value
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: previous CAA node timeout = 30000
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: Restoring CAA node_timeout to '30000'
Tue Jan 26 10:58:22 UTC 2016 cl_2dr: clctrl -tune -o node_timeout=30000
smcaactrl:0:[182](0.009):           Running smcaactrl at Tue Jan 26 10:58:22 UTC 2016
with the following parameters:
    -0 MOD_TUNE -P CHECK -T 2 -c 7ae36082-c418-11e5-8039-fa976d972a20 -t
    7ae36082-c418-11e5-8039-fa976d972a20,LPMCluster,0 -i -v node_timeout,600000
...
--> Enable SAN heartbeating (for two nodes)
Tue Jan 26 11:00:26 UTC 2016 cl_2dr: cl_rsh AIX720_LPM1 "if [ -s
/var/hacmp/ifrestrict ]; then mv /var/hacmp/ifrestrict /etc/cluster/ifrestrict;
else rm -f /etc/cluster/ifrestrict
; fi"
Tue Jan 26 11:00:26 UTC 2016 cl_2dr: cl_rsh to node AIX720_LPM1 completed, RC: 0
Tue Jan 26 11:00:26 UTC 2016 cl_2dr: cl_rsh AIX720_LPM2 "if [ -s
/var/hacmp/ifrestrict ]; then mv /var/hacmp/ifrestrict /etc/cluster/ifrestrict;
else rm -f /etc/cluster/ifrestrict
; fi"
Tue Jan 26 11:00:26 UTC 2016 cl_2dr: cl_rsh to node AIX720_LPM2 completed, RC: 0
Tue Jan 26 11:00:26 UTC 2016 cl_2dr: clusterconf
```

```
Tue Jan 26 11:00:27 UTC 2016 cl_2dr: clusterconf completed, RC: 0
Tue Jan 26 11:00:27 UTC 2016 cl_2dr: Launch the SAN communication reconfiguration
in background.
```

```
...
```



IBM PowerHA SystemMirror User Interface

This chapter covers the following areas of the IBM PowerHA SystemMirror User Interface (SMUI):

- ▶ Installation:
 - Planning
 - SMUI Client: Cluster nodes
 - SMUI server
 - Adding and removing clusters
- ▶ Navigating:
 - Event summaries
 - Log files
 - General
 - Network
 - Terminal session
- ▶ Troubleshooting:
 - Log files
 - Login problems
 - Adding clusters
 - Status not updating

9.1 Introduction

New in PowerHA SystemMirror Version 7.2.1 is a browser-based GUI to monitor your cluster environment, which is called the SMUI.

Highlights

The SMUI provides the following advantages over the PowerHA SystemMirror command line:

- ▶ Monitors the status for all clusters, sites, nodes, and resource groups (RGs) in your environment.
- ▶ Scans event summaries and read a detailed description for each event. If the event occurred because of an error or issue in your environment, you can read suggested solutions to fix the problem.
- ▶ Searches and compares log files. There are predefined search terms along with the ability to enter your own:
 - Error
 - Fail
 - Could not

Also, the format of the log file is easy to read and identify important information. While viewing any log that has multiple versions, such as hacmp.out and hacmp.out.1, they are merged together into a single log.

The logs include:

- hacmp.out
- cluster.log
- clutils.log
- clstrmgr.debug
- syslog.caa
- clverify.log
- autoverify.log

- ▶ View properties for a cluster, such as the PowerHA SystemMirror version, name of sites and nodes, and repository disk information.

Filesets

The SMUI consists of the following filesets:

cluster.es.smui.agent	This fileset installs the agent files. Installing this fileset does not start the agent. This fileset is automatically installed when you use the <code>smit install_all</code> command to install PowerHA SystemMirror Version 7.2.1 or later. This fileset is automatically installed when you add clusters to the PowerHA SystemMirror GUI.
cluster.es.smui.common	This fileset installs common files that are required by both the agent and the PowerHA SystemMirror GUI server. The Node.js files are an example of common files. This fileset is automatically installed when you use the <code>smit install_all</code> command to install PowerHA SystemMirror Version 7.2.1 or later.

<code>cluster.es.smui.server</code>	This fileset installs the PowerHA SystemMirror GUI server files. The node on which you install the <code>cluster.es.smui.server</code> fileset is known as the PowerHA SystemMirror GUI server. Installing this fileset does not start the server. You do not need to install this fileset on every node in the cluster. You can install this fileset on a single node to manage multiple clusters. Also, you do not have to install it on <i>any</i> cluster node as you can also install it on a stand-alone AIX system.
-------------------------------------	--

9.2 Installation

Before installing the SMUI, your environment must meet certain requirements, as explained in 9.2.1, “Planning” on page 305.

Tip: A demonstration of installing the PowerHA SystemMirror User Interface by using a server that has no internet access is demonstrated in this [YouTube video](#).

9.2.1 Planning

To use the SMUI, proper planning is necessary. The cluster nodes and SMUI server must be at one of the following levels of AIX:

- ▶ IBM AIX 6.1 with Technology Level 9 with Service Pack 15 or later
- ▶ IBM AIX 7.1 with Technology Level 3 or later
- ▶ IBM AIX Version 7.2 or later

Also, these additional cluster filesets must be installed on *all* nodes in the cluster:

- ▶ `cluster.es.smui.agent`
- ▶ `cluster.es.smui.common`

For the SMUI server, these filesets must be installed:

- ▶ `cluster.es.smui.common`
- ▶ `cluster.es.smui.server`

All of these filesets are available in the PowerHA SystemMirror 7.2.1 installation media.

Although the SMUI server *can* be a cluster node, it is expected to be on a separate stand-alone AIX system. Also, ideally, the SMUI server should have internet access to download additional open source packages as required. However, this section describes how to work around this requirement.

The SMUI is supported only on the following web browsers:

- ▶ Google Chrome Version 50 or later
- ▶ Firefox Version 45 or later

9.2.2 SMUI Client: Cluster nodes

For all client cluster nodes, the only thing that must be done is install the two additional filesets `cluster.es.smui.agent` and `cluster.es.smui.common`. To install them, complete the following steps:

1. From the command line, run the `smit install_all` command.
2. Specify the input device or directory that contains the filesets.
3. Select the filesets from the list.
4. Press Enter to install the fileset.

9.2.3 SMUI server

For the SMUI server, two additional filesets, `cluster.es.smui.common` and `cluster.es.smui.server`, must be installed, followed by the execution of `smuiinst.ksh` script. The `smuiinst.ksh` command automatically downloads and installs the remaining files that are required to complete the PowerHA SystemMirror GUI installation process. These downloaded files are not shipped in the filesets because the files are licensed under the General Public License (GPL).

To install, complete the following steps:

1. From the command line, run the `smit install_all` command.
2. Specify the input device or directory that contains the filesets.
3. Select the filesets from the list.
4. Press Enter to install the fileset.

The PowerHA SystemMirror GUI server should have internet access to run the `smuiinst.ksh` command. However, if the server does not have internet access, complete the following steps:

1. Copy the `smuiinst.ksh` file from the node to a system that is running the AIX operating system that has internet access. In our case, we copy it to our NIM server.
2. Run the `smuiinst.ksh -d /directory` command, where `/directory` is the location where you want to download the files. We saved it a directory that was also NFS exported to our SMUI server.

The following additional packages were downloaded:

- `bash-4.2-5.aix5.3.ppc.rpm`
- `cpio-2.11-2.aix6.1.ppc.rpm`
- `gettext-0.17-6.aix5.3.ppc.rpm`
- `info-4.13-3.aix5.3.ppc.rpm`
- `libgcc-4.9.2-1.aix6.1.ppc.rpm`
- `libgcc-4.9.2-1.aix7.1.ppc.rpm`
- `libiconv-1.13.1-2.aix5.3.ppc.rpm`
- `libstdc++-4.9.2-1.aix6.1.ppc.rpm`
- `libstdc++-4.9.2-1.aix7.1.ppc.rpm`
- `readline-6.2-2.aix5.3.ppc.rpm`

3. Copy the downloaded files to a directory on the PowerHA SystemMirror GUI server. In our case, we use an NFS-mounted directory, so we skip this step.
4. From the PowerHA SystemMirror GUI server, run the `smuiinst.ksh -i /directory` command, where `/directory` is the location where you copied the downloaded files.

During the **smuiinst.ksh** execution, the rpms are installed, the SMUI server service and uiserver are started, and a message displays a URL for the PowerHA SystemMirror GUI server similar to what is shown in Figure 9-1. Enter the specified URL into a web browser and the SMUI login window is displayed, as shown in Figure 9-2.

```
# /usr/es/sbin/cluster/ui/server/bin/smuiinst.ksh -i ./
Attempting to install any needed prerequisites.

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/0.13.0-ddl.sql"...
The database is now configured.

Attempting to start the server...
The server was successfully started.

The installation completed successfully. To use the PowerHA SystemMirror GUI,
open a web browser and enter the following URL:

https://shawnssmui.cleartechnologies.net:8080/#/login

After you log in, you can add existing clusters in your environment to the
PowerHA SystemMirror GUI.
```

Figure 9-1 SMUI server installation script output

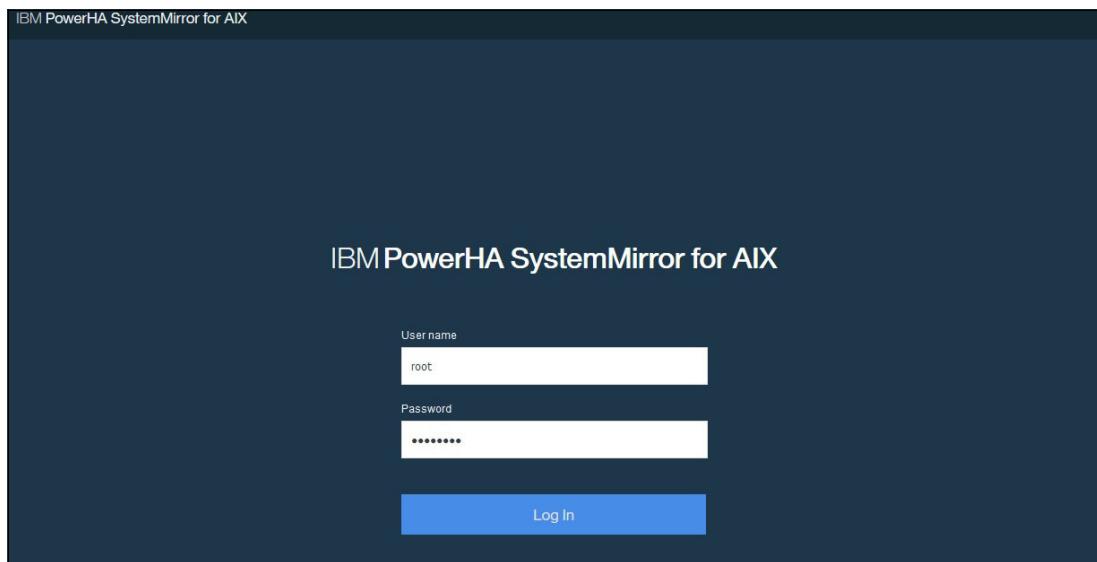


Figure 9-2 SMUI login window

9.2.4 Adding and removing clusters

After logging in, you can see text on the left frame to add a cluster. You can choose it or follow these steps:

1. Click the keypad icon in the top center of the window.
2. Click **Add clusters**.
3. Enter either host name or IP address of one of the cluster nodes, and the user and password as required.
4. Click **Discover clusters**.

These steps are shown within the SMUI in Figure 9-3.

Attention: During our testing, we had mixed results when using the host name to gather all the cluster data. However, the IP address seemed to be reliable. At the time of writing, this was a known issue by development. If you experience the same issue, then contact IBM PowerHA support.

Upon successful discovery, the cluster information is shown, and you can close the window, as shown in Figure 9-4 on page 309.

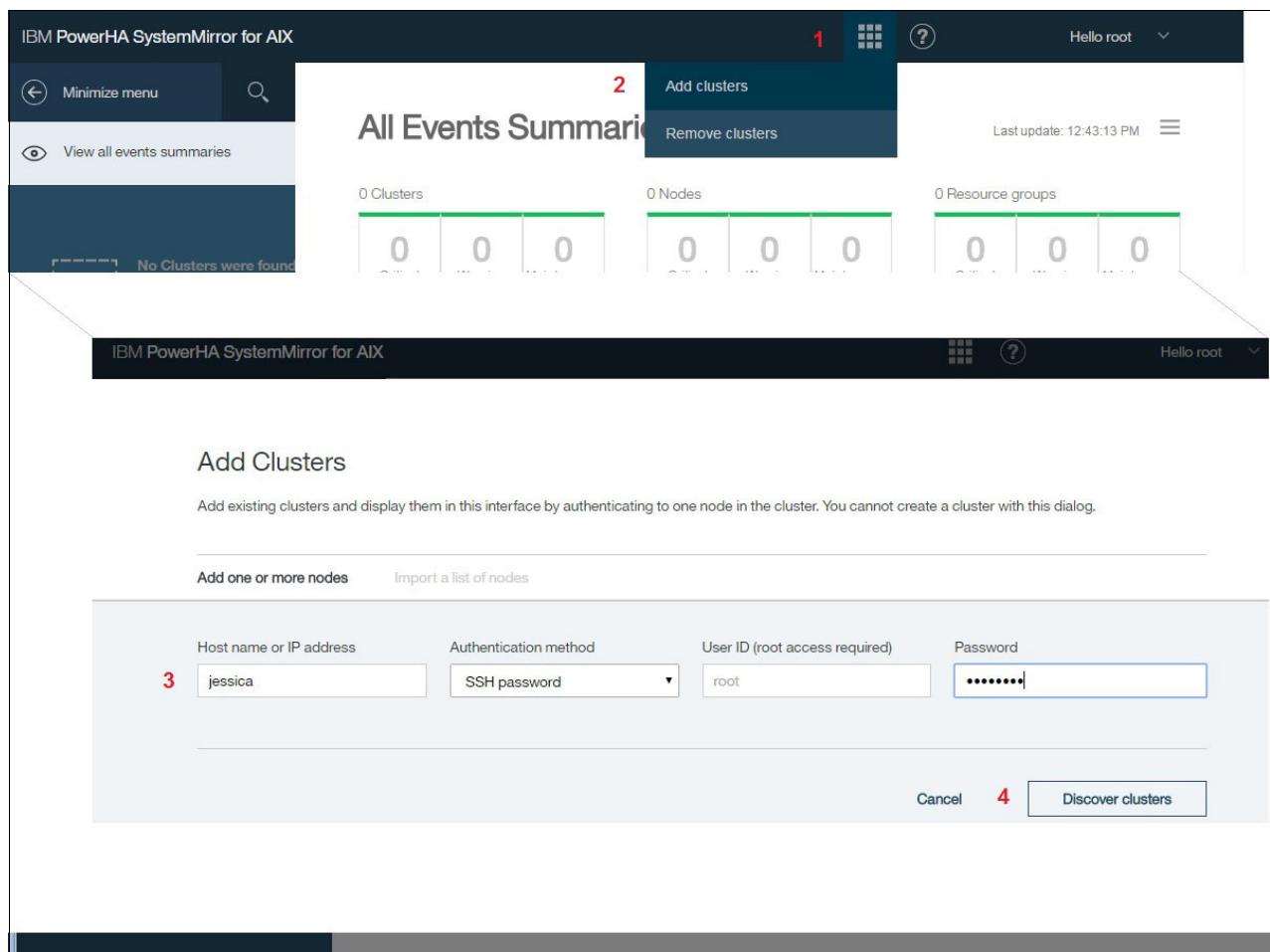


Figure 9-3 Add clusters

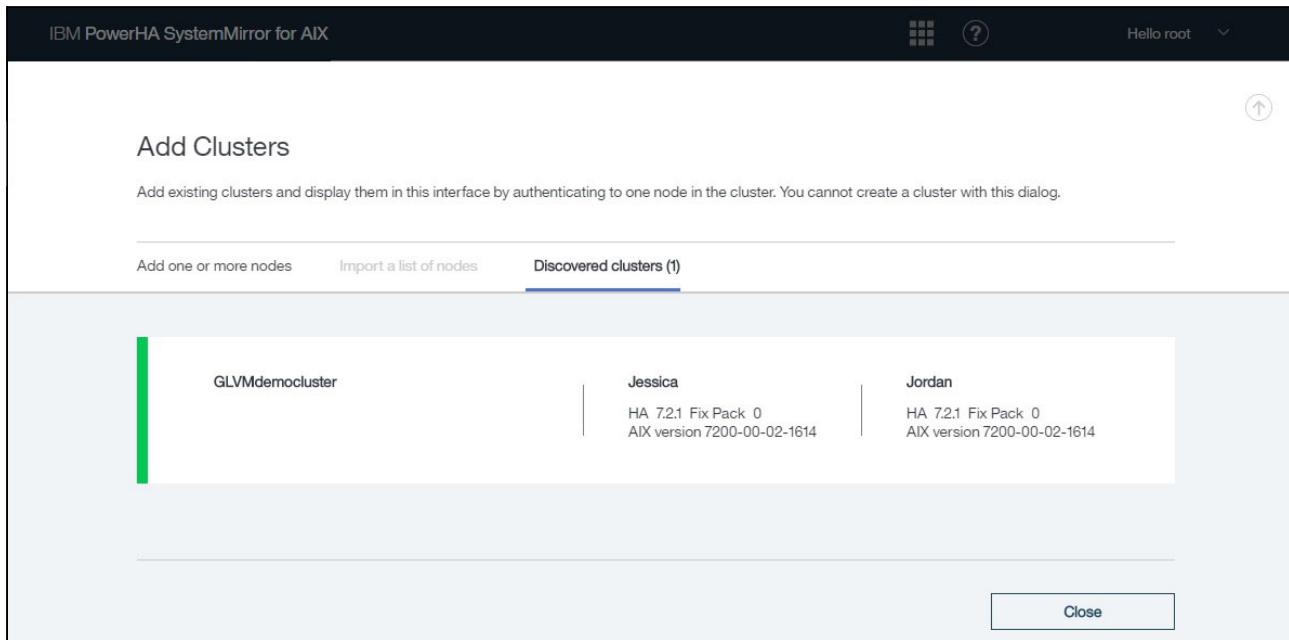


Figure 9-4 Discovered cluster

To remove a cluster:

1. Click the keypad icon in the top center of the window.
2. Click **Remove clusters**.
3. Check the box next to the correct cluster.
4. Click **Remove**.

These steps are shown within the SMUI in Figure 9-5.

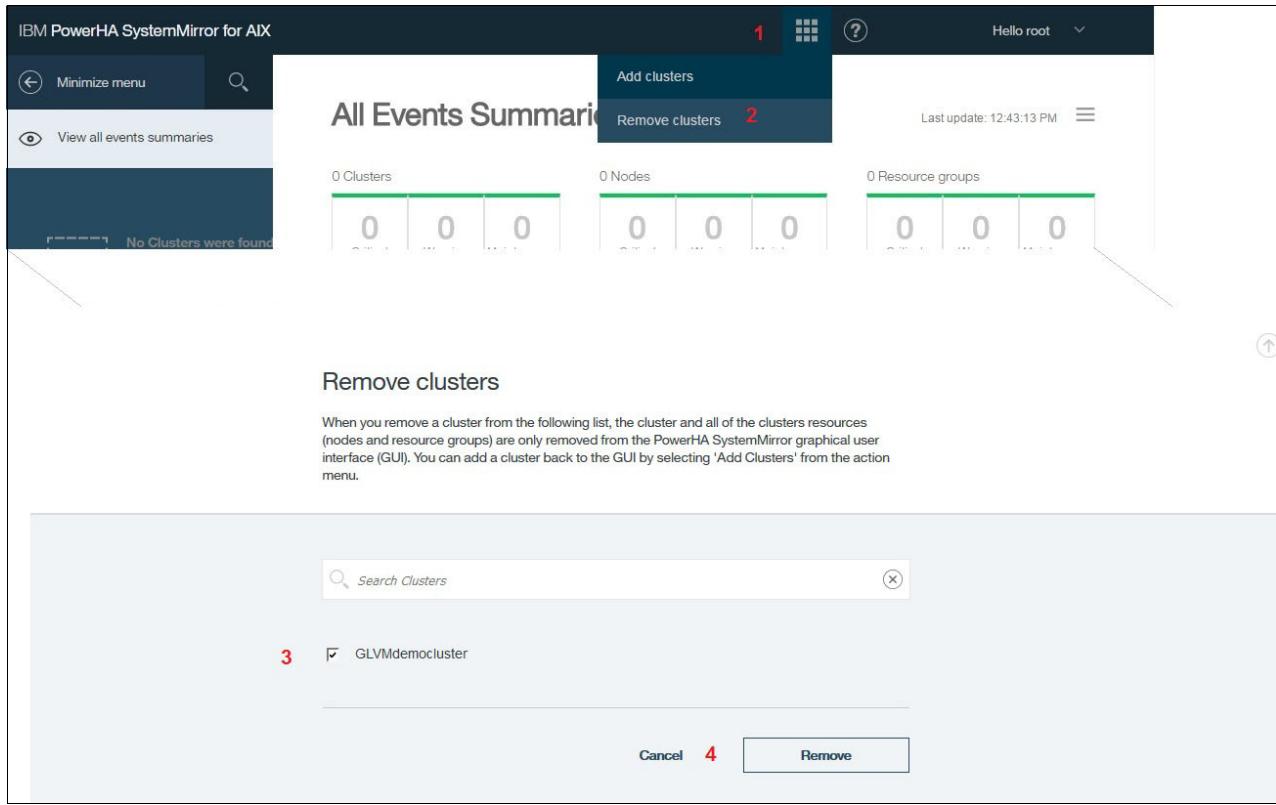


Figure 9-5 Remove clusters

9.3 Navigating

The SMUI provides a web browser interface that can monitor your PowerHA SystemMirror environment. The following sections explain and show examples of the SMUI.

9.3.1 Event summaries

In the PowerHA SystemMirror GUI, you can quickly view all events for a cluster in your environment. Figure 9-6 on page 311 identifies the different areas of the PowerHA SystemMirror GUI that are used to view events:

1. Navigation pane

This area displays all the clusters, sites, nodes, and RGs in a hierarchy that is discovered by the PowerHA SystemMirror GUI. You can click to view resources for each cluster.

Notice that the clusters are displayed in alphabetical order. However, any clusters that are in a Critical or Warning state are listed at the top of the list.

2. Scoreboard

This area displays the number of clusters, nodes, and RGs that are in a Critical, Warning, or Maintenance state. You can click **Critical**, **Warning**, or **Maintenance** to view all the messages for a specified resource. For example, in Figure 9-6 on page 311, there are six RGs that are identified. If the warning icon is highlighted and you click the warning icon, all messages (critical, warning, and normal) for the RGs are displayed.

3. Event filter

In this area, you can click the icons to display all events in your environment that correspond to a specific state. You can also search for specific event names.

4. Event timeline

This area displays events across a timeline of when the event occurred. This area allows you to view the progression of events that lead to a problem. You can zoom in and out of the time range by using the + or – keys or by using the mouse scroll wheel.

5. Event list

This area displays the name of the event, the time when each event occurred, and a description of the event. The information that is displayed in this area corresponds to the events that you selected from the event timeline area. The most recent event that occurred is displayed first. You can click this area to display more detailed information about the event, such as possible causes and suggested actions.

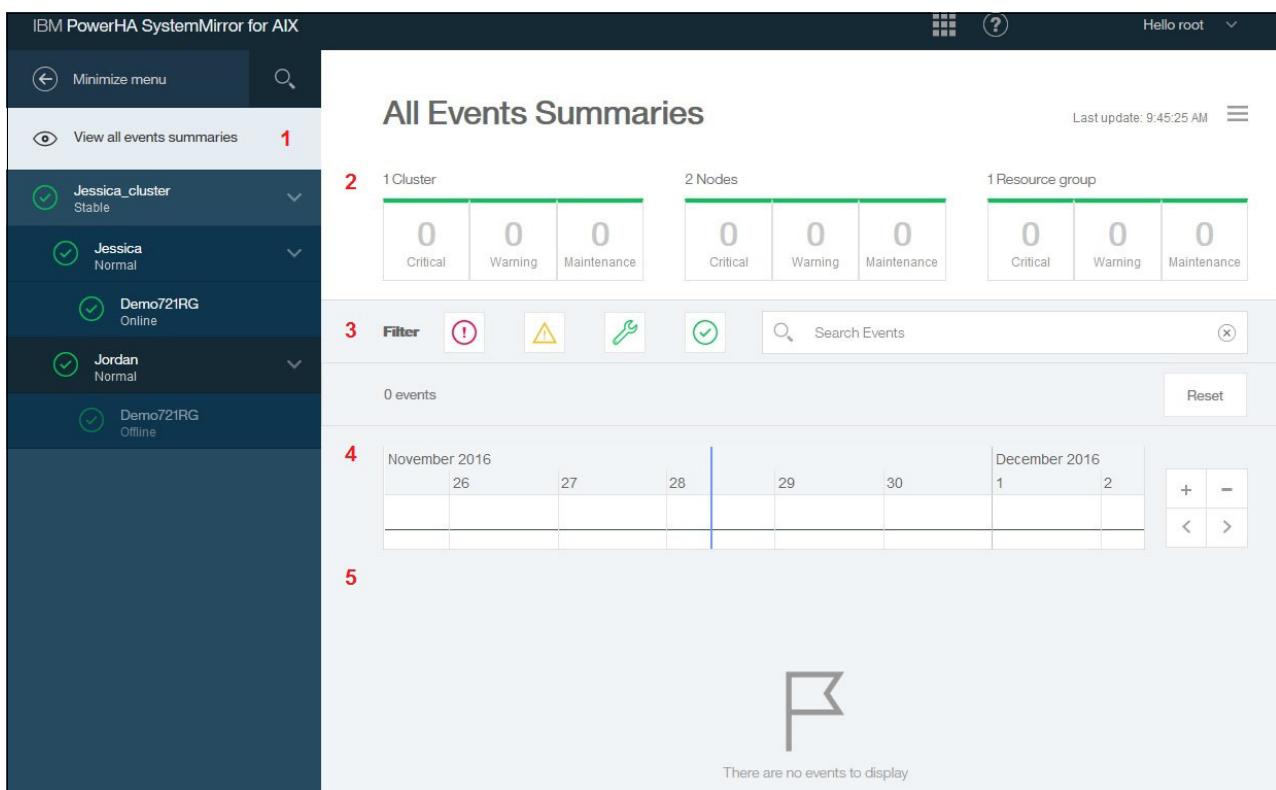


Figure 9-6 Event summaries

You can customize the view that is shown in Figure 9-7 by hiding either the score board, as shown in Figure 9-8, or hiding the timeline, as shown in Figure 9-9 on page 313.

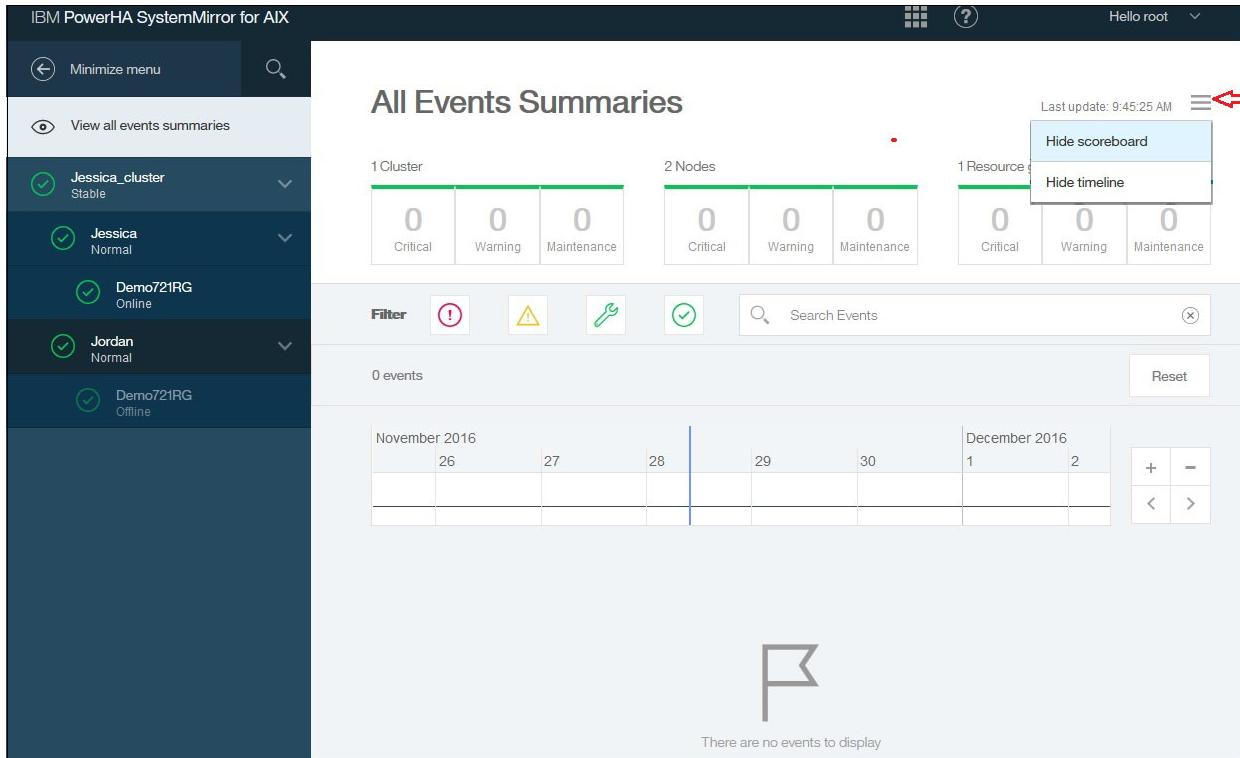


Figure 9-7 Customize to no scoreboard

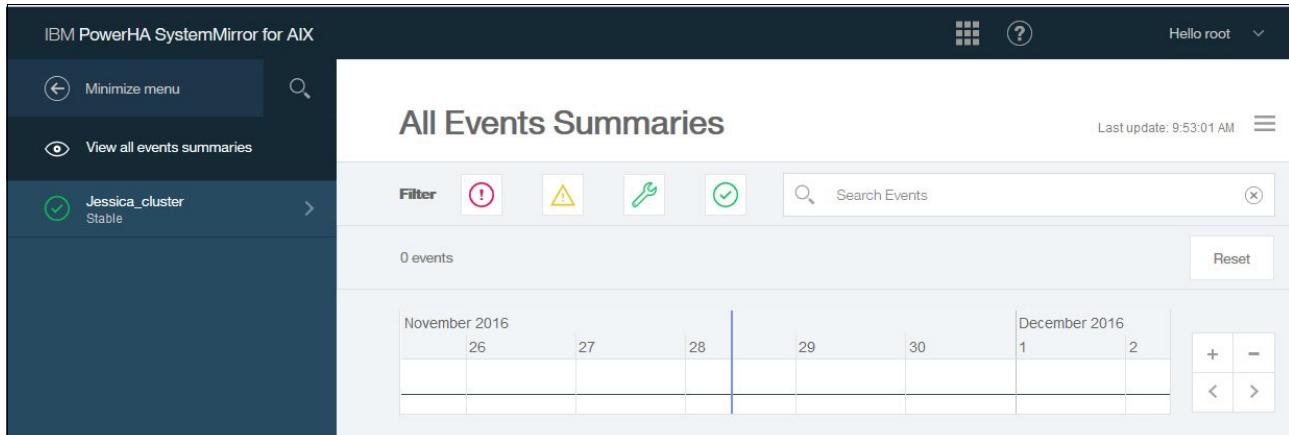


Figure 9-8 No scoreboard view

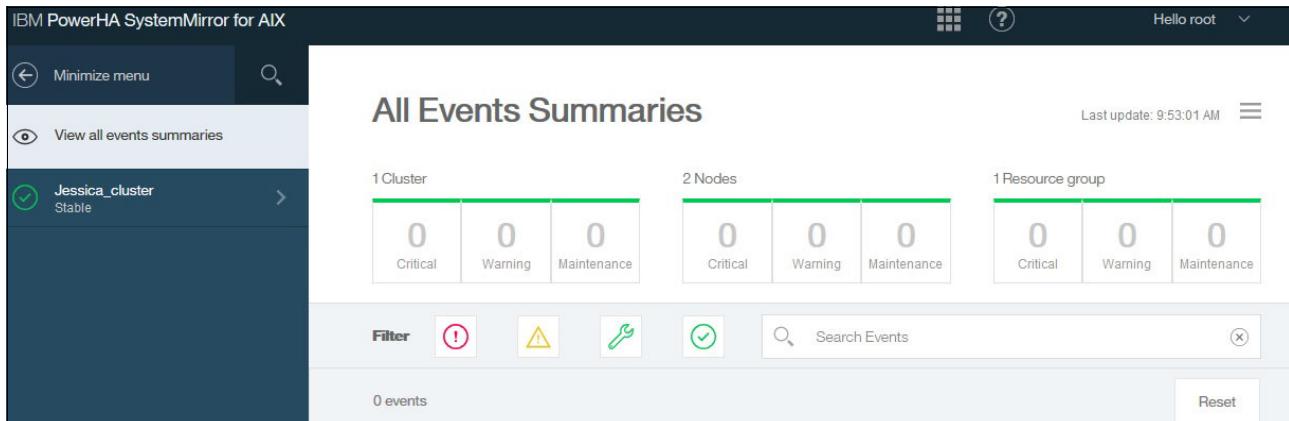


Figure 9-9 No timeline view

9.3.2 Log files

To easily compare and identify the log files that are displayed in the PowerHA SystemMirror GUI, the log files correspond to a particular color. For example, in Figure 9-10, all the log files for the `hacmp.out` file are displayed in a blue color, and all the log files for the `cluster.log` file are displayed in a yellow color.

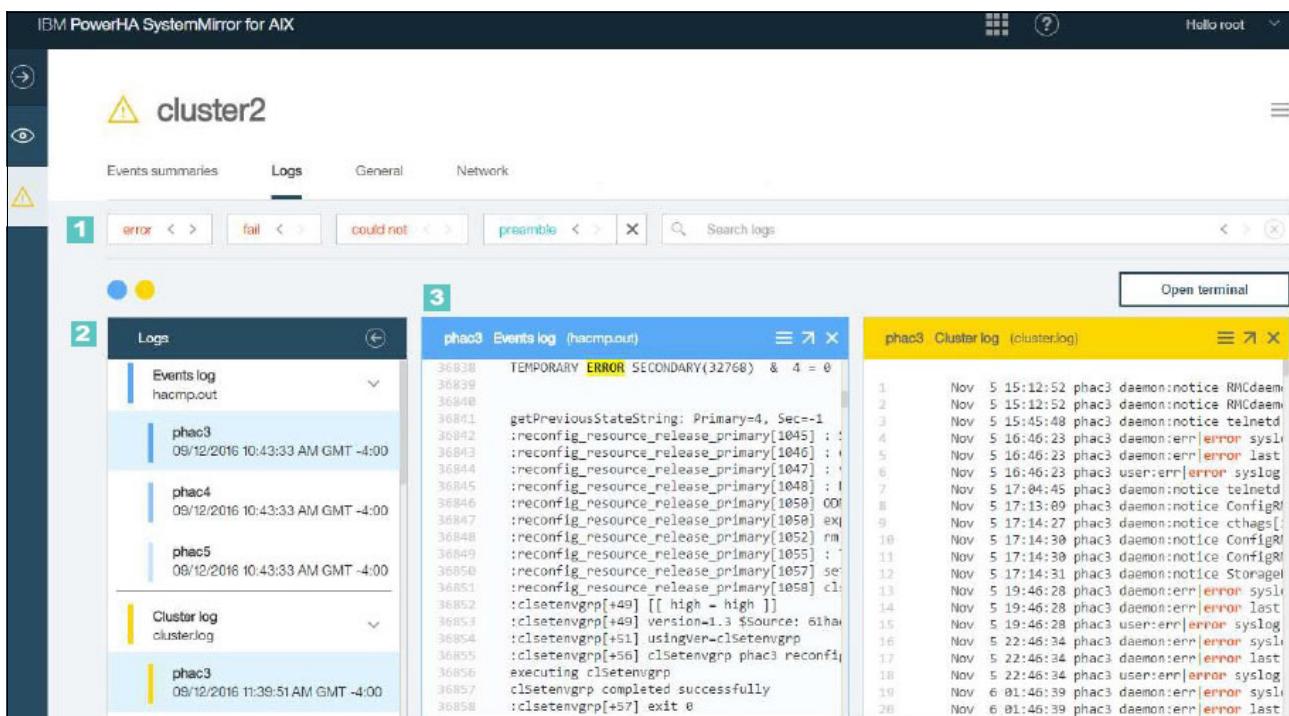


Figure 9-10 Log viewing

1. You click the following predefined search terms to locate the specified term in the log file:

- Error
- Fail
- Could not

You can click the < and > arrows to move to the previous and next instance of the search term in the selected log file. You can also enter your own search term and create a user-defined search term. A user-defined search term functions similar to the predefined search terms. For example, in Figure 9-10 on page 313, preamble is a user-defined search term.

2. Log file selection

You can view the following log files:

- hacmp.out
- cluster.log
- clutils.log
- clstrmgr.debug
- syslog.caa
- clverify.log
- autoverify.log

3. Log file viewer

In this area, you can view the log file information. To easily locate important information in the log files, the scripts are within collapsed sections in the log files. You can expand sections within the log file to view more detailed scripts. You can also open the log file in a separate browser window by clicking the right up diagonal arrow.

9.3.3 General

This window gives an overall view of the cluster configuration. A small portion of it is shown in Figure 9-11. You can expand or condense any or all sections as wanted. Currently, it can only be viewed and not saved in a report. IBM intends to deliver it as a possible enhancement.

The screenshot shows the 'General' tab of the cluster configuration interface. The main panel displays the following details for the 'Jessica_cluster':

- General:**
 - Cluster name: Jessica_cluster
 - Version: 7.2.10
 - Type: NonSite
 - Edition: ENTERPRISE
 - Synchronization: Synchronized
 - Heartbeat type: UNICAST
 - Heartbeat frequency: 30
 - Nodes: Jessica (Jessica), Jordan (Jordan)
 - Controlling node: Jessica (Jessica)
- Security:** Level: DISABLED
- Tuning:** Heartbeat frequency: 30, Grace period: 10, Site heartbeat cycle: 0, Site grace period: 0, Sync file collections: 10, Event timeout: 180 seconds, Auto verify cluster config: Enabled, Config_Too_Long time limit: 360 seconds
- Split/Merge policies:** Split policy: Merge policy: Action plan: Tie breaker: Notify method: Notify interval: Maximum notifications: Default surviving site: Apply to PPRC takeover: no
- Verification:** Auto daily verification: Enabled, Verification node: Default, Verification hour: 0, Verification debugging: Enabled, Custom verification method:

The left sidebar shows the cluster structure: Jessica_cluster (Stable) with nodes Jessica (Normal) and Jordan (Normal), and a Demo72IRG (Offline) entry.

Figure 9-11 General cluster configuration

9.3.4 Network

This window displays the cluster networking topology, as shown in Figure 9-12.

The screenshot shows the IBM PowerHA SystemMirror for AIX interface. The left sidebar lists nodes: Jessica (Normal, Online), Demo721RG (Offline), Jordan (Normal, Offline), and PHA721service. The main panel is titled 'Jessica_cluster' and shows a table of network interfaces. The table has columns: Network/IP label, IP address, Interface type, Node, Interface, and Netmask. It shows one network ('net_ether_01') with three interfaces: Jessica (IP 10.2.30.172, boot, en0), Jordan (IP 10.2.30.173, boot, en0), and PHA721service (IP 192.168.1.172, service, en0). Buttons for 'Open all networks' and 'Close all networks' are at the top right of the table.

Network/IP label	IP address	Interface type	Node	Interface	Netmask
net_ether_01					
Jessica	10.2.30.172	boot	Jessica	en0	255.255.255.0
Jordan	10.2.30.173	boot	Jordan	en0	255.255.255.0
PHA721service	192.168.1.172	service	Jessica	en0	255.255.255.0

Figure 9-12 Network information

9.3.5 Terminal session

The terminal session can be accessed from any of the three previous windows: Logs, General, and Network. You can open a terminal session directly to any known cluster node. A system name, identification, and password are all required, as shown in Figure 9-13.

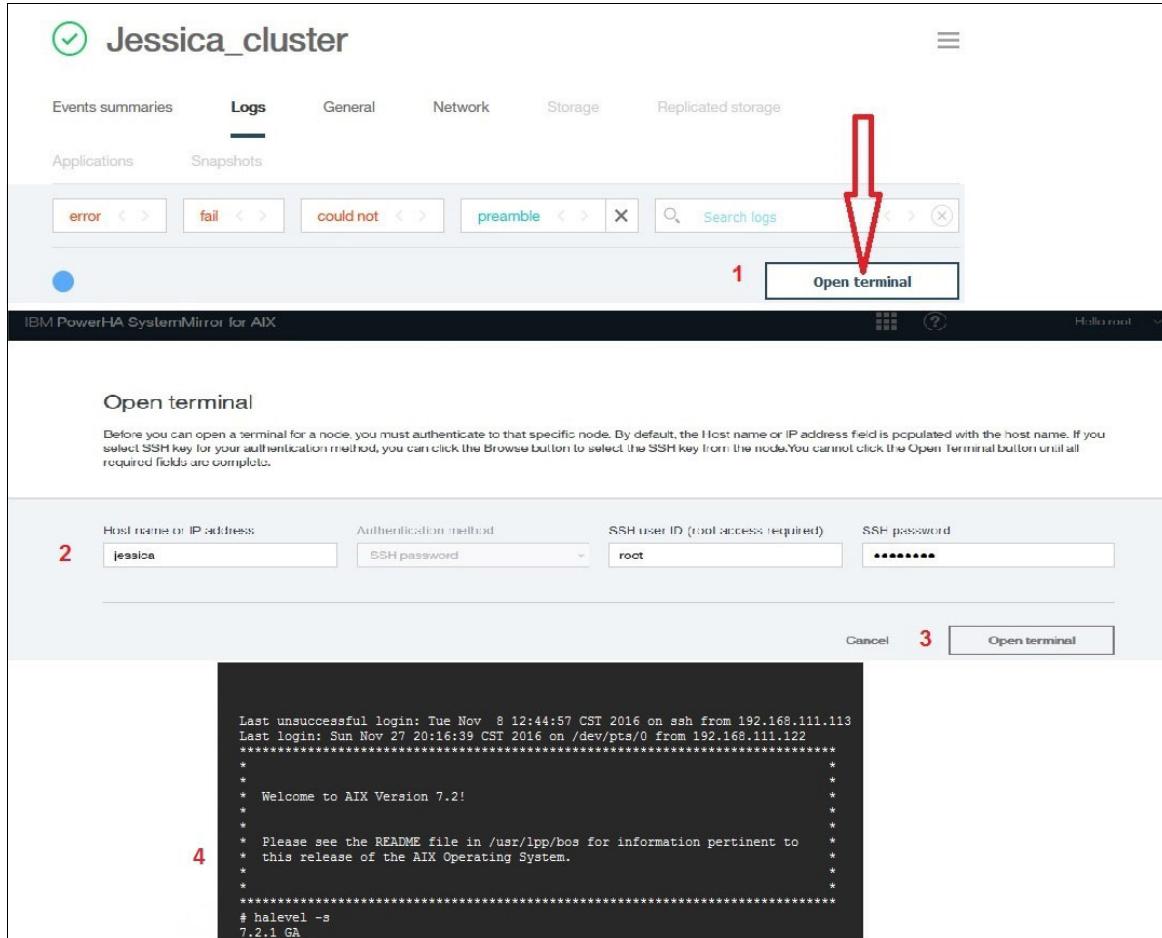


Figure 9-13 Terminal session

9.4 Troubleshooting

This section provides details about how to troubleshoot issues with your cluster.

9.4.1 Log files

As is the case with most PowerHA components, there is a set of log files specifically for the SMUI. These log files are as follows:

- | | |
|-----------------|--|
| smui-server.log | This log file is in the /usr/es/sbin/cluster/ui/server/logs/ directory. The smui-server.log file contains information about the PowerHA SystemMirror GUI server. |
| smui-agent.log | This log file is in the /usr/es/sbin/cluster/ui/agent/logs/ directory. The smui-agent.log file contains information about the agent that is installed on each PowerHA SystemMirror node. |

notify-event.log	This log file is in the /usr/es/sbin/cluster/ui/agent/logs/ directory. The notify-event.log file contains information about all PowerHA SystemMirror events that are sent from the agent to the PowerHA SystemMirror server.
------------------	---

9.4.2 Login problems

If you are experiencing problems logging in to the PowerHA SystemMirror GUI, complete the following steps:

1. Check for issues in the /usr/es/sbin/cluster/ui/server/logs/smui-server.log file.
2. Verify that the **smuiauth** command is installed correctly. Also, verify that the **smuiauth** command has the correct permissions by running the **ls -l** command from the /usr/es/sbin/cluster/ui/server/node_modules/smui-server/lib/auth/smuiauth directory. An output similar to the following example is displayed:

```
-r-x----- 1 root      system      21183 Oct 18 22:56
```
3. Verify that you can run the **smuiauth** command by running the **smuiauth -h** command.
4. Verify that the pluggable authentication module (PAM) framework is configured correctly by locating the following lines in the /etc/pam.conf file:

```
smuiauth auth    required pam_aix
smuiauth account required pam_aix
```

The PAM configuration occurs when you install the cluster.es.smui.server fileset.

9.4.3 Adding clusters

If you are not able to add clusters to the PowerHA SystemMirror GUI, complete the following steps:

1. Check for issues in the /usr/es/sbin/cluster/ui/server/logs/smui-server.log file:
 - a. If SFTP-related signatures exist in the log file, such as Received exit code 127 during the time of establishing an SFTP session, a problem exists with the SSH communication between the PowerHA SystemMirror GUI server and the cluster you are trying to add.
 - b. From the command line, verify that you can connect to the target system by using SSH File Transfer Protocol (SFTP). If you cannot connect, verify that the daemon is running on the PowerHA SystemMirror GUI server and the target node by running the **ps -ef | grep -w sshd |grep -v grep** command. You can also check the SFTP subsystem configuration in the /etc/ssh/sshd_config file and verify that following path is correct:

```
Subsystem sftp /usr/sbin/sftp-server
```

If the path is not correct, you must enter the correct path in the /etc/ssh/sshd_config file, and then restart the sshd subsystem.
2. Check for issues in the /usr/es/sbin/cluster/ui/agent/logs/agent_deploy.log file on the target cluster.
3. Check for issues in the /usr/es/sbin/cluster/ui/agent/logs/agent_distribution.log file on the target cluster.

9.4.4 Status not updating

If the SMUI is not updating the cluster status or displaying new events, complete the following steps:

1. Check for issues in the `/usr/es/sbin/cluster/ui/server/logs/smui-server.log` file.
2. Check for issues in the `/usr/es/sbin/cluster/ui/agent/logs/smui-agent.log` file. If certificate-related problems exist in the log file, the certificate on the target cluster and the certificate on the server do not match. An example of a certificate error follows:

```
WebSocket server - Agent authentication failed,  
remoteAddress:::ffff:10.40.20.186, Reason:SELF_SIGNED_CERT_IN_CHAIN
```



Cluster partitioning management update

From Version 7.1 forward, PowerHA SystemMirror provides more split and merge policies. Split and merge policies are important features in PowerHA SystemMirror because they are used to protect customers' data consistency and maintain application running stably in cluster split scenarios and other unstable situations. They are vital for customer environments.

This chapter describes split and merge policies.

This chapter covers the following topics:

- ▶ Introduction to cluster partitioning
- ▶ PowerHA cluster split and merge policies (before PowerHA V7.2.1)
- ▶ PowerHA quarantine policy
- ▶ Changes in split and merge policies in PowerHA V7.2.1
- ▶ Considerations for using split and merge quarantine policies
- ▶ Split and merge policy testing environment
- ▶ Scenario: Default split and merge policy
- ▶ Scenario: Split and merge policy with a disk tie breaker
- ▶ Scenario: Split and merge policy with the NFS tie breaker
- ▶ Scenario: Split and merge policy is manual
- ▶ Scenario: Active node halt policy quarantine
- ▶ Scenario: Enabling the disk fencing quarantine policy

10.1 Introduction to cluster partitioning

During normal operation, cluster nodes regularly exchange messages, commonly called heartbeats, to determine the health of each other. Figure 10-1 depicts a healthy two-node PowerHA cluster.

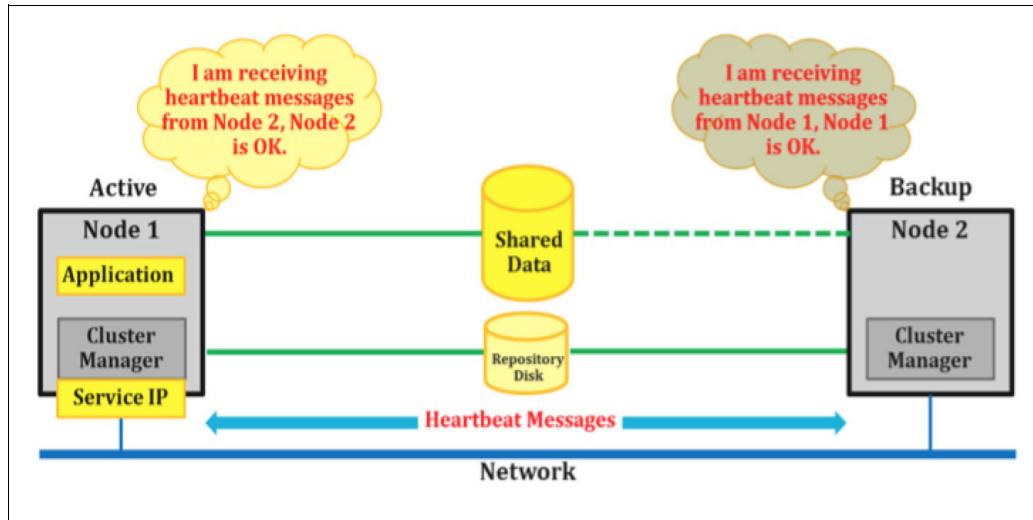


Figure 10-1 A healthy two-node PowerHA Cluster with heartbeat messages exchanged

When both the active and backup nodes fail to receive heartbeat messages, each node falsely declares the other node to be down, as shown in Figure 10-2. When this happens, the backup node attempts to takeover the shared resources, including shared data volumes. As a result, both nodes might be writing to the shared data and caused data corruption.

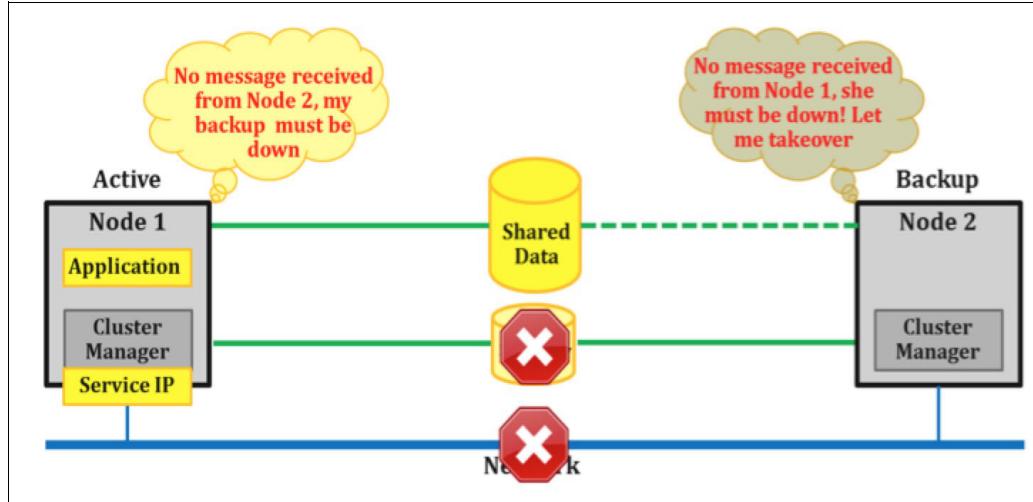


Figure 10-2 Cluster that is partitioned when nodes failed to communicate through heartbeat message exchange

When a set of nodes fails to communicate with the remaining set of nodes in a cluster, the cluster is said to be *partitioned*. This is also known as *node isolation*, or more commonly, *split brain*.

Note: As two-node clusters are by far the most common PowerHA cluster configuration, we introduce cluster partitioning concepts in the following sections in the context of a two-node cluster. These basic concepts can be applied similarly to clusters with more than two nodes and are further elaborated where necessary.

10.1.1 Causes of a partitioned cluster

Loss of all heartbeats can be caused by one of the following situations:

- ▶ When all communication paths between the nodes fail (as shown in Figure 10-2 on page 320).

Here is an example scenario based on a real-world experience:

- a. A cluster had two communication paths for heartbeat, the network and repository disk. The PowerHA network heartbeat mode was configured as multicast.
 - b. One day, a network configuration change was made that disabled the multicast network communication. As a result, network heartbeating no longer worked. But, system administrators were unaware of this problem because they did not monitor the PowerHA network status. The network heartbeat failure was left uncorrected.
 - c. The cluster continued to operate with heartbeat through the repository disk.
 - d. Some days later, the repository disk failed and the cluster was partitioned.
- ▶ One of the nodes is sick but not dead.
One node cannot send/receive heartbeat messages for a period, but resumes sending/receiving heartbeat messages afterward.
 - ▶ Another possible scenario is:
 - a. There is a cluster with nodes in separate physical hosts with dual Virtual I/O Servers (VIOSSs).
 - b. Due to some software or firmware defect, one node cannot perform I/O through the VIOSSs for a period but resumes I/O afterward. This causes an intermittent loss of heartbeats through all communication paths between the nodes.
 - c. When the duration of *I/O freeze* exceeds the node failure detection time, the nodes declare each other as down and the cluster is partitioned.

Although increasing the number of communication paths for heartbeating can minimize the occurrence of cluster partitioning due to communication path failure, the possibility cannot be eliminated completely.

10.1.2 Terminology

Here is the terminology that is used throughout this chapter:

Cluster split	When the nodes in a cluster fail to communicate with each other for a period, each node declares the other node as down. The cluster is split into partitions. A cluster split is said to have occurred.
Split policy	A PowerHA split policy defines the behavior of a cluster when a cluster split occurs.
Cluster merge	A PowerHA cluster merge policy defines the behavior of a cluster when a cluster merge occurs.

Merge policy	A PowerHA merge policy defines the behavior of a cluster when a cluster merge occurs.
Quarantine policy	A PowerHA quarantine policy defines how a standby node isolates or <i>quarantines</i> an active node or partition from the shared data to prevent data corruption when a cluster split occurs.
Critical resource group	When multiple resource groups (RGs) are configured in a cluster, the RG that is considered as most important or critical to the user is defined as the Critical Resource Group for a quarantine policy. For more information, see 10.3.1, “Active node halt quarantine policy” on page 332.
Standard cluster	A standard cluster is a traditional PowerHA cluster.
Stretched cluster	A stretched cluster is a PowerHA V7 cluster with nodes that are in sites within the same geographic location. All cluster nodes are connected to the same active and backup repository disks in a common storage area network (SAN).
Linked cluster	A linked cluster is a PowerHA V7 cluster with nodes that are in sites in different geographic locations. Nodes in each site have their own active and backup repository disks. The active repository disks in the two sites are kept in sync by Cluster Aware AIX (CAA).

10.2 PowerHA cluster split and merge policies (before PowerHA V7.2.1)

This section provides an introduction to PowerHA split and merge policies before PowerHA for AIX V7.2.1.

For more information, see the following IBM Redbooks:

- ▶ *IBM PowerHA SystemMirror for AIX Cookbook*, SG24-7739
- ▶ *IBM PowerHA SystemMirror V7.2 for IBM AIX Updates*, SG24-8278

10.2.1 Split policy

Before PowerHA V7.1, when a cluster split occurs, the backup node tries to take over the resources of the primary node, which results in a split-brain situation.

The PowerHA split policy was first introduced in PowerHA V7.1.3 with two options:

- ▶ None

This is the default option where the primary and backup nodes operate independently of each other after a split occurs, resulting in the same behavior as earlier versions during a split-brain situation.

- ▶ Tie breaker

This option is applicable to only clusters with sites configured. When a split occurs, the partition that fails to acquire the SCSI reservation on the tie-breaker disk has its nodes restarted. For a two-node cluster, one node is restarted, as shown in Figure 10-3 on page 323.

Note: EMC PowerPath disks are not supported as tie-breaker disks.

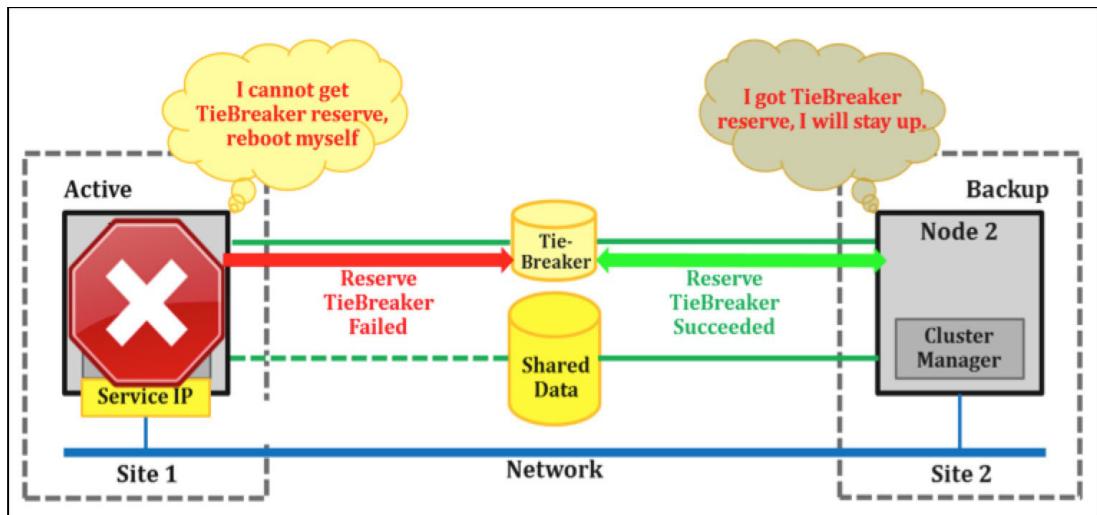


Figure 10-3 Disk tie-breaker split policy

PowerHA V7.2 added the following options to the split policy:

- ▶ Manual option

Initially, this option was applicable only to linked clusters. However, in PowerHA V7.2.1, it is now available for all cluster types. When a split occurs, each node waits for input from the user at the console to choose whether to continue running cluster services or restart the node.

- ▶ NFS support for the tie-breaker option

When a split occurs, the partition that fails to acquire a lock on the tie-breaker NFS file has its nodes restarted. For a two-node cluster, one node is restarted, as shown in Figure 10-4.

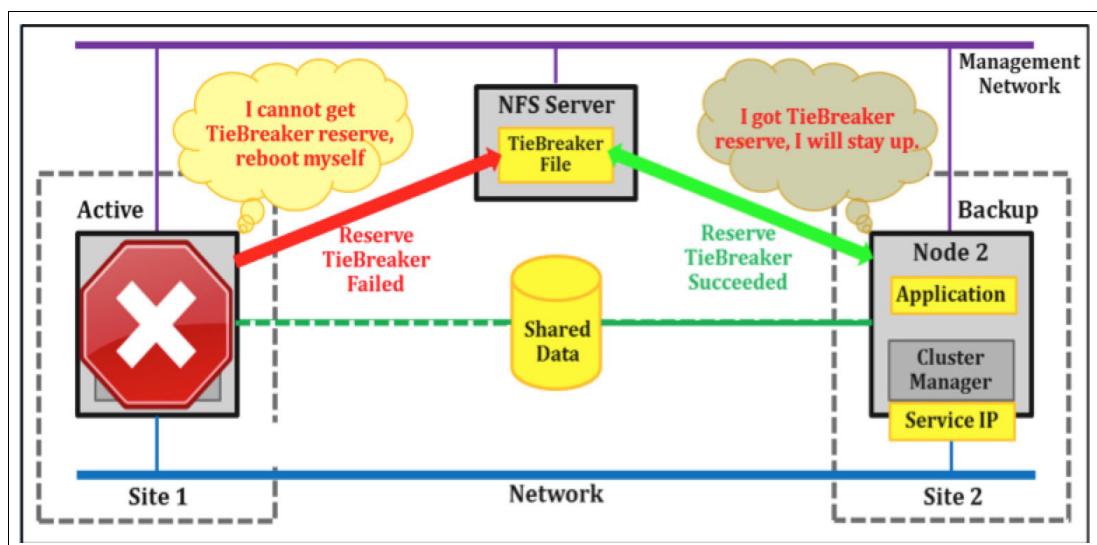


Figure 10-4 NFS tie-breaker split policy

Note: PowerHA V7.2.1 running and migrated to AIX 7.2.1 supports subcluster split and merge functions among all types of PowerHA clusters.

10.2.2 Merge policy

Before PowerHA V7.1, the default action when a merge occurs is to halt one of the nodes based on a predefined algorithm, such as halting the node with the highest node ID. There is no guarantee that the active node is not the one that is halted. The intention is to minimize the possibility of data corruption after a split-brain situation occurs.

The PowerHA merge policy was first introduced in PowerHA V7.1.3 with two options:

- ▶ Majority

This is the default option. The partition with the highest number of nodes remains online. If each partition has the same number of nodes, then the partition that has the lowest node ID is chosen. The partition that does not remain online is restarted, as specified by the chosen action plan. This behavior is similar to previous versions, as shown in Figure 10-5.

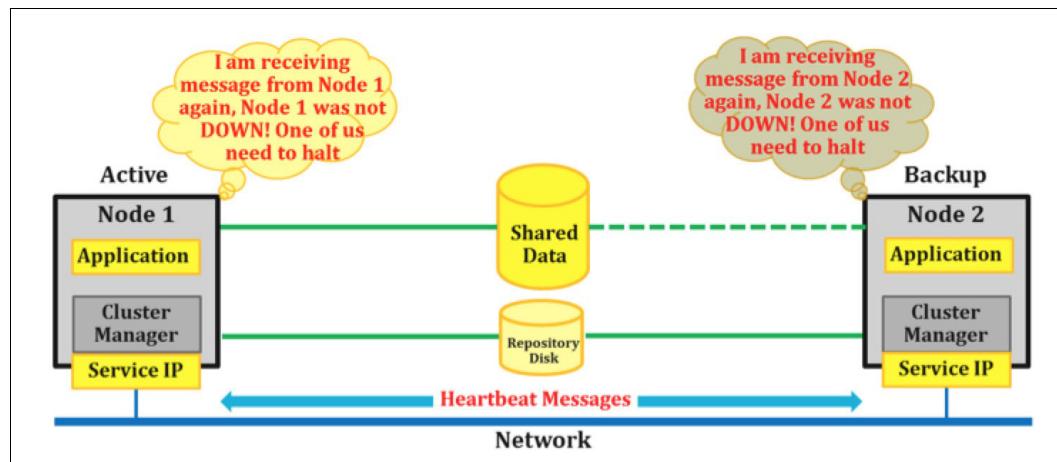


Figure 10-5 Default merge policy: Halt one of the nodes

- ▶ Tie breaker

Each partition attempts to acquire a SCSI reserve on the tie-breaker disk. The partition that cannot reserve the disk is restarted, or has cluster services that are restarted, as specified by the chosen action plan. If this option is selected, the split-policy configuration must also use the tie-breaker option.

PowerHA V7.2 added the following options to the merge policy:

- ▶ Manual option

This option is applicable only to linked clusters. When a split occurs, each node waits for input from the user at the console to choose whether to continue running cluster services or restart the node.

- ▶ Priority option

This policy indicates that the highest priority site continues to operate when a cluster merge event occurs. The sites are assigned with a priority based on the order they are listed in the site list. The first site in the site list is the highest priority site. This policy is only available for linked clusters.

3. NFS support for the tie-breaker option

When a split occurs, the partition that fails to acquire a lock on the tie-breaker NFS file has its nodes restarted. If this option is selected, the split-policy configuration must also use the tie-breaker option.

10.2.3 Configuration for the split and merge policy

Complete the following steps:

1. In the SMIT interface, select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split Management Policy**, as shown in Figure 10-6.

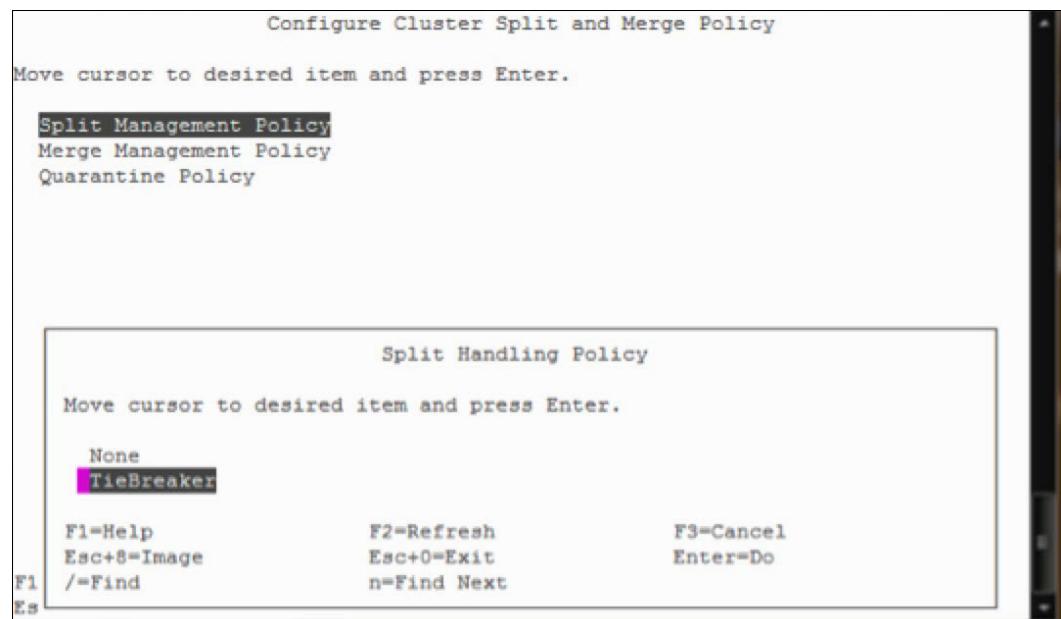


Figure 10-6 Configuring the cluster split and merge policy

2. Select TieBreaker. The manual option is not available if the cluster you are configuring is not a linked cluster. Select either Disk or NFS as the tie breaker, as shown in Figure 10-7.

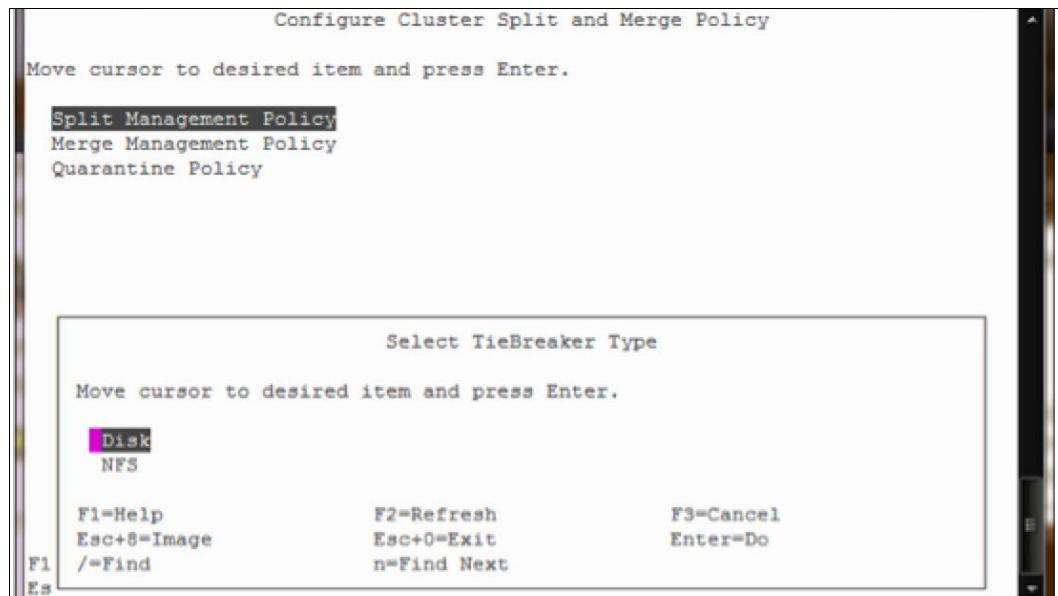


Figure 10-7 Selecting the tie-breaker disk

Disk tie-breaker split and merge policy

Select the disk to be used as the tie-breaker disk and synchronize the cluster. Figure 10-8 shows select hdisk3 as the tie breaker device.

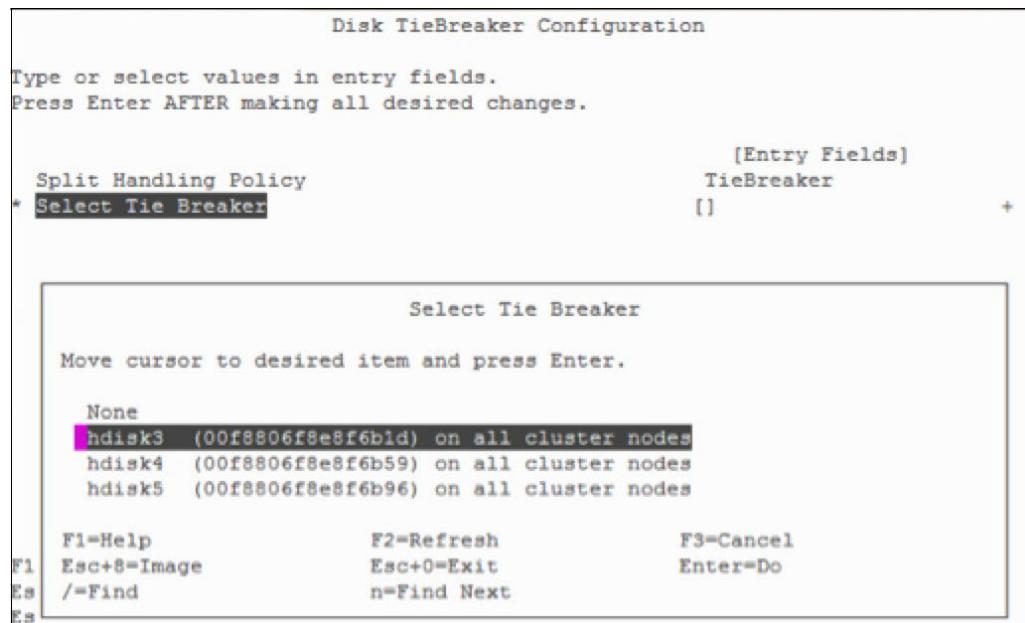


Figure 10-8 Tie-breaker disk split policy

Figure 10-9 shows the result after confirming the configuration.

```
COMMAND STATUS

Command: OK          stdout: yes          stderr: no

Before command completion, additional instructions may appear below.

hdisk3 changed
The PowerHA SystemMirror split and merge policies have been updated.
Current policies are:
    Split Handling Policy : Tie Breaker
    Merge Handling Policy : Tie Breaker
    Tie Breaker : hdisk3
    Split and Merge Action Plan : Reboot
The configuration must be synchronized to make this change known across the cluster.

F1=Help      F2=Refresh      F3=Cancel      Esc+6=Command
Esc+8=Image   Esc+9=Shell     Esc+0=Exit     /=Find
n=Find Next
```

Figure 10-9 Tie-breaker disk successfully added

Before configuring a disk as tie breaker, you can check its current reservation policy by using the AIX command **devrsrv**, as shown in Example 10-1.

Example 10-1 The devrsrv command shows no reserve

```
root@testnode1[/]# devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name      : hdisk3
Device Open On Current Host? : NO
ODM Reservation Policy : NO RESERVE
Device Reservation State : NO RESERVE
```

When cluster services are started, the first time after a disk tie breaker is configured on a node, the reservation policy of the tie-breaker disk is properly set to PR_exclusive with a persistent reserve key, as shown in Example 10-2.

Example 10-2 The devrsrv command shows PR_exclusive

```
root@testnode1[/]# devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name      : hdisk3
Device Open On Current Host? : NO
ODM Reservation Policy : PR EXCLUSIVE
ODM PR Key Value : 8477804151029074886
Device Reservation State : NO RESERVE
Registered PR Keys : No Keys Registered
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C
PR Capabilities Byte[3] : 0xa1 PTPL_A
PR Types Supported : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR
```

For a detailed description of how SCSI-3 PR (Persistent Reserve) of a tie-breaker disk works, refer to “SCSI reservation” in Appendix A of the *IBM PowerHA SystemMirror V7.2 for IBM AIX Updates*, SG24-8278.

When the Tie Breaker option of the split policy is selected, the merge policy is automatically set with the same tie-breaker option.

NFS tie-breaker split and merge policy

This section describes the tie-breaker split and merge policy tasks.

NFS server that is used for a tie breaker

The NFS server that is used for tie breaker is connected to a physical network other than the *service* networks that are configured in PowerHA. A logical choice is the management network that usually exists in all data center environment.

To configure the NFS server, complete the following steps:

1. Add /etc/host entries for the cluster nodes, for example:

```
172.16.25.31 testnode1  
172.16.15.32 testnode2
```

2. Configure the NFS domain by running the following command:

```
chnfsdom powerha
```

3. Start nfsrgyd by running the following command:

```
startsrc -s nfsrgyd
```

4. Add an NFS file system for storing the tie-breaker files, as shown in Figure 10-10.

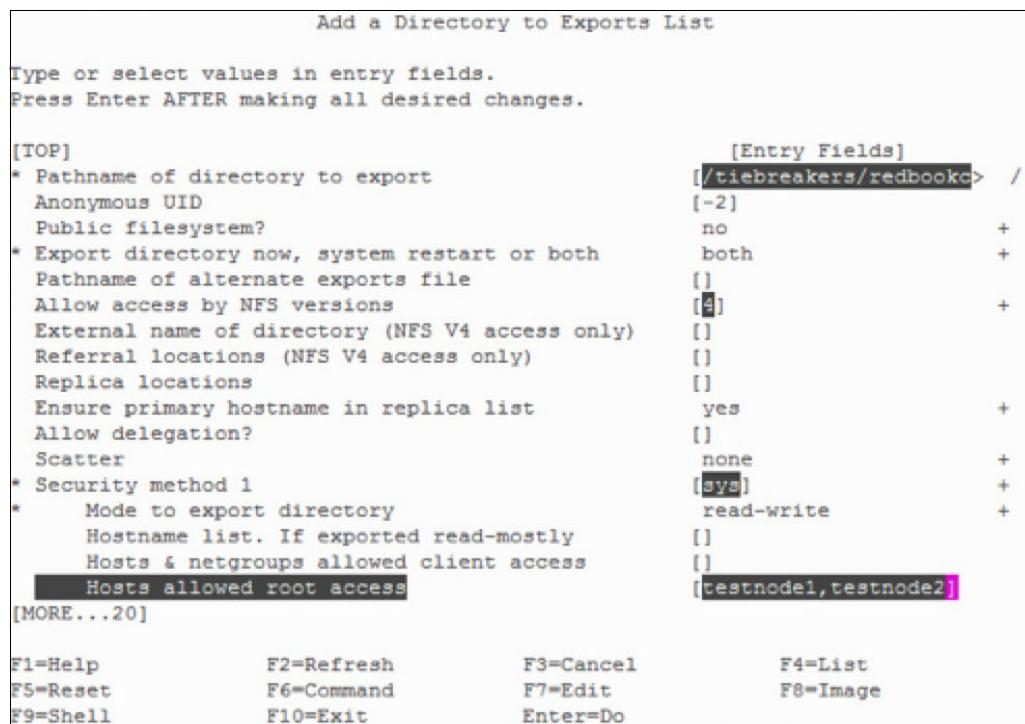


Figure 10-10 Adding a directory for NFS export

Here the NFS server is used as tie breaker for two clusters, redbookcluster and RBcluster, as shown in Example 10-3 on page 329.

Example 10-3 Directories exported

Example:

```
[root@atsnim:/]#exportfs
/software           -vers=3,public,sec=sys:krb5p:krb5i:krb5:dh,rw
/pha               -vers=3:4,sec=sys:krb5p:krb5i:krb5:dh,rw
/docs              -vers=3,public,sec=sys:krb5p:krb5i:krb5:dh,rw
/sybase             -sec=sys:krb5p:krb5i:krb5:dh,rw,root=172.16.0.0
/leilintemp         -sec=sys:none,rw
/powerhatest       -sec=sys:krb5p:krb5i:krb5:dh,rw,root=testnode1
/tiebreakers/redbookcluster -vers=4,sec=sys,rw,root=testnode1:testnode2
/tiebreakers/RBcluster   -vers=4,sec=sys,rw,root=testnode3:testnode4
```

On each PowerHA node

Complete the following tasks:

1. Add an entry for the NFS server to /etc/hosts:

```
10.1.1.3 tiebreaker
```

2. Configure the NFS domain by running the following command:

```
chnfsdom powerha
```

3. Start nfsrgyd by running the following command:

```
starts src -s nfsrgyd
```

4. Add the NFS tie-breaker directory to be mounted, as shown in Figure 10-11.

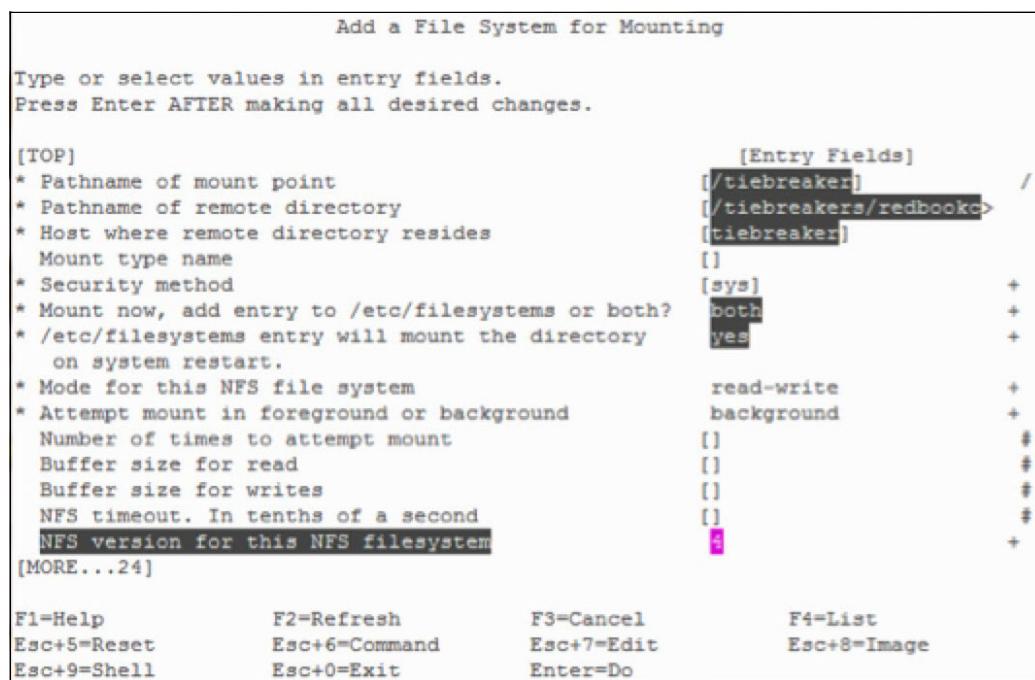


Figure 10-11 NFS directory to mount

Configuring PowerHA on one of the PowerHA nodes

Complete the following steps:

1. Configure the PowerHA tie-breaker split/merge policy, as shown in Figure 10-12.

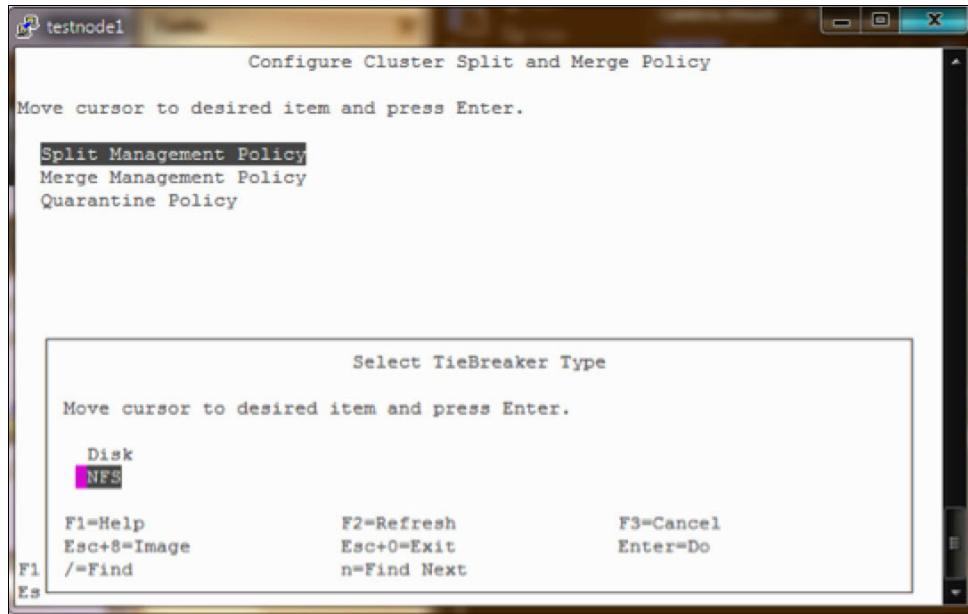


Figure 10-12 NFS tie-breaker split policy

- a. Input the host name of NFS server exporting the tie-breaker directory for the NFS tie breaker, for example, tiebreakers.
- b. Add the IP entry for the host name of the NFS server to /etc/hosts:
 - Full path name of local mount point for mounting the NFS tiebreaker directory. For example, /tiebreaker.
 - Full path name of the directory that is exported from the NFS server. In this case /tiebreaker.

Figure 10-13 on page 331 shows an example of the NFS tie-breaker configuration.

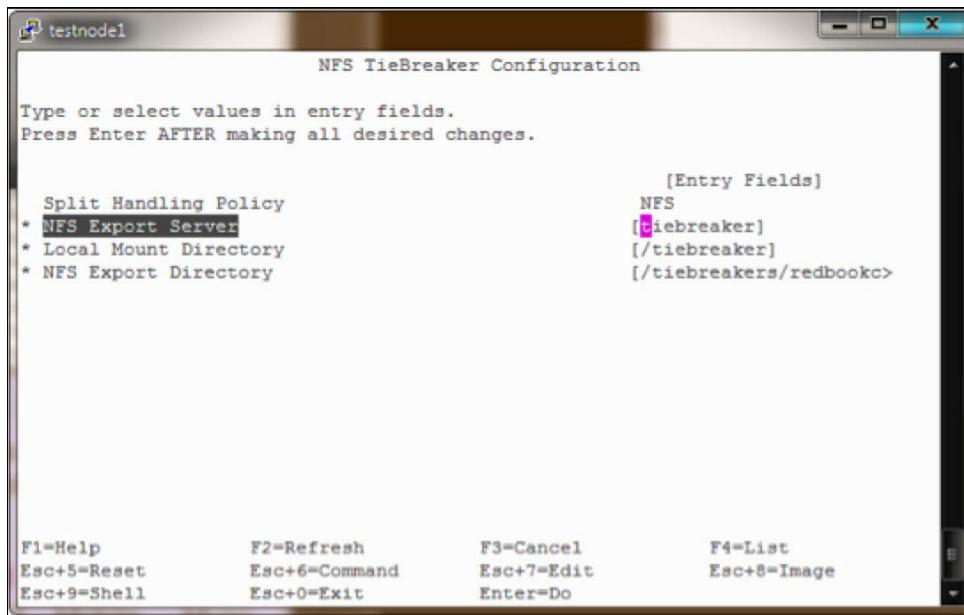


Figure 10-13 NFS tie-breaker configuration

2. Sync cluster

When cluster services are started on each node, tie-breaker files are created on the NFS server, as shown in Example 10-4.

Example 10-4 NFS tie-breaker files created

```
[root@tiebreaker:]#ls -Rl /tiebreakers
total 0
drwxr-xr-x    3 root      system          256 Nov 23 20:50 RBcluster
drwxr-xr-x    2 root      system          256 May 27 2016  lost+found
drwxr-xr-x    3 root      system          256 Nov 23 20:51 redbookcluster

/tiebreakers/RBcluster:
total 0
-rwx-----    1 root      system          0 Nov 23 20:50 PowerHA_NFS_Reserve
drwxr-xr-x    2 root      system          256 Nov 23 20:50
PowerHA_NFS_ReserverviewFilesDir

/tiebreakers/RBcluster/PowerHA_NFS_ReserverviewFilesDir:
total 16
-rwx-----    1 root      system          257 Nov 23 20:50 testnode3view
-rwx-----    1 root      system          257 Nov 23 20:50 testnode4view

/tiebreakers/redbookcluster:
total 0
-rwx-----    1 root      system          0 Nov 23 20:51 PowerHA_NFS_Reserve
drwxr-xr-x    2 root      system          256 Nov 23 20:51
PowerHA_NFS_ReserverviewFilesDir

/tiebreakers/redbookcluster/PowerHA_NFS_ReserverviewFilesDir:
total 16
-rwx-----    1 root      system          257 Nov 23 20:51 testnode1view
-rwx-----    1 root      system          257 Nov 23 20:51 testnode2view
```

10.3 PowerHA quarantine policy

This section introduces the PowerHA quarantine policy. For more information about PowerHA quarantine policies, see *IBM PowerHA SystemMirror V7.2 for IBM AIX Updates*, SG24-8278.

Quarantine policies were first introduced in PowerHA V7.2. A quarantine policy isolates the previously active node that was hosting a critical RG after a cluster split event or node failure occurs. The quarantine policy ensures that application data is not corrupted or lost.

There are two quarantine policies:

1. Active node halt
2. Disk fencing

10.3.1 Active node halt quarantine policy

When an RG is online on a cluster node, the node is said to be the *active* node for that RG. The *backup* or *standby* node for the RG is a cluster node where the RG comes online when the active node fails or when the RG is manually moved over.

With the *active node halt policy* (ANHP), in the event of a *cluster split*, the standby node for a critical RG attempts to halt the active node before taking over the RG and any other related RGs. This task is done by issuing command to the HMC, as shown in Figure 10-14.

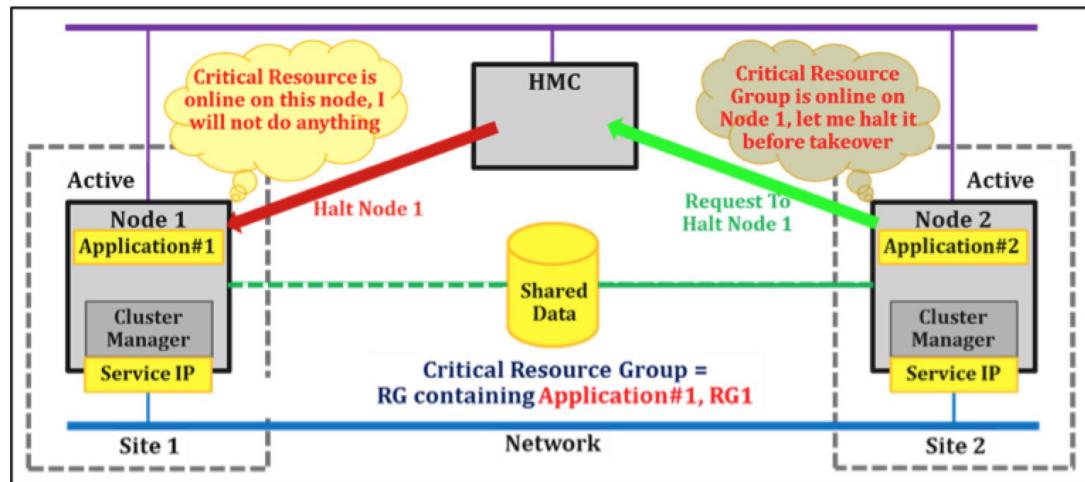


Figure 10-14 Active node halt process

If the backup node fails to halt the active node, for example, the communication failure with HMC, the RG is not taken over. This policy prevents application data corruption due to the same RGs being online on more than one node at the same time.

Now, let us elaborate why we need to define a critical RG.

In the simplest configuration of a two-node cluster with one RG, there is no ambiguity as to which node can be halted by the ANHP in the event of a cluster split. But, when there are multiple RGs in a cluster, it is not as simple:

- ▶ In a mutual takeover cluster configuration, different RGs are online on each cluster node and the nodes back up each other. An active node for one RG also is a backup or standby node for another RG. When a cluster split occurs, which node halts?
- ▶ When a cluster with multiple nodes and RGs is partitioned or split, some of the nodes in each partition might have RGs online, for example, there are multiple active nodes in each partition. Which partition can have its nodes halted?

It is unwanted to have nodes halting one another, resulting in the cluster down as a whole.

PowerHA V7.2 introduces the *Critical Resource Groups* for a user to define which RG is the most important one when multiple RGs are configured. The ANHP can then use the critical RG to determine which node is halted or restarted. The node or the partition with the critical RG online is halted/ restarted and *quarantined*, as shown in Figure 10-14 on page 332.

10.3.2 Disk fencing quarantine

With this policy, the backup node fences off the active node from the shared disks before taking over the active node's resources, as shown in Figure 10-15. This action prevents application data corruption by preventing the RG coming online on more than one node at a time. As for the ANHP, the user also must define the Critical Resource Group for this policy.

Because this policy only fences off disks from the active node without halt or restarting it, it is configured together with a split and merge policy.

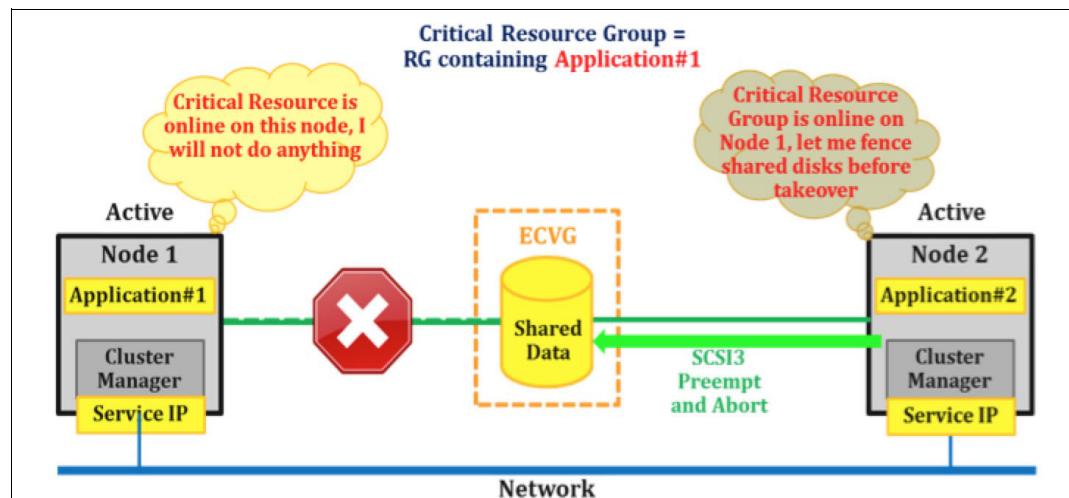


Figure 10-15 Disk fencing quarantine

10.3.3 Configuration of quarantine policies

In the SMIT interface, select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Quarantine Policy**, as shown in Figure 10-16.

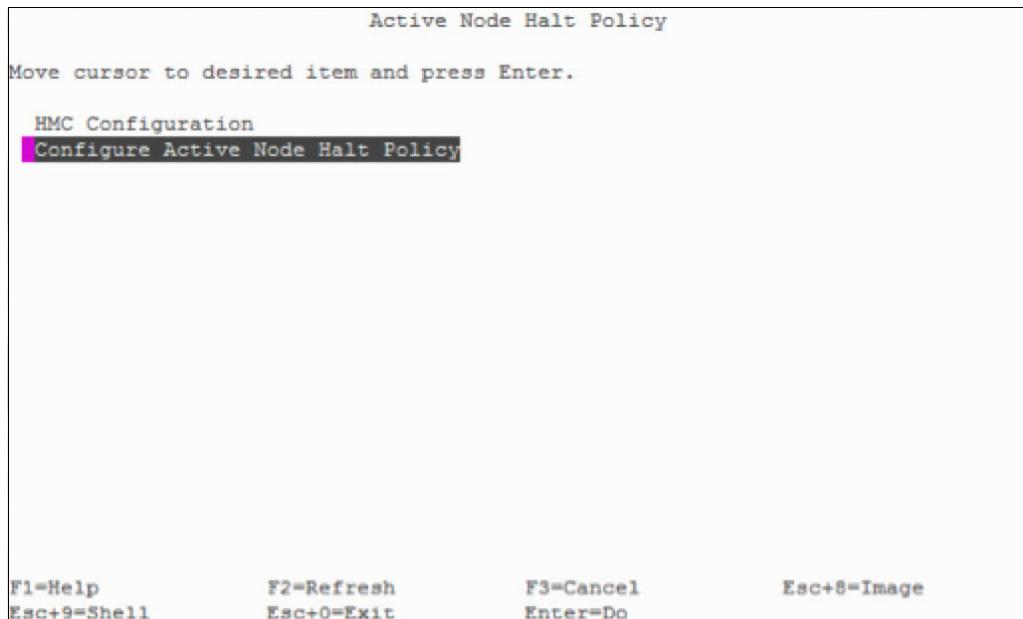


Figure 10-16 Active node halt policy

The active node halt

This task consists of the following steps:

1. Configure the HMC for the cluster nodes to run HMC commands remotely without the need to specify password.
2. Add the public keys (`id_rsa.pub`) of cluster nodes to the `authorized_keys2` in the `.ssh` directory on the HMC.

3. Configure the HMC to be used for halting nodes when the split occurs, as shown in Figure 10-17, Figure 10-18, and Figure 10-19 on page 336.

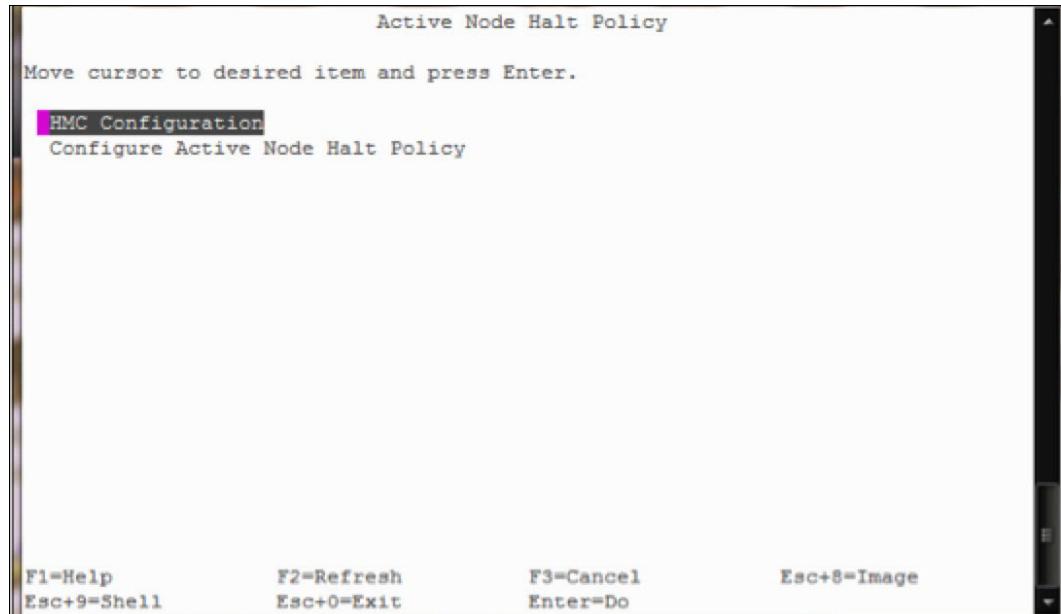


Figure 10-17 Active node halt policy HMC configuration

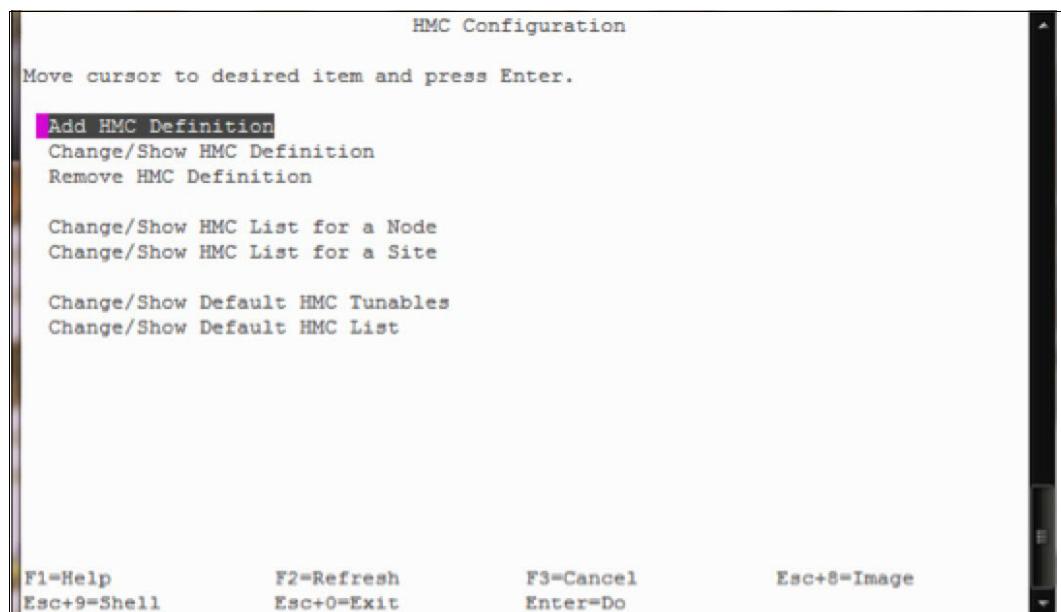


Figure 10-18 HMC definition for active node halt policy

Add HMC Definition

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

	[Entry Fields]	
* HMC name	[172.16.15.56]	+
DLPAR operations timeout (in minutes)	[]	#
Number of retries	[]	#
Delay between retries (in seconds)	[]	#
Nodes	[testnode2 testnode1]	+
Sites	[]	+
Check connectivity between HMC and nodes	Yes	+

F1=Help F2=Refresh F3=Cancel F4=List
Esc+5=Reset Esc+6=Command Esc+7>Edit Esc+8=Image
Esc+9=Shell Esc+0=Exit Enter=Do

Figure 10-19 Add HMC for active node halt policy

4. Configure the ANHP and specify the Critical Resource Group, as shown in Figure 10-20, Figure 10-21 on page 337, and Figure 10-22 on page 337.

Active Node Halt Policy

Move cursor to desired item and press Enter.

HMC Configuration			
Configure Active Node Halt Policy			

F1=Help F2=Refresh F3=Cancel Esc+8=Image
Esc+9=Shell Esc+0=Exit Enter=Do

Figure 10-20 Configure active node halt policy

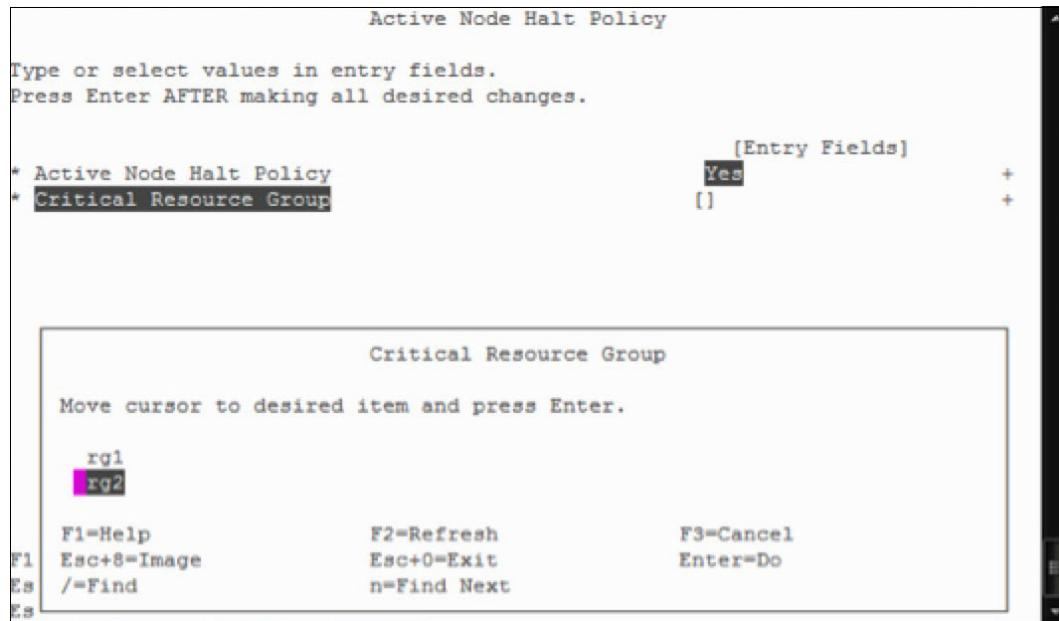


Figure 10-21 Critical resource group for active node halt policy

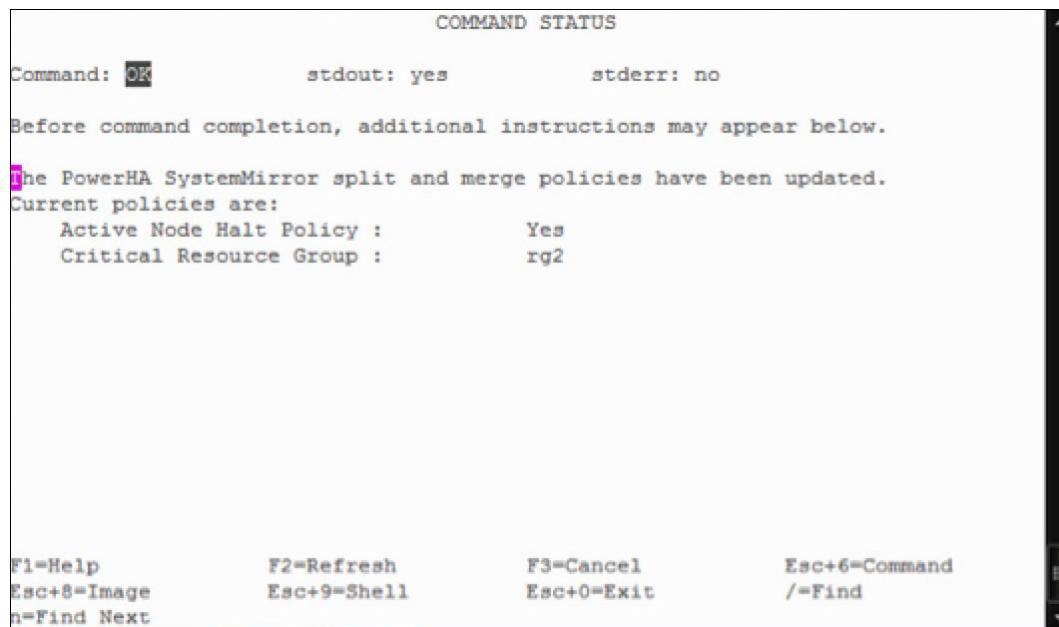


Figure 10-22 Critical resource group add success

Disk fencing

Similar to the ANHP, a critical RG must be selected to go along with it, as shown in Figure 10-23 and Figure 10-24.

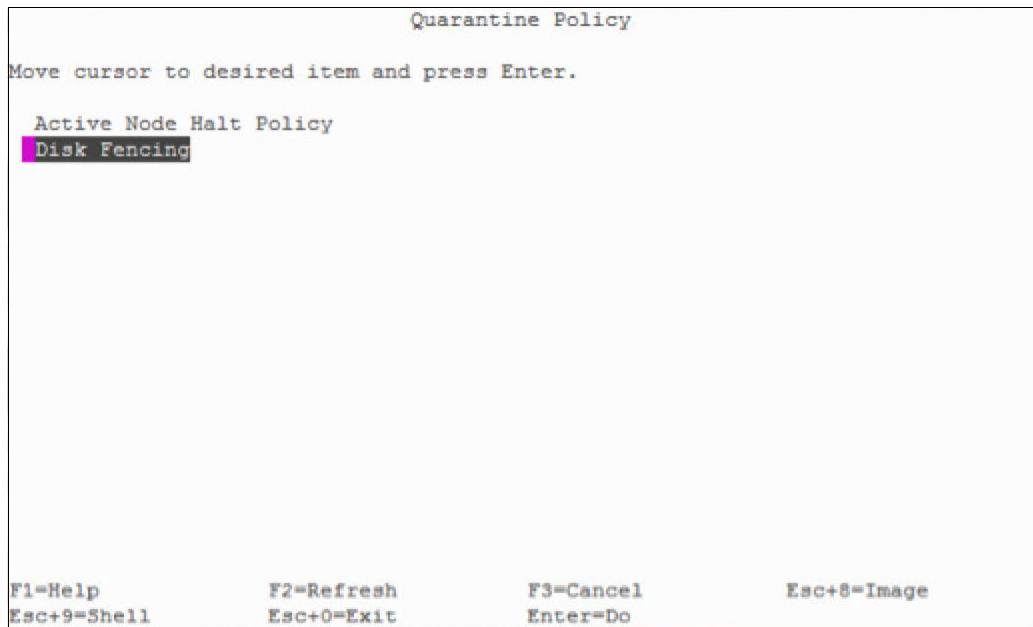


Figure 10-23 Disk fencing quarantine policy

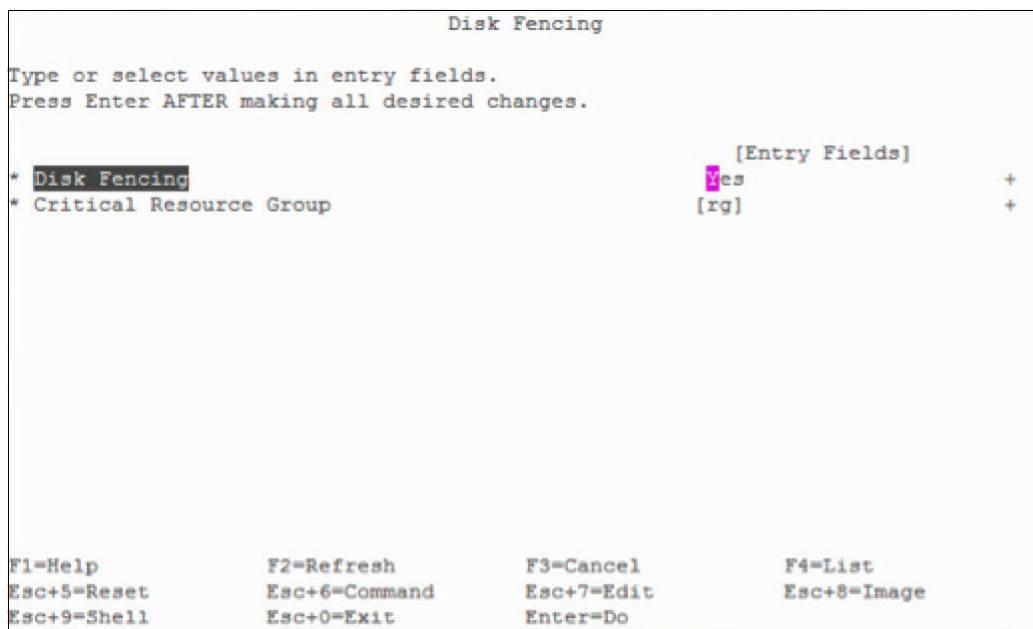


Figure 10-24 Disk fencing critical resource group

The current setting of the quarantine policy can be checked by using **clmgr**, as shown in Example 10-5.

Example 10-5 The clmgr command displaying the current quarantine policy

```
root@testnode1[/]#clmgr query cluster | grep -i quarantine  
QUARANTINE_POLICY="fencing"
```

Important: The disk fencing quarantine policy cannot be enabled or disabled if cluster services are active.

When cluster services are started on a node after enabling the Disk Fencing quarantine policy, the reservation policy and state of the shared volumes are set to PR Shared with the PR keys of both nodes registered. This action can be observed by using the **devrsrv** command, as shown in Example 10-6.

Example 10-6 Query reservation policy

```
root@testnode3[/]#clmgr query cluster | grep -i cluster_name  
CLUSTER_NAME="RBcluster"  
  
root@testnode3[/]#clmgr query nodes  
testnode4  
testnode3  
  
root@testnode3[/]#clmgr query resource_group  
rg  
root@testnode3[/]#clmgr query resource_group rg | grep -i volume  
VOLUME_GROUP="vg1"  
root@testnode3[/]#lspv  
hdisk0      00f8806f26239b8c          rootvg      active  
hdisk2      00f8806f909bc31a          caavg_private active  
hdisk3      00f8806f909bc357          vg1         concurrent  
hdisk4      00f8806f909bc396          vg1         concurrent  
  
root@testnode3[/]#clRGinfo  
-----  
Group Name           State       Node  
-----  
rg                  ONLINE     testnode3  
                      OFFLINE    testnode4  
  
root@testnode3[/]#devrsrv -c query -l hdisk3  
Device Reservation State Information  
=====  
Device Name          : hdisk3  
Device Open On Current Host? : YES  
ODM Reservation Policy : PR SHARED  
ODM PR Key Value   : 4503687425852313  
Device Reservation State : PR SHARED  
PR Generation Value : 15  
PR Type             : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)  
PR Holder Key Value : 0  
Registered PR Keys  : 4503687425852313 9007287053222809  
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C  
PR Capabilities Byte[3] : 0xa1 PTPL_A  
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR  
  
root@testnode3[/]#devrsrv -c query -l hdisk4
```

```

Device Reservation State Information
=====
Device Name          : hdisk4
Device Open On Current Host? : YES
ODM Reservation Policy : PR SHARED
ODM PR Key Value   : 4503687425852313
Device Reservation State : PR SHARED
PR Generation Value : 15
PR Type             : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value: 0
Registered PR Keys  : 4503687425852313 9007287053222809
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C
PR Capabilities Byte[3] : 0xa1 PTPL_A
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

root@testnode4[/]#lspv
hdisk0      00f8806f26239b8c           rootvg      active
hdisk2      00f8806f909bc31a           caavg_private active
hdisk3      00f8806f909bc357           vg1         concurrent
hdisk4      00f8806f909bc396           vg1         concurrent

root@testnode4[/]#devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name          : hdisk3
Device Open On Current Host? : YES
ODM Reservation Policy : PR SHARED
ODM PR Key Value   : 9007287053222809
Device Reservation State : PR SHARED
PR Generation Value : 15
PR Type             : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value: 0
Registered PR Keys  : 4503687425852313 9007287053222809
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C
PR Capabilities Byte[3] : 0xa1 PTPL_A
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

root@testnode4[/]#devrsrv -c query -l hdisk4
Device Reservation State Information
=====
Device Name          : hdisk4
Device Open On Current Host? : YES
ODM Reservation Policy : PR SHARED
ODM PR Key Value   : 9007287053222809
Device Reservation State : PR SHARED
PR Generation Value : 15
PR Type             : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value: 0
Registered PR Keys  : 4503687425852313 9007287053222809
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C
PR Capabilities Byte[3] : 0xa1 PTPL_A
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

```

The PR Shared reservation policy uses the SCSI-3 reservation of type *WRITE EXCLUSIVE, ALL REGISTRANTS*, as shown in Example 10-7 on page 341. Only nodes that are registered can write to the shared volumes. When a cluster split occurs, the standby node ejects the PR registration of the active node on all shared volumes of the affected RGs. In Example 10-6 on page 339, the only registrations that are left on hdisk3 and hdisk4 are of testnode4, effectively fencing off testnode3 from the shared volumes.

Note: Only a registered node can eject the registration of other nodes.

Example 10-7 WRITE EXCLUSIVE, ALL REGISTRANTS PR type

```
root@testnode4[/]#devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name          : hdisk3
Device Open On Current Host? : YES
ODM Reservation Policy : PR SHARED
ODM PR Key Value   : 9007287053222809
Device Reservation State : PR SHARED
PR Generation Value : 15
PR Type             : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value : 0
Registered PR Keys  : 9007287053222809
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C
PR Capabilities Byte[3] : 0xa1 PTPL_A
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE
PR_EA_AR

root@testnode4[/]#devrsrv -c query -l hdisk4
Device Reservation State Information
=====
Device Name          : hdisk4
Device Open On Current Host? : YES
ODM Reservation Policy : PR SHARED
ODM PR Key Value   : 9007287053222809
Device Reservation State : PR SHARED
PR Generation Value : 15
PR Type             : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value : 0
Registered PR Keys  : 9007287053222809
PR Capabilities Byte[2] : 0x15 CRH ATP_C PTPL_C
```

Node *testnode3* is again registered on hdisk3 and hdisk4 when it has successfully rejoins testnode4 to form a cluster. You must perform a restart of cluster services on testnode3.

10.4 Changes in split and merge policies in PowerHA V7.2.1

This section provides a list of changes that are associated with the split and merge policies that are introduced in PowerHA V7.2.1 for AIX 7.2.1:

- ▶ Split and merge policies are configurable for all cluster types when AIX is at Version 7.2.1, as summarized in Table 10-1.

Table 10-1 Split and merge policies for all cluster types

Cluster Type	Pre AIX 7.2.1		AIX 7.2.1
	Split policy	Merge policy	Split and merge policy
Standard	Not supported		None-Majority TB (Disk)-TB (Disk) TB (NFS)-TB (NFS) Manual-Manual
Stretched	None	Majority	
	TieBreaker	TieBreaker	
Linked	None	Majority	None-Majority TB (Disk)-TB (Disk) TB (NFS)-TB (NFS) Manual-Manual
	TieBreaker	TieBreaker	
	Manual	Manual	

- ▶ Split and merge policies are configured as a whole instead of separately. These options can also vary a bit based on the exact AIX dependency.
- ▶ The action plan for the split and merge policy is configurable.
- ▶ An entry is added to the Problem Determination Tools menu for starting cluster services on merged node after a cluster split.
- ▶ Changes were added to `clmgr` for configuring the split and merge policy.

10.4.1 Configuring the split and merge policy by using SMIT

The split and merge policies are now configured as a whole, as shown in Figure 10-25, instead of separately, as described in 10.2.3, “Configuration for the split and merge policy” on page 325.

Figure 10-25 Configuring the split handling policy

All three options, None, Tie Breaker, and Manual, are now available for all cluster types, which includes standard, stretched, and linked clusters.

Before PowerHA V7.2.1, the split policy has a default setting of None and the merge policy has default setting of Majority and the default action was Reboot (Figure 10-26). This behavior has not changed.

Figure 10-26 Split and merge action plan menu

For the Tie Breaker option, the action plan for split and merge is now configurable as follows (Figure 10-27):

- ## ► Reboot.

This is the default option before PowerHA V7.1.2. The nodes of the losing partition are restarted when a cluster split occurs.

- ▶ Disable applications auto-start and reboot.

On a split event, the nodes on the losing partition are restarted, and the RGs cannot be brought online automatically after restart.

- ▶ Disable Cluster Services Auto-Start and Reboot.

Upon a split event, the nodes on the losing partition are restarted. The cluster services, CAA/RSCT/PowerHA, are not started on restarted. After the split condition is healed, select Start CAA on Merged Node from SMIT to enable the cluster services and bring the cluster to a stable state.

Note: If you specify the Split-Merge policy as None-None, the action plan is not implemented and a restart does not occur after the cluster split and merge events. This option is only available in your environment if it is running IBM AIX 7.2 with Technology Level 1, or later.

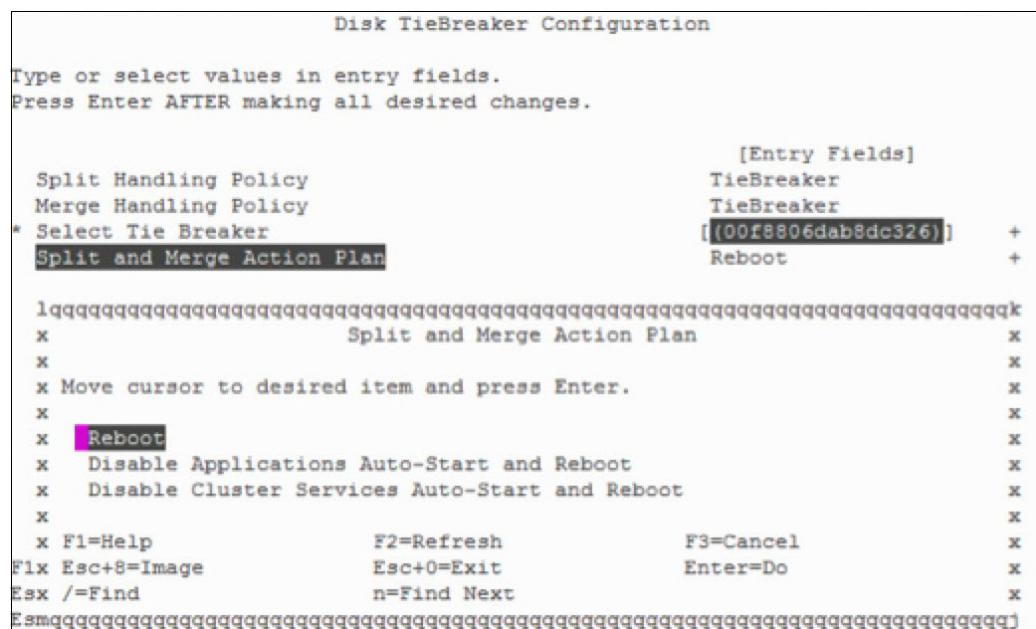


Figure 10-27 Disk tie breaker split and merge action plan

Similarly, Figure 10-28 shows the NFS TieBreaker policy SMIT window.

Figure 10-28 NFS tie breaker split and merge action plan

10.4.2 Configuring the split and merge policy by using clmgr

The **c1mgr** utility has the following changes for the split and merge policy configuration (Figure 10-29):

- ▶ Added a **none** option to the merge policy.
 - ▶ There is a local and remote quorum directory.
 - ▶ Added **disable_rgs_autostart** and **disable_cluster_services_autostart** options to the action plan.

```
clmgr modify cluster \
    [ SPLIT_POLICY={none|tiebreaker|manual|NFS} ] \
    [ TIEBREAKER=<disk> ] \
    [ MERGE_POLICY={none|majority|tiebreaker|manual|NFS} ] \
    [ NFS_QUORUM_SERVER=<server> ] \
    [ LOCAL_QUORUM_DIRECTORY=<local_mount>] \
    [ REMOTE_QUORUM_DIRECTORY=<remote_mount>] \
    [ QUARANTINE_POLICY=<disable|node_halt|fencing|halt_with_fencing>] \
    [ CRITICAL_RG=<rgname> ] \
    [ NOTIFY_METHOD=<method> ] \
    [ NOTIFY_INTERVAL=### ] \
    [ MAXIMUM_NOTIFICATIONS=### ] \
    [ DEFAULT_SURVIVING_SITE=<site> ] \
    [ APPLY_TO_PPRC_TAKEOVER={yes|no} ] \
[ ACTION_PLAN={reboot|disable_rgs_autostart|disable_cluster_services_autostart} ]
```

Figure 10-29 The clmgr split and merge options

The Split/Merge policy of *none/none* can be configured only by using **clmgr**, as shown in Example 10-8. There is no SMIT option to configure this option.

Example 10-8 The clmgr modify split/merge policy to none

```
# clmgr modify cluster SPLIT_POLICY=none MERGE_POLICY=none
The PowerHA SystemMirror split and merge policies have been updated.
Current policies are:
    Split Handling Policy :          None
    Merge Handling Policy :         None
    Split and Merge Action Plan : Reboot
```

The configuration must be synchronized to make this change known across the cluster.

10.4.3 Starting cluster services after a split

If the split-merge action plan of *disable cluster services auto start* is chosen in the configuration, then on a split event the losing partition nodes are restarted without bringing the cluster services online until these services are manually enabled.

It is required to enable the cluster services after a split situation is healed. Until the user resolves this enablement, the cluster services are not running on the losing partition nodes even after the networks rejoin. The losing partition nodes join the existing CAA cluster after the re-enable is performed. This is done by running **smitty sysmirror** and selecting **Problem Determination Tools** → **Start CAA on Merged Node**, as shown in Figure 10-30.



Figure 10-30 Starting Cluster Aware AIX on the merged node

10.4.4 Migration and limitation

Multiple split or merge situations cannot be handled at one time. For example, in the case of an asymmetric topology (AST), where some nodes have visibility to both islands, the nodes do not form a clean split. In such cases, a split event is not generated when AST halts a node to correct the asymmetry.

With the NFS tiebreaker split policy configured, if the tie breaker group leader (TBGL) node is restarted, then all other nodes in the winning partition are restarted. No preemption is supported in this case.

Tie-breaker disk preemption does not work in the case of a TBGL hard restart or power off.

The merge events are not available in a stretched cluster with versions earlier to AIX 7.2.1, as shown in Figure 10-31.

	Before Migration	After Migration to PowerHA721
1	None-Majority None-Priority None-Manual	None-Majority
2	Tie-breaker - Tie-breaker Tie-breaker - Priority	Tie-breaker - Tie-breaker
3	Manual - Manual	Manual - Manual

Figure 10-31 Split merge policies pre- and post-migration

10.5 Considerations for using split and merge quarantine policies

A split and merge policy is used for deciding which node or partition can be restarted when a cluster split occurs. A quarantine policy is used for fencing off, or *quarantining*, the active node from shared disks when a cluster split occurs. Both types of policies are designed to prevent data corruption in the event of cluster partitioning.

The quarantine policy does not require additional infrastructure resources, but the split and merge policy does. Users select the appropriate policy or combination of policies that suit their data center environments.

For example, instead of using the disk tie-breaker split and merge policy that requires one disk tie breaker per cluster, you want to use a single NFS server as a tie breaker for multiple clusters (Figure 10-32) to minimize resource requirements. This is a tradeoff between resource and effectiveness.

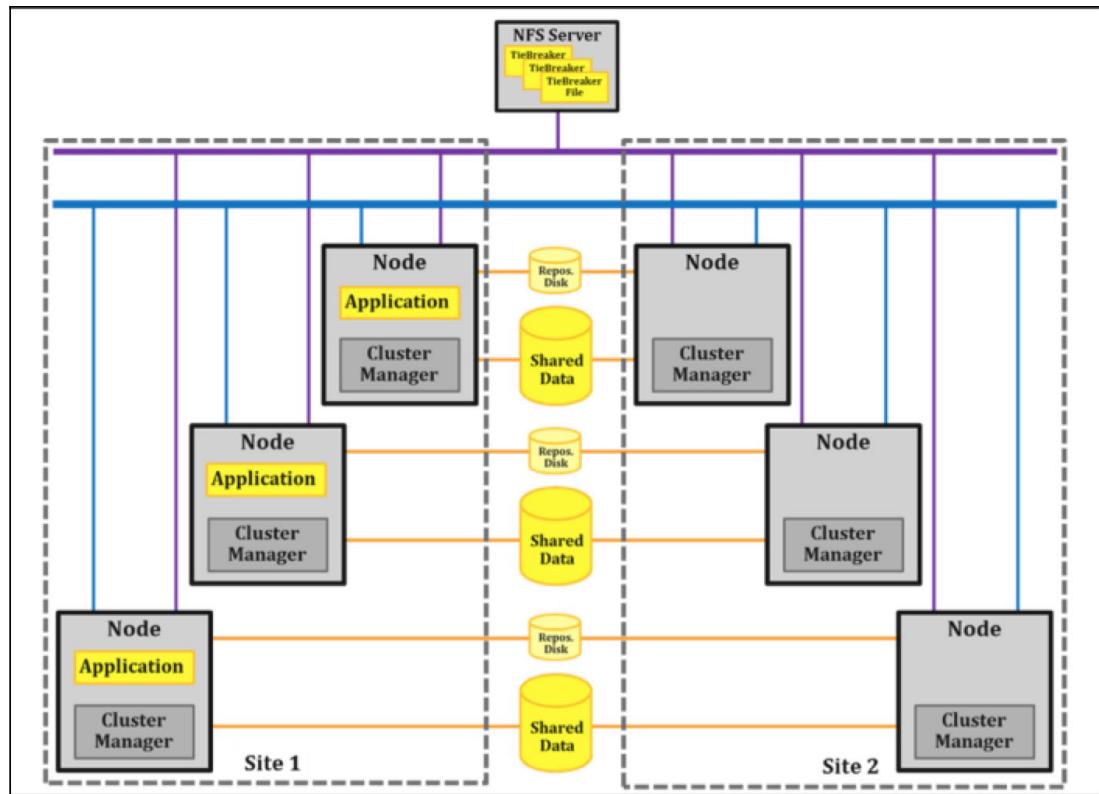


Figure 10-32 Using a single NFS server as a tie breaker for multiple clusters

For those who want to prevent only the possibility of data corruption with minimal configuration, and are satisfied with possible manual intervention that is required in the event of a cluster split, you can use the disk fencing quarantine policy. Again, this is a tradeoff. Figure 10-33 presents a comparison summary of these policies.

Policy Type	Policy	Cluster Type Applicable	Method to protect against data corruption	Additional Resource Required	Additional Configuration Required	Comment
Split	None	All	n/a	None	None	
Merge	Majority	All	Halt node with highest node id	None	None	Can only minimize but not eliminate the possibility of data corruption
Split/Merge	Disk Tie Breaker	Stretched, Linked	Halt one node. Use tie breaker to determine which node to reboot	Tie Breaker Disk	<ul style="list-style-type: none"> • Site configuration • Disk Tiebreaker configuration 	<ul style="list-style-type: none"> • Each cluster requires one tie breaker disk. • EMC PowerPath device not supported
Split/Merge	NFS Tie Breaker	Stretched, Linked	Halt one node. Use tie breaker to determine which node to reboot	NFSv4 Server	<ul style="list-style-type: none"> • Site configuration • NFS client/server configuration 	TieBreaker NFS server can serve multiple clusters
Split/Merge	Manual	Linked	Relying on human manual intervention	None	Site configuration	For human decision of which partition should remain online in the event of cluster partitioning. Mainly for DR solutions.
Quarantine	Active Node Halt	All	Backup node halt active node before taking over resources from active node	None	Passwordless ssh connection to HMC for cluster nodes, Critical Resource group	If fail to halt active node, resource will not be taken over and user will be alerted Configured with a split/merge policy is recommended
Quarantine	Disk Fencing	All	Fence off active node from shared disk	None	Critical Resource group	Configured with a split/merge policy is recommended

Figure 10-33 Comparison summary of split and merge policies

10.6 Split and merge policy testing environment

Figure 10-34 shows the topology of testing scenarios in this chapter.

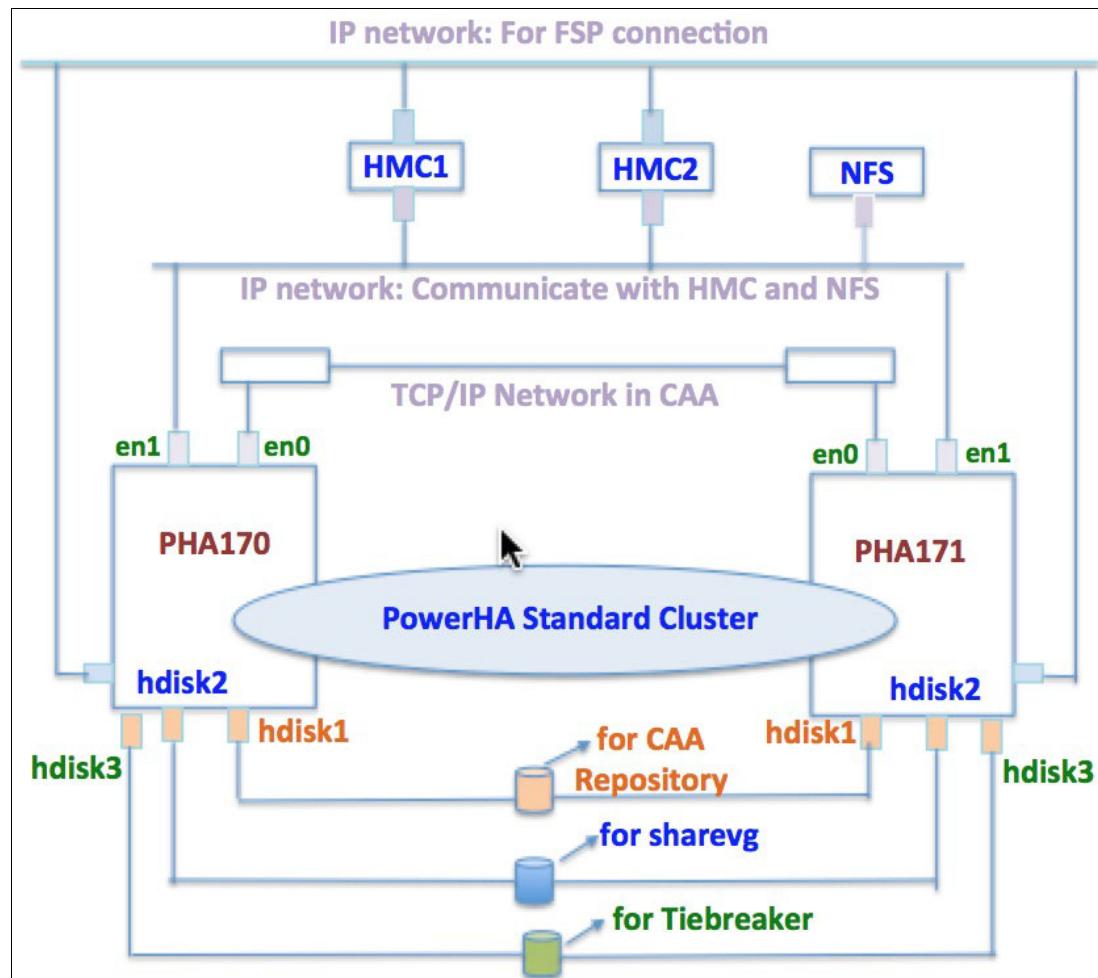


Figure 10-34 Testing scenario for the split and merge policy

Our testing environment is a single PowerHA standard cluster. It includes two AIX LPARs with nodes host names *PHA170* and *PHA171*. Each node has two network interfaces. One interface is used for communication with HMCs and NFS server, and the other is used in the PowerHA cluster. Each node has three FC adapters. The first adapter is used for rootvg, the second adapter is used for user shared data access, and the third one is used for tie-breaker access.

The PowerHA cluster is a basic configuration with the specific configuration option for different split and merge policies.

10.6.1 Basic configuration

Table 10-2 shows the PowerHA cluster's attributes. This is a basic two-node PowerHA standard cluster.

Table 10-2 PowerHA cluster's configuration

Component	PHA170	PHA171
Cluster name	PHA_cluster Cluster type: Standard Cluster or No Site Cluster (NSC)	
Network interface	en0: 172.16.51.170 netmask: 255.255.255.0 Gateway: 172.16.51.1 en1: 172.16.15.242	en0: 172.16.51.171 Netmask: 255.255.255.0 Gateway: 172.16.51.1 en1: 172.16.15.243
Network	net_ether_01 (172.16.51.0/24)	
CAA	Unicast Repository disk: hdisk1	
Shared VG	sharevg:hdisk2	
Service IP	172.16.51.172 PHASvc	
Resource Group	Resource Group testRG: ▶ Startup Policy: Online On Home Node Only ▶ Failover Policy: Failover To Next Priority Node In The List ▶ Fallback Policy: Never Fallback ▶ Participating Nodes: PHA170 PHA171 ▶ Service IP Label: PHASvc ▶ Volume Group: sharevg	

10.6.2 Specific hardware configuration for some scenarios

This section describes the specific hardware configurations for some scenarios.

Split and merge policy is tie breaker (disk)

In this scenario, add one shared disk, hdisk2, to act as the tie breaker.

Split and merge policy is tie breaker (NFS)

In this scenario, add one network file system (NFS) node to act as the tie breaker.

Quarantine policy is active node halt policy

In this scenario, add two HMCs that are used to shut down the relevant LPARs in case of a cluster split scenario.

The following sections contain the detailed PowerHA configuration of each scenario.

10.6.3 Initial PowerHA service status for each scenario

Each scenario has the same start status for the PowerHA and CAA service's status. We show the status in this section because we do not show it in each scenario.

PowerHA configuration

Example 10-9 shows PowerHA basic configuration with the **cltopinfo** command.

Example 10-9 PowerHA basic configuration that is shown with the cltopinfo command

```
# cltopinfo
Cluster Name: PHA_Cluster
Cluster Type: Standard
Heartbeat Type: Unicast
Repository Disk: hdisk1 (00fa2342a1093403)
```

There are 2 node(s) and 1 network(s) defined

```
NODE PHA170:
    Network net_ether_01
        PHASvc 172.16.51.172
        PHA170 172.16.51.170

NODE PHA171:
    Network net_ether_01
        PHASvc 172.16.51.172
        PHA171 172.16.51.171

Resource Group testRG
    Startup Policy Online On Home Node Only
    Fallback Policy Never Fallback
    Participating Nodes PHA170 PHA171
    Service IP Label PHASvc
    Volume Group sharevg
```

PowerHA service

Example 10-10 shows the PowerHA nodes status from each PowerHA node.

Example 10-10 PowerHA nodes status in each scenario before a cluster split

```
# clmgr -cv -a name,state,raw_state query node
# NAME:STATE:RAW_STATE
PHA170:NORMAL:ST_STABLE
PHA171:NORMAL:ST_STABLE
```

Example 10-11 shows the PowerHA RG status from each PowerHA node. The RG (testRG) is online on PHA170 node.

Example 10-11 PowerHA Resource Group status in each scenario before the cluster split

```
# clRGinfo -v

Cluster Name: PHA_Cluster

Resource Group Name: testRG
Startup Policy: Online On Home Node Only
Fallback Policy: Fallback To Next Priority Node In The List
Fallback Policy: Never Fallback
Site Policy: ignore
Node State
```

PHA170	ONLINE
PHA171	OFFLINE

CAA service status

Example 10-12 shows the CAA configuration with the `lscluster -c` command.

Example 10-12 Showing the CAA cluster configuration with the lscluster -c command

```
# lscluster -c
Cluster Name: PHA_Cluster
Cluster UUID: 28bf3ac0-b516-11e6-8007-faac90b6fe20
Number of nodes in cluster = 2
    Cluster ID for node PHA170: 1
    Primary IP address for node PHA170: 172.16.51.170
    Cluster ID for node PHA171: 2
    Primary IP address for node PHA171: 172.16.51.171
Number of disks in cluster = 1
    Disk = hdisk1 UUID = 58a286b2-fe51-5e39-98b1-43acf62025ab cluster_major = 0 cluster_minor = 1
Multicast for site LOCAL: IPv4 228.16.51.170 IPv6 ff05::e410:33aa
Communication Mode: unicast
Local node maximum capabilities: SPLT_MRG, CAA_NETMON, AUTO_REPOS_REPLACE, HNAME_CHG, UNICAST, IPV6, SITE
Effective cluster-wide capabilities: SPLT_MRG, CAA_NETMON, AUTO_REPOS_REPLACE, HNAME_CHG, UNICAST, IPV6, SITE
Local node max level: 50000
Effective cluster level: 50000
```

Example 10-13 shows the CAA configuration with the `lscluster -d` command.

Example 10-13 CAA cluster configuration

```
# lscluster -d
Storage Interface Query

Cluster Name: PHA_Cluster
Cluster UUID: 28bf3ac0-b516-11e6-8007-faac90b6fe20
Number of nodes reporting = 2
Number of nodes expected = 2

Node PHA170
Node UUID = 28945a80-b516-11e6-8007-faac90b6fe20
Number of disks discovered = 1
    hdisk1:
        State : UP
        uDid : 33213600507680284001D5800000000005C8B04214503IBMfcp
        uUid : 58a286b2-fe51-5e39-98b1-43acf62025ab
        Site uUid : 51735173-5173-5173-5173-517351735173
        Type : REPDISK

Node PHA171
Node UUID = 28945a3a-b516-11e6-8007-faac90b6fe20
Number of disks discovered = 1
    hdisk1:
        State : UP
        uDid : 33213600507680284001D5800000000005C8B04214503IBMfcp
        uUid : 58a286b2-fe51-5e39-98b1-43acf62025ab
        Site uUid : 51735173-5173-
```

Note: For production environments, configure additional backup repository disks.

PowerHA V7.2 supports up to six backup repository disks. It also supports automatic repository disk replacement in the event of repository disk failure. For more information, see *IBM PowerHA SystemMirror V7.2 for IBM AIX Updates*, SG24-8278.

Example 10-14 and Example 10-15 show output from PHA170 and PHA171 nodes with the **lscluster -m** command. The current heartbeat channel is the network.

Example 10-14 CAA information from node PHA170

```
# hostname
PHA170
# lscluster -m
Calling node query for all nodes...
Node query number of nodes examined: 2

Node name: PHA171
    Cluster shorthand id for node: 2
    UUID for node: 28945a3a-b516-11e6-8007-faac90b6fe20
    State of node: UP
        Reason: NONE
    Smoothed rtt to node: 7
    Mean Deviation in network rtt to node: 3
    Number of clusters node is a member in: 1
    CLUSTER NAME      SHID      UUID
    PHA_Cluster       0         28bf3ac0-b516-11e6-8007-faac90b6fe20
    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       172.16.51.170->172.16.51.171
```

Example 10-15 CAA information from node PHA171

```
# hostname
PHA171
# lscluster -m
Calling node query for all nodes...
Node query number of nodes examined: 2

Node name: PHA170
    Cluster shorthand id for node: 1
    UUID for node: 28945a80-b516-11e6-8007-faac90b6fe20
    State of node: UP
        Reason: NONE
    Smoothed rtt to node: 7
    Mean Deviation in network rtt to node: 3
    Number of clusters node is a member in: 1
    CLUSTER NAME      SHID      UUID
    PHA_Cluster       0         28bf3ac0-b516-11e6-8007-faac90b6fe20
    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->01	UP	IPv4	none	172.16.51.171->172.16.51.170

Example 10-16 shows the current heartbeat devices that are configured in the testing environment. There is not a SAN-based heartbeat device.

Example 10-16 CAA interfaces

```
# lscluster -g
Network/Storage Interface Query

Cluster Name: PHA_Cluster
Cluster UUID: 28bf3ac0-b516-11e6-8007-faac90b6fe20
Number of nodes reporting = 2
Number of nodes stale = 0
Number of nodes expected = 2

Node PHA171
Node UUID = 28945a3a-b516-11e6-8007-faac90b6fe20
Number of interfaces discovered = 2
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
        NDD type = 7 (NDD_IS088023)
        MAC address length = 6
        MAC address = FA:9D:66:B2:87:20
        Smoothed RTT across interface = 0
        Mean deviation in network RTT across interface = 0
        Probe interval for interface = 990 ms
        IFNET flags for interface = 0x1E084863
        NDD flags for interface = 0x0021081B
        Interface state = UP
        Number of regular addresses configured on interface = 1
        IPv4 ADDRESS: 172.16.51.171 broadcast 172.16.51.255 netmask
        255.255.255.0
            Number of cluster multicast addresses configured on interface = 1
            IPv4 MULTICAST ADDRESS: 228.16.51.170
            Interface number 2, dpcm
                IFNET type = 0 (none)
                NDD type = 305 (NDD_PINGCOMM)
                Smoothed RTT across interface = 750
                Mean deviation in network RTT across interface = 1500
                Probe interval for interface = 22500 ms
                IFNET flags for interface = 0x00000000
                NDD flags for interface = 0x00000009
                Interface state = UP RESTRICTED AIX_CONTROLLED

Node PHA170
Node UUID = 28945a80-b516-11e6-8007-faac90b6fe20
Number of interfaces discovered = 2
    Interface number 1, en0
        IFNET type = 6 (IFT_ETHER)
```

```

NDD type = 7 (NDD_IS088023)
MAC address length = 6
MAC address = FA:AC:90:B6:FE:20
Smoothed RTT across interface = 0
Mean deviation in network RTT across interface = 0
Probe interval for interface = 990 ms
IFNET flags for interface = 0x1E084863
NDD flags for interface = 0x0161081B
Interface state = UP
Number of regular addresses configured on interface = 1
IPv4 ADDRESS: 172.16.51.170 broadcast 172.16.51.255 netmask
255.255.255.0
Number of cluster multicast addresses configured on interface = 1
IPv4 MULTICAST ADDRESS: 228.16.51.170
Interface number 2, dpcm
IFNET type = 0 (none)
NDD type = 305 (NDD_PINGCOMM)
Smoothed RTT across interface = 594
Mean deviation in network RTT across interface = 979
Probe interval for interface = 15730 ms
IFNET flags for interface = 0x00000000
NDD flags for interface = 0x00000009
Interface state = UP RESTRICTED AIX_CONTROLLED

```

Note: To identify physical FC adapters that can be used in the PowerHA cluster as the SAN-based heartbeat, go to the [IBM Knowledge Center](#).

At the time of writing, there is no plan to support this feature for all 16-Gb FC adapters.

Shared file system status

Example 10-17 shows that the /sharefs file system is mounted on PHA170 node. This is because the RG is online on this node.

Example 10-17 Shared file system status

(0) root @ PHA170: /	# df	Filesystem	512-blocks	Free	%Used	Iused	%Iused	Mounted on
		...						
		/dev/sharelv	1310720	1309864	1%	4	1%	/sharefs

10.7 Scenario: Default split and merge policy

This section shows a scenario with the default split and merge policy.

10.7.1 Scenario description

Figure 10-35 shows the topology of the default split and merge scenario.

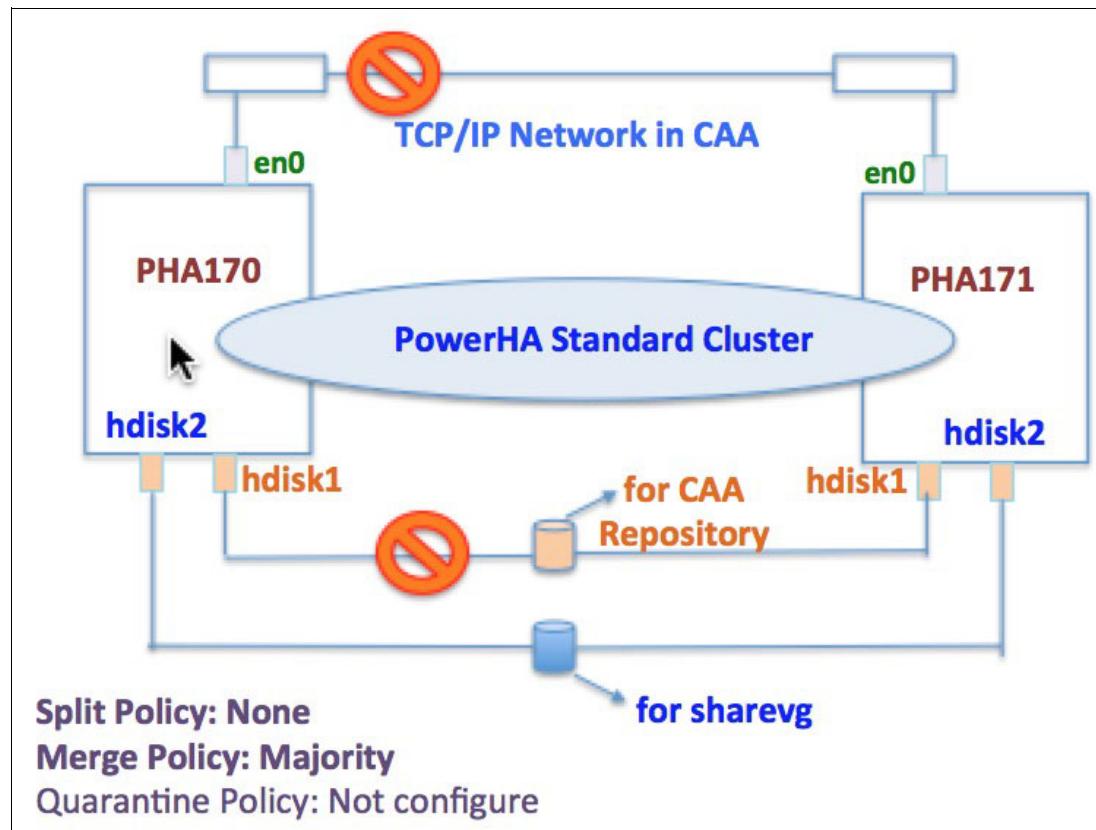


Figure 10-35 Topology of the default split and merge scenario

This scenario keeps the default configuration for the split and merge policy and does not set the quarantine policy. To simulate a cluster split, break the network communication between the two PowerHA nodes, and disable the repository disk access from the PHA170 node.

After a cluster split occurs, restore communications to generate a cluster *merge* event.

10.7.2 Split and merge configuration in PowerHA

In this scenario, it is not required to set specific parameters for the split and merge policy because it is the default policy. The **clmgr** command can be used to display the current policy, as shown in Example 10-18.

Example 10-18 The clmgr command displays the current split/merge settings

```
# clmgr view cluster SPLIT-MERGE
SPLIT_POLICY="none"
MERGE_POLICY="majority"
ACTION_PLAN="reboot"
<...>
```

Complete the following steps:

1. To change the current split and merge policy from the default by using SMIT, use the fast path of **smitty cm_cluster_sm_policy_chk**. Otherwise, run **smitty sysmirror** and select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy**. Example 10-19 shows the window where you select the None option.

Example 10-19 Split handling policy

Split Handling Policy

Move cursor to desired item and press Enter.

[None](#)
[TieBreaker](#)
[Manual](#)

After pressing Enter, the menu shows the policy, as shown in Example 10-20.

Example 10-20 Split and merge management policy

Split and Merge Management Policy

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

[Entry Fields]	
Split Handling Policy	None
Merge Handling Policy	Majority +
Split and Merge Action Plan	Reboot

2. Keep the default values and upon pressing Enter, you see the summary that is shown in Example 10-21.

Example 10-21 Successful setting of the split and merge policy

Command: OK	stdout: yes	stderr: no
-------------	-------------	------------

Before command completion, additional instructions may appear below.

The PowerHA SystemMirror split and merge policies have been updated.

Current policies are:

Split Handling Policy :	None
Merge Handling Policy :	Majority

Split and Merge Action Plan : Reboot
The configuration must be synchronized to make this change known across the cluster.

3. Synchronize the cluster. After the synchronization operation is complete, the cluster can be activated.

10.7.3 Cluster split

Before simulating a cluster split, check the status, as described in 10.6.3, “Initial PowerHA service status for each scenario” on page 351.

In this case, we sever all communications between two nodes at 21:55:23.

Steps of CAA and PowerHA on PHA170 node

The following events occur:

- ▶ 21:55:23: All communication between the two nodes is broken.
- ▶ 21:55:23: The PHA170 node marks REP_DOWN for the repository disk.
- ▶ 21:55:33: The PHA170 node CAA marks ADAPTER_DOWN for the PHA171 node.
- ▶ 21:56:02: The PHA170 node CAA marks NODE_DOWN for the PHA171 node.
- ▶ 21:56:02: PowerHA triggers the split_merge_prompt split event.
- ▶ 21:56:11: PowerHA triggers the split_merge_prompt quorum event.

Then, keep the current PowerHA service status.

Steps of CAA and PowerHA on PHA171 node

The following events occur:

- ▶ 21:55:23: All communication between the two nodes is broken.
- ▶ 21:55:33: PHA171 node CAA marked ADAPTER_DOWN for PHA170 node.
- ▶ 21:56:02: PHA171 node CAA marked NODE_DOWN for PHA170 node.
- ▶ 21:56:02: PowerHA triggered split_merge_prompt split event.
- ▶ 21:56:07: PowerHA triggered split_merge_prompt quorum event.

Note: The log file of the CAA service is /var/adm/ras/syslog.caa.

Then, PHA171 takes over the RG.

You see that PHA171 took over the RG while the RG is still online on the PHA170 node.

Note: The duration between REP_DOWN or ADAPTER_DOWN to NODE DOWN is 30 seconds. This duration is controlled by the CAA parameter `node_timeout`. Its value can be shown by running the following command:

```
# clctr1 -tune -L node_timeout
```

Here is the output:

NAME	DEF	MIN	MAX	UNIT	SCOPE	CUR
<hr/>						
node_timeout	20000	10000	600000	milliseconds	c n	30000

To change this value, either run the PowerHA `c1mgr` command or use the SMIT menu:

- ▶ From the SMIT menu, run `smitty sysmirror`, select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Manage the Cluster** → **Cluster heartbeat settings**, and then change the Node Failure Detection Timeout parameter.
- ▶ To use the `c1mgr` command, run the following command:

```
c1mgr modify cluster HEARTBEAT_FREQUENCY= <the value you want to set,  
default is 30>
```

Displaying the resource group status from the PHA170 node after the cluster split

Example 10-22 shows that the PHA170 node cannot get the PHA171 node's status.

Example 10-22 Resource group unknown status post split

```
# hostname  
PHA170  
# c1mgr -cv -a name,state,raw_state query node  
# NAME:STATE:RAW_STATE  
PHA170:NORMAL:ST_RP_RUNNING  
PHA171:UNKNOWN:UNKNOWN
```

Example 10-23 shows that the RG is online on PHA170 node.

Example 10-23 Resource group still online PHA170 post split

```
# hostname  
PHA170  
# c1RGinfo  
Node State  
-----  
PHA170 ONLINE  
PHA171 OFFLINE
```

Example 10-24 shows that the VG sharevg is varied on, and the file system /sharefs is mounted on PHA170 node and is writable.

Example 10-24 Volume group still online PHA170 post split

```
# hostname
PHA170
# lsvg sharevg
VOLUME GROUP:      sharevg          VG IDENTIFIER:
00fa4b4e00004c0000000158a8e55930
VG STATE:          active           PP SIZE:      32 megabyte(s)
VG PERMISSION:     read/write      TOTAL PPs:    29 (928 megabytes)
MAX LVs:           256             FREE PPs:    8 (256 megabytes)

# df
Filesystem   512-blocks   Free %Used   Iused %Iused Mounted on
...
/dev/sharelv 1310720    1309864    1%        4       1% /sharefs
```

Displaying the resource group status from the PHA171 node after the cluster split

Example 10-25 shows that the PHA171 node cannot get the PHA170 node's status.

Example 10-25 Resource group warning and unknown on PHA171

```
# hostname
PHA171
# clmgr -cv -a name,state,raw_state query node
# NAME:STATE:RAW_STATE
PHA170:UNKNOWN:UNKNOWN
PHA171:WARNING:WARNING
```

Example 10-26 shows that the RG is online on PHA171 node too.

Example 10-26 Resource group online PHA171 post split

Node	State
-----	-----
PHA170	OFFLINE
PHA171	ONLINE

Example 10-27 shows that the VG sharevg is varied on and the file system /sharefs is mounted on PHA171 node, and it is writable too.

Example 10-27 Sharevg online on PHA171 post split

```
# hostname
PHA171
# lsvg sharevg
VOLUME GROUP:      sharevg          VG IDENTIFIER:
00fa4b4e00004c0000000158a8e55930
VG STATE:          active           PP SIZE:      32 megabyte(s)
VG PERMISSION:     read/write      TOTAL PPs:    29 (928 megabytes)
```

```

MAX LVs:          256           FREE PPs:      8 (256 megabytes)
<...>

# df
Filesystem   512-blocks   Free %Used   Iused %Iused Mounted on
<...>
/dev/sharelv   1310720    1309864     1%        4      1% /sharefs

```

As seen in Example 10-7 on page 341, the /sharefs file system is mounted on both nodes and in writable mode. Applications on two nodes can write at the same time. This is risky and easily can result in data corruption.

Note: This situation must always be avoided in a production environment.

10.7.4 Cluster merge

After the cluster split occurs, the RG was online on PHA171 node while it was still online on the PHA170 node. When the PowerHA cluster heartbeat communication is restored at 22:24:08, a PowerHA merge event was triggered.

The default merge policy is *Majority* and the action plan is *Reboot*. However, in our case, the rule in the cluster merge event is:

The node that has a lower node ID survives, and the other node is restarted by RSCT.

This rule is also introduced in 10.2.2, “Merge policy” on page 324.

Example 10-28 shows how to display a PowerHA node’s node ID. You can see that PHA170 has the lower ID, so it is expected that PHA171 node is restarted.

Example 10-28 How to show a node ID for PowerHA nodes

```

# ./cl_query_hn_id
CAA host PHA170 with node id 1 corresponds to PowerHA node PHA170
CAA host PHA171 with node id 2 corresponds to PowerHA node PHA171

# lscluster -c
Cluster Name: PHA_Cluster
Cluster UUID: 28bf3ac0-b516-11e6-8007-faac90b6fe20
Number of nodes in cluster = 2
  Cluster ID for node PHA170: 1
    Primary IP address for node PHA170: 172.16.51.170
    Cluster ID for node PHA171: 2
    Primary IP address for node PHA171: 172.16.51.171
Number of disks in cluster = 1
  Disk = hdisk1 UUID = 58a286b2-fe51-5e39-98b1-43acf62025ab cluster_major = 0 cluster_minor = 1
Multicast for site LOCAL: IPv4 228.16.51.170 IPv6 ff05::e410:33aa

```

Example 10-29 shows that the PHA171 node was rebooted at 22:25:02.

Example 10-29 Display error report with the errpt -c command

```

# hostname
PHA171
# errpt -c
A7270294  1127222416 P S cluster0      A merge has been detected.
78142BB8  1127222416 I O ConfigRM      ConfigRM received Subcluster Merge event

```

F0851662	1127222416 I S ConfigRM	The sub-domain containing the local node
9DEC29E1	1127222416 P O cthags	Group Services daemon exit to merge doma
9DBCFDEE	1127222516 T O errdemon	ERROR LOGGING TURNED ON
69350832	1127222516 T S SYSPROC	SYSTEM SHUTDOWN BY USER

```
# errpt -aj 69350832
LABEL:          REBOOT_ID
IDENTIFIER:    69350832

Date/Time:      Sun Nov 27 22:25:02 CST 2016
Sequence Number: 701
Machine Id:     00FA23424C00
Node Id:        PHA171
Class:          S
Type:           TEMP
WPAR:           Global
Resource Name:  SYSPROC

Description
SYSTEM SHUTDOWN BY USER

Probable Causes
SYSTEM SHUTDOWN

Detail Data
USER ID
      0
0=SOFT IPL 1=HALT 2=TIME REBOOT
      0
TIME TO REBOOT (FOR TIMED REBOOT ONLY)
      0
PROCESS ID
      13959442
PARENT PROCESS ID
      4260250
PROGRAM NAME
hagsd
PARENT PROGRAM NAME
srcmstr
```

10.7.5 Scenario summary

With the default split and merge policy, when a cluster split happens, the RG is online on both PowerHA nodes. This is a risky situation that can result in data corruption. Careful planning must be done to avoid this scenario.

10.8 Scenario: Split and merge policy with a disk tie breaker

This section describes the split and merge policy scenario with a disk tie breaker.

10.8.1 Scenario description

Figure 10-36 is the reference topology for this scenario.

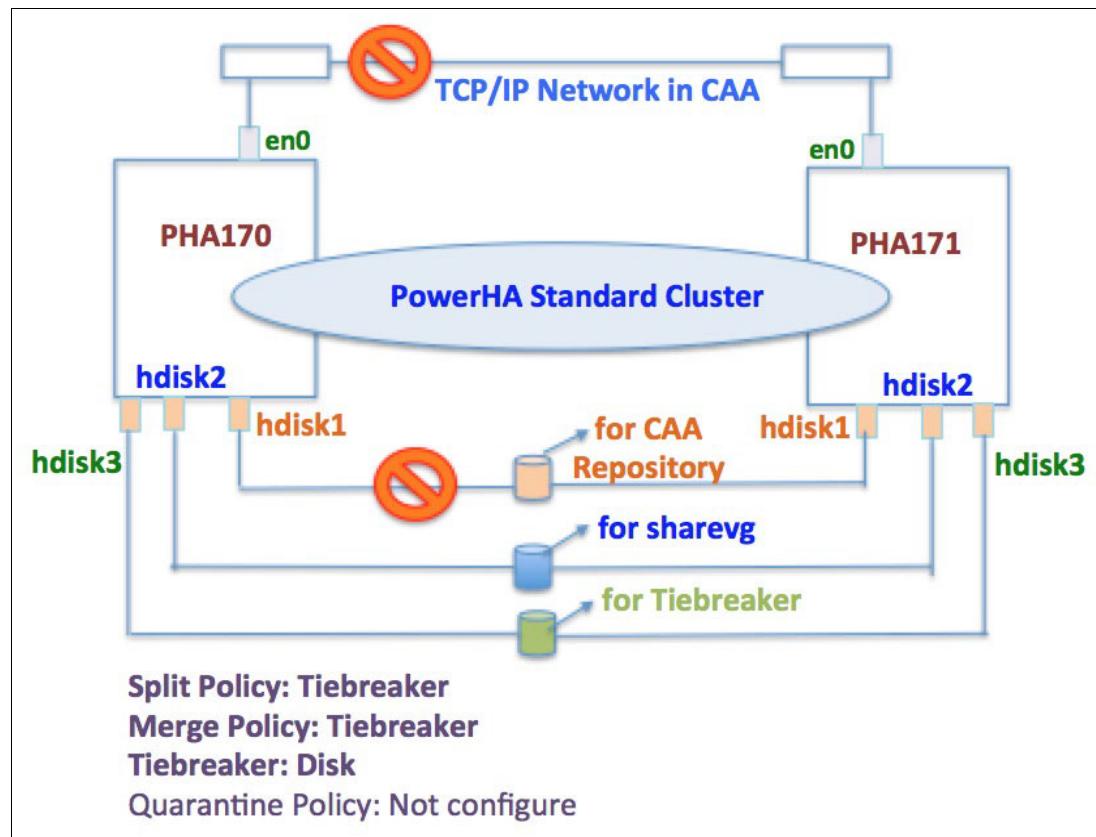


Figure 10-36 Split and merge topology scenario

There is one new shared disk, hdisk3, that is added in this scenario, which is used for the disk tie breaker.

Note: When using a tie-breaker disk for split and merge recovery handling, the disk must also be supported by the **devrsrv** command. This command is part of the AIX operating system.

At the time of writing, the EMC PowerPath disks are not supported for use as a tie-breaker disk.

Note: The tie-breaker disk is set to `no_reserve` for the `reserve_policy` with the **chdev** command before the start of the PowerHA service on both nodes. Otherwise, the tie-breaker policy cannot take effect in a cluster split event.

10.8.2 Split and merge configuration in PowerHA

Complete the following steps:

1. The fast path to set the split and merge policy is `smitty cm_cluster_sm_policy_chk`. The whole path is running `smitty sysmirror` and selecting **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy**.

Example 10-30 shows the window to select the split handling policy; in this case, TieBreaker is selected.

Example 10-30 TieBreaker split handling policy

Split Handling Policy

Move cursor to desired item and press Enter.

None
TieBreaker
Manual

2. After pressing Enter, select the Disk option, as shown in Example 10-31.

Example 10-31 Select Tiebreaker

Select TieBreaker Type

Move cursor to desired item and press Enter.

Disk
NFS

F1=Help Esc+8=Image	F2=Refresh Esc+0=Exit	F3=Cancel Enter=Do
------------------------	--------------------------	-----------------------

3. Pressing Enter shows the disk tie breaker configuration window, as shown in Example 10-32. The merge handling policy is TieBreaker too, and you cannot change it. Also, keep the default action plan as Reboot.

Example 10-32 Disk tiebreaker configuration

Disk TieBreaker Configuration

Type or select values in entry fields.

Press Enter AFTER making all desired changes.

Split Handling Policy	[Entry Fields]
Merge Handling Policy	TieBreaker
* Select Tie Breaker	TieBreaker
Split and Merge Action Plan	<input type="checkbox"/> + Reboot

4. In the Select Tie Breaker field, press F4 to list the disks that can be used for the disk tie breaker, as shown in Example 10-33. We select hdisk3.

Example 10-33 Select tie breaker disk

Select Tie Breaker

Move cursor to desired item and press Enter.

None

hdisk3 (00fa2342a10932bf) on all cluster nodes

F1=Help

F2=Refresh

F3=Cancel

Esc+8=Image

Esc+0=Exit

Enter=Do

/=Find

n=Find Next

5. Press Enter to display the summary, as shown in Example 10-34.

Example 10-34 Select the disk tie breaker status

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

hdisk3 changed

The PowerHA SystemMirror split and merge policies have been updated.

Current policies are:

Split Handling Policy : Tie Breaker

Merge Handling Policy : Tie Breaker

Tie Breaker : hdisk3

Split and Merge Action Plan : Reboot

The configuration must be synchronized to make this change known across the cluster.

6. Synchronize the cluster. After the synchronization operation is complete, the cluster can be activated.
7. Run the **clmgr** command to query the current split and merge policy, as shown in Example 10-35.

Example 10-35 Display the newly set split and merge policies

```
# clmgr view cluster SPLIT-MERGE
SPLIT_POLICY="tiebreaker"
MERGE_POLICY="tiebreaker"
ACTION_PLAN="reboot"
TIEBREAKER="hdisk3"
<...>
```

After the PowerHA service start completes, you see that the reserve_policy of this disk is changed to PR_exclusive and one reserve key value is generated for this disk on each node. This disk is not reserved by any of the nodes. Example 10-36 shows the result from the two nodes.

Example 10-36 Reserve_policy on each node

```
(127) root @ PHA170: /
# lsattr -El hdisk3|egrep "PR_key_value|reserve_policy"
PR_key_value      2763601723737305030 Persistant Reserve Key Value      True+
reserve_policy    PR_exclusive          Reserve Policy                True+

# devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name          : hdisk3
Device Open On Current Host? : NO
ODM Reservation Policy : PR EXCLUSIVE
ODM PR Key Value   : 2763601723737305030
Device Reservation State : NO RESERVE
Registered PR Keys  : No Keys Registered
PR Capabilities Byte[2] : 0x11 CRH  PTPL_C
PR Capabilities Byte[3] : 0x80
PR Types Supported   : PR_WE_AR  PR_EA_RO  PR_WE_RO  PR_EA  PR_WE  PR_EA_AR

(0) root @ PHA171: /
# lsattr -El hdisk3|egrep "PR_key_value|reserve_policy"
PR_key_value      6664187022250383046 Persistant Reserve Key Value      True+
reserve_policy    PR_exclusive          Reserve Policy                True+

# devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name          : hdisk3
Device Open On Current Host? : NO
ODM Reservation Policy : PR EXCLUSIVE
ODM PR Key Value   : 6664187022250383046
Device Reservation State : NO RESERVE
Registered PR Keys  : No Keys Registered
PR Capabilities Byte[2] : 0x11 CRH  PTPL_C
PR Capabilities Byte[3] : 0x80
PR Types Supported   : PR_WE_AR  PR_EA_RO  PR_WE_RO  PR_EA  PR_WE  PR_EA_ARfor
```

10.8.3 Cluster split

Before simulating a cluster split, check the current cluster status. For more information, see 10.6.3, “Initial PowerHA service status for each scenario” on page 351.

When the tie breaker split and merge policy is enabled, the rule is the TBGL node has higher priority to the reserve tiebreaker device than other nodes. If this node reserves the tie-breaker device successfully, then other nodes are restarted.

For this scenario, Example 10-37 shows that the PHA171 node is the current TBGL. So, it is expected that the PHA171 node reserves the tie breaker device, and the PHA170 node is restarted. Any RG on the PHA170 node is taken over to the PHA171 node.

Example 10-37 Display the tiebreaker group leader

```
# lssrc -ls IBM.ConfigRM|grep Group
Group IBM.ConfigRM:
  GroupLeader: PHA171, 0xdc7bf2c9d20096c6, 2
  TieBreaker GroupLeader: PHA171, 0xdc7bf2c9d20096c6, 2
```

To change the TBGL manually, see 10.8.4, “How to change the tie breaker group leader manually” on page 370.

In this case, we broke all communication between the two nodes at 01:36:12.

Result and log on the PHA170 node

The following events occur:

- ▶ 01:36:12: All communication between the two nodes is broken.
- ▶ 01:36:22: The PHA170 node CAA marks ADAPTER_DOWN for the PHA171 node.
- ▶ 01:36:52: The PHA170 node CAA marks NODE_DOWN for the PHA171 node.
- ▶ 01:36:52: PowerHA triggers the split_merge_prompt split event.
- ▶ 01:36:57: PowerHA triggers the split_merge_prompt quorum event.
- ▶ 01:37:00: The PHA170 node restarts.

Example 10-38 shows output of the **errpt** command on the PHA170 node. The PHA170 node restarts at 01:37:00.

Example 10-38 PHA170 restart post split

```
C7E7362C 1128013616 T S cluster0      Node is heartbeating solely over disk or
4D91E3EA 1128013616 P S cluster0      A split has been detected.
2B138850 1128013616 I O ConfigRM      ConfigRM received Subcluster Split event
DC73C03A 1128013616 T S fscsil        SOFTWARE PROGRAM ERROR
<...>
C62E1EB7 1128013616 P H hdisk1       DISK OPERATION ERROR
<...>
80732E3   1128013716 P S ConfigRM      The operating system is being rebooted t

# errpt -aj B80732E3|more
-----
LABEL:           CONFIGRM_REBOOTOS_E
IDENTIFIER:     B80732E3

Date/Time:      Mon Nov 28 01:37:00 CST 2016
Sequence Number: 1620
Machine Id:    00FA4B4E4C00
Node Id:       PHA170
Class:          S
Type:           PERM
WPAR:           Global
Resource Name:  ConfigRM

Description
The operating system is being rebooted to ensure that critical resources are
stopped so that another sub-domain that has operational quorum may recover
```

these resources without causing corruption or conflict.

Probable Causes

Critical resources are active and the active sub-domain does not have operational quorum.

Failure Causes

Critical resources are active and the active sub-domain does not have operational quorum.

Recommended Actions

After node finishes rebooting, resolve problems that caused the operational quorum to be lost.

Detail Data

DETECTING MODULE

RSCT,PeerDomain.C,1.99.22.299,23992

ERROR ID

Result and log on the PHA171 node

The following events occur:

- ▶ 01:36:12: All communication between the two nodes is broken.
- ▶ 01:36:22: The PHA171 node CAA marks ADAPTER_DOWN for the PHA170 node.
- ▶ 01:36:52: The PHA171 node CAA marks NODE_DOWN for the PHA170 node.
- ▶ 01:36:52: PowerHA triggers a split_merge_prompt split event.
- ▶ 01:37:04: PowerHA triggers a split_merge_prompt quorum event, and then PHA171 takes over the RG.
- ▶ 01:37:15: PowerHA completes the RG takeover operation.

As shown in Example 10-38 on page 368 with the time stamp, PHA170 restarts at 01:37:00. PHA171 starts the takeover of the RG at 01:37:04. There is no opportunity for both nodes to mount the /sharefs file system at the same time so that the data integrity is maintained.

The PHA171 node holds the tiebreaker disk during as cluster split

Example 10-39 shows that the tiebreaker disk is reserved by the PHA171 node after the cluster split event happens.

Example 10-39 Tiebreaker disk reservation from PHA171

```
# hostname
PHA171

# lsattr -El hdisk3|egrep "PR_key_value|reserve_policy"
PR_key_value      6664187022250383046 Persistant Reserve Key Value      True+
reserve_policy    PR_exclusive          Reserve Policy            True+

# devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name          : hdisk3
Device Open On Current Host?   : NO
```

ODM Reservation Policy	:	PR EXCLUSIVE
ODM PR Key Value	:	6664187022250383046
Device Reservation State	:	PR EXCLUSIVE
PR Generation Value	:	152
PR Type	:	PR_WE_RO (WRITE EXCLUSIVE, REGISTRANTS ONLY)
PR Holder Key Value	:	6664187022250383046
Registered PR Keys	:	6664187022250383046 6664187022250383046
PR Capabilities Byte[2]	:	0x11 CRH PTPL_C
PR Capabilities Byte[3]	:	0x81 PTPL_A
PR Types Supported	:	PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

10.8.4 How to change the tie breaker group leader manually

To change the TBGL manually, simply restart the current TBGL. For example, if the PHA170 node is the current TBGL, to change PHA171 as the tie breaker leader, restart the PHA170 node. During this restart, the TBGL is switched to the PHA171 node. After the PHA170 comes back, the group leader does not change until PHA171 is shut down or restarts.

10.8.5 Cluster merge

After the PHA170 node restart completes, restore all communications between the two nodes. If you want to enable the tiebreaker disk on the PHA170 node, just after the FC link is restored, run the **cfgmgr** command. Then, the paths of the tiebreaker disk are in active status, as shown in Example 10-40.

Example 10-40 Path status post split

```
# hostname
PHA170

# lspath -l hdisk1
Missing hdisk1 fscsi1
Missing hdisk1 fscsi1

-> After run 'cfgmgr' command
# lspath -l hdisk1
Enabled hdisk1 fscsi1
Enabled hdisk1 fscsi1
```

Within 1 minute of the repository disk being enabled, the CAA services start automatically. You can monitor the process by viewing the /var/adm/ras/syslog.caa log file.

Using the **lscuster -m** command, check whether the CAA service started. When ready, start the PowerHA service with the **smitty clstart** or **clmgr start node PHA170** command.

You can also bring the CAA services and PowerHA services online together manually by running the following command:

```
clmgr start node PHA170 START_CAA=yes
```

During the start of the PowerHA services, the tie breaker device reservation is released on the PHA171 node automatically. Example 10-41 shows the device reservation state after the PowerHA service starts.

Example 10-41 Disk reservation post merge

```
# hostname
PHA171

# devrsrv -c query -l hdisk3
Device Reservation State Information
=====
Device Name          : hdisk3
Device Open On Current Host? : NO
ODM Reservation Policy : PR EXCLUSIVE
ODM PR Key Value   : 6664187022250383046
Device Reservation State : NO RESERVE
Registered PR Keys  : No Keys Registered
PR Capabilities Byte[2] : 0x11 CRH PTPL_C
PR Capabilities Byte[3] : 0x81 PTPL_A
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR
```

10.8.6 Scenario summary

If you set a disk tie breaker as split and merge policy for the PowerHA cluster, when the cluster split occurs, the TBGL has a higher priority to reserve the tie breaker device. Other nodes restart. The RGs are online on the surviving node.

During the cluster merge process, the tiebreaker reservation is automatically released.

10.9 Scenario: Split and merge policy with the NFS tie breaker

This section describes the split and merge scenario with the NFS tie-breaker policy.

10.9.1 Scenario description

Figure 10-37 shows the topology of this scenario.

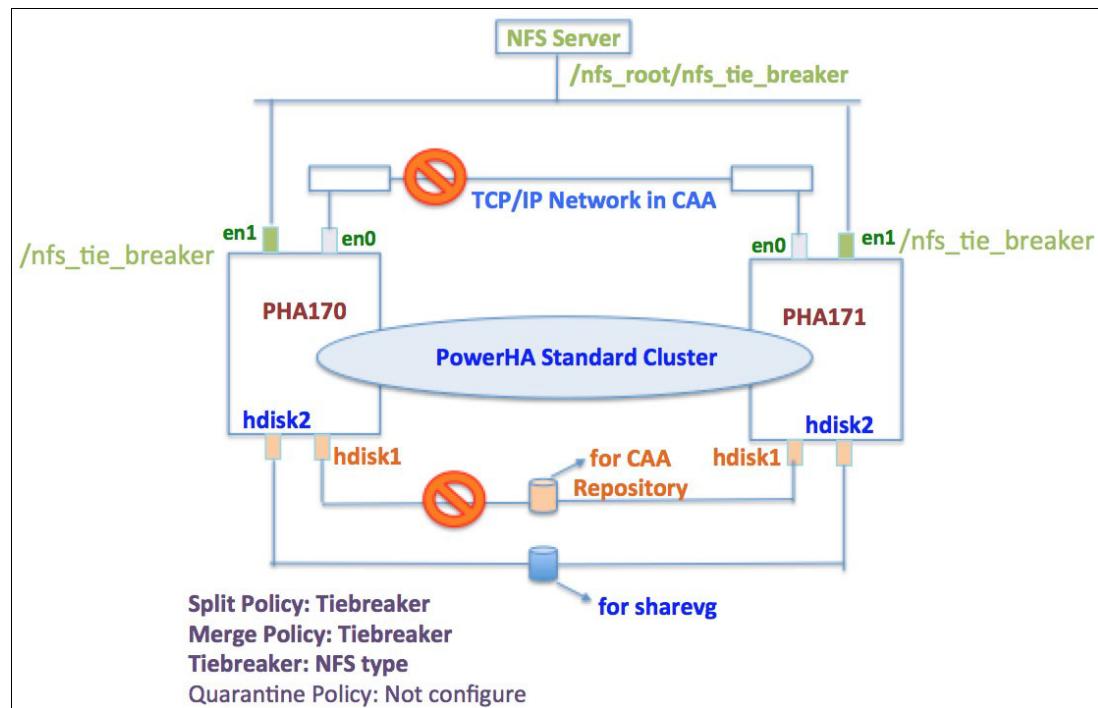


Figure 10-37 Split and merge topology scenario with the NFS tie breaker

In this scenario, there is one NFS server. Each PowerHA node has one network interface, en1, which is used to communicate with the NFS server. The NFS tie breaker requires NFS protocol version 4.

10.9.2 Setting up the NFS environment

On the NFS server, complete the following steps:

1. Edit /etc/hosts and add the PowerHA nodes definition, as shown in Example 10-42.

Example 10-42 Add nodes to NFS server /etc/hosts

```
cat /etc/hosts
<...>
172.16.15.242  PHA170_hmc
172.16.15.243  PHA171_hmc
```

2. Create the directory for export by running the following command:

```
mkdir -p /nfs_tiebreaker
```

3. Configure the NFS domain by running the following command:

```
chnfsdom nfs_local_domain
```

- Start the nfsrsgyd service by running the following command:

```
starts rc -s nfsrsgyd
```

- Change the NFS version 4 root location to / by running the following command:

```
chnfs -r /
```

- Add the /nfs_tiebreaker directory to the export list by running the following command:

```
/usr/sbin/mknfsexp -d '/nfs_tiebreaker' '-B' '-v '4' '-S  
'sys,krb5p,krb5i,krb5,dh' '-t 'rw' '-r 'PHA170_hmc,PHA171_hmc'
```

Alternatively, you can run **smitty nfs**, as shown in Example 10-43.

Example 10-43 NFS add directory to export

Add a Directory to Exports List

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

[TOP]	[Entry Fields]
* Pathname of directory to export	[/nfs_tiebreaker]
Anonymous UID	[-2]
Public filesystem?	no +
* Export directory now, system restart or both	both
+ Pathname of alternate exports file	[]
Allow access by NFS versions	[4] +
External name of directory (NFS V4 access only)	[]
Referral locations (NFS V4 access only)	[]
Replica locations	[]
Ensure primary hostname in replica list	yes +
Allow delegation?	[]
Scatter	none +
* Security method 1	[sys,krb5p,krb5i,krb5,dh] +
* Mode to export directory	read-write +
Hostname list. If exported read-mostly	[]
Hosts & netgroups allowed client access	[]
Hosts allowed root access	[PHA170_hmc1,PHA171_hmc]

You can verify that the directory is exported by viewing the /etc/exports file, as shown in Example 10-44.

Example 10-44 The /etc/exports file

```
# cat /etc/exports  
/nfs_tiebreaker -vers=4,sec=sys:krb5p:krb5i:krb5:dh,rw,root=PHA170_hmc:PHA171_hmc
```

On the NFS clients and PowerHA nodes, complete the following tasks:

- Edit /etc/hosts and add the NFS server definition, as shown in Example 10-45.

Example 10-45 NFS clients /etc/hosts

```
# hostname  
PHA170  
# cat /etc/hosts  
...  
172.16.51.170 PHA170
```

```
172.16.51.171  PHA171  
172.16.51.172  PHASvc  
172.16.15.242  PHA170_hmc
```

172.16.15.222 nfsserver

- ▶ Now, verify that the new NFS mount point can be mounted on all the nodes, as shown in Example 10-46.

Example 10-46 Mount the NFS directory

```
(0) root @ PHA170: /  
# mount -o vers=4 nfsserver:/nfs_tiebreaker /mnt  
  
# df|grep mnt  
nfsserver:/nfs_tiebreaker      786432    429256   46%    11704    20% /mnt  
  
# echo "test.." > /mnt/1.out  
# cat /mnt/1.out  
test..  
# rm /mnt/1.out  
  
# umount /mnt
```

10.9.3 Setting the NFS split and merge policies

When the NFS configuration finishes, configure PowerHA by completing the following steps:

1. The fast path to set the split and merge policy is `smitty cm_cluster_sm_policy_chk`. The full path is to run `smitty sysmirror` and select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy**.
2. Select TieBreaker, as shown in Example 10-30 on page 365. After pressing Enter, select the NFS option, as shown in Example 10-47.

Example 10-47 NFS TieBreaker

Select TieBreaker Type

Move cursor to desired item and press Enter.

Disk
NFS

F1=Help Esc+8=Image	F2=Refresh Esc+0=Exit	F3=Cancel Enter=Do
------------------------	--------------------------	-----------------------

- After pressing Enter, the NFS tiebreaker configuration panel opens, as shown in Example 10-48. The merge handling policy is TieBreaker too, and it cannot be changed. Also, keep the default action plan as Reboot.

Example 10-48 NFS TieBreaker configuration menu

NFS TieBreaker Configuration	
Type or select values in entry fields.	
Press Enter AFTER making all desired changes.	
Split Handling Policy	[Entry Fields]
Merge Handling Policy	NFS
* NFS Export Server	NFS
* Local Mount Directory	[nfsserver]
* NFS Export Directory	[/nfs_tiebreaker]
Split and Merge Action Plan	[/nfs_tiebreaker]
	Reboot

After pressing enter, Example 10-49 shows the NFS TieBreaker configuration summary.

Example 10-49 NFS TieBreaker configuration summary

Command: OK	stdout: yes	stderr: no
-------------	-------------	------------

Before command completion, additional instructions may appear below.

The PowerHA SystemMirror split and merge policies have been updated.

Current policies are:

Split Handling Policy :	NFS
Merge Handling Policy :	NFS
NFS Export Server :	nfsserver
Local Mount Directory :	/nfs_tiebreaker
NFS Export Directory :	/nfs_tiebreaker
Split and Merge Action Plan :	Reboot

The configuration must be synchronized to make this change known across the cluster.

The configuration is added to the HACMPsplitmerge ODM database, as shown in Example 10-50.

Example 10-50 HACMPsplitmerge ODM

```
# odmget HACMPsplitmerge
```

```
HACMPsplitmerge:
    id = 0
    policy = "split"
    value = "NFS"

HACMPsplitmerge:
    id = 0
    policy = "merge"
    value = "NFS"

HACMPsplitmerge:
    id = 0
```

```

        policy = "action"
        value = "Reboot"

HACMPsplitmerge:
    id = 0
    policy = "nfs_quorumserver"
    value = "nfsserver"

HACMPsplitmerge:
    id = 0
    policy = "local_quorundirectory"
    value = "/nfs_tiebreaker"

HACMPsplitmerge:
    id = 0
    policy = "remote_quorundirectory"
    value = "/nfs_tiebreaker"

```

4. Synchronize the cluster. After the synchronization operation completes, the cluster can be activated.

Upon the cluster start, the PowerHA nodes mount the NFS automatically on both nodes, as shown in Example 10-51.

Example 10-51 NFS mount on both nodes

```

# clcmd mount|egrep -i "node|nfs"
NODE PHA171
  node      mounted      mounted over     vfs      date      options
  nfsserver /nfs_tiebreaker /nfs_tiebreaker nfs4  Dec 01 08:50 vers=4,fg,soft,retry=1,timeo=10

NODE PHA170
  node      mounted      mounted over     vfs      date      options
  nfsserver /nfs_tiebreaker /nfs_tiebreaker nfs4  Dec 01 08:50 vers=4,fg,soft,retry=1,timeo=10

```

10.9.4 Cluster split

If you enable the tie breaker split and merge policy, in a cluster split scenario, the rule is that the TBGL node has a higher priority to reserve a tie-breaker device than the other nodes. The node add its node name to the PowerHA_NFS_Reserve file, gets the reservation, and locks it. In this scenario, the file is in the /nfs_tiebreaker directory.

In our case, the PHA171 node is the current TBGL, as shown in Example 10-52 on page 377. So, it is expected that the PHA171 node survives and the PHA170 node restarts. The RG on the PHA170 node is taken to the PHA171 node.

Example 10-52 NFS Tiebreaker groupleader

```
# lssrc -ls IBM.ConfigRM|grep Group
Group IBM.ConfigRM:
  GroupLeader: PHA171, 0xdc7bf2c9d20096c6, 2
  TieBreaker GroupLeader: PHA171, 0xdc7bf2c9d20096c6, 2
```

To change the TBGL manually, see 10.8.4, “How to change the tie breaker group leader manually” on page 370.

In this case, we broke all communication between both nodes at 07:23:49.

Result and log on the PHA170 node

The following events occur:

- ▶ 07:23:49: All communication between the two nodes is broken.
- ▶ 07:23:59: The PHA170 node CAA marks ADAPTER_DOWN for the PHA171 node.
- ▶ 07:24:29: The PHA170 node CAA mark NODE_DOWN for the PHA171 node.
- ▶ 07:24:29: PowerHA triggers the split_merge_prompt split event.
- ▶ 07:24:35: PowerHA triggers the split_merge_prompt quorum event.
- ▶ 07:24:38: The PHA170 node is restarted by RSCT.

Example 10-53 shows the output of the **errpt** command on the PHA170 node. This node restarts at 07:24:38.

Example 10-53 Errpt on PHA170

C7E7362C	1128072416 T S cluster0	Node is heartbeating solely over disk or
4D91E3EA	1128072416 P S cluster0	A split has been detected.
2B138850	1128072416 I O ConfigRM	ConfigRM received Subcluster Split event
<...>		
A098BF90	1128072416 P S ConfigRM	The operational quorum state of the acti
AB59ABFF	1128072416 U U LIBLVM	Remote node Concurrent Volume Group fail
421B554F	1128072416 P S ConfigRM	The operational quorum state of the acti
AB59ABFF	1128072416 U U LIBLVM	Remote node Concurrent Volume Group fail
B80732E3	1128072416 P S ConfigRM	The operating system is being rebooted t

```
# errpt -aj B80732E3
LABEL:           CONFIGRM_REBOOTOS_E
IDENTIFIER:     B80732E3

Date/Time:      Mon Nov 28 07:24:38 CST 2016
Sequence Number: 1839
Machine Id:    00FA4B4E4C00
Node Id:       PHA170
Class:          S
Type:           PERM
WPAR:          Global
Resource Name: ConfigRM
```

Description

The operating system is being rebooted to ensure that critical resources are stopped so that another sub-domain that has operational quorum may recover these resources without causing corruption or conflict.

Probable Causes

Critical resources are active and the active sub-domain does not have

operational quorum.

Failure Causes

Critical resources are active and the active sub-domain does not have operational quorum.

Recommended Actions

After node finishes rebooting, resolve problems that caused the operational quorum to be lost.

Detail Data

DETECTING MODULE

RSCT,PeerDomain.C,1.99.22.299,23992

ERROR ID

REFERENCE CODE

Result and log on the PHA171 node

The following events occur:

- ▶ 07:23:49: All communication between the two nodes is broken.
- ▶ 07:24:02: The PHA170 node CAA marks ADAPTER_DOWN for the PHA171 node.
- ▶ 07:24:32: The PHA170 node CAA marks NODE_DOWN for the PHA171 node.
- ▶ 07:24:32: PowerHA triggers a split_merge_prompt split event.
- ▶ 07:24:42: PowerHA triggers a split_merge_prompt quorum event.
- ▶ 07:24:43: PowerHA starts to online RG on the PHA171 node.
- ▶ 07:25:03: Complete the RG online operation.

From the time stamp information that is shown in Example 10-53 on page 377, PHA170 restarts at 07:24:38, and PHA171 starts to take over RGs at 07:24:43. There is no opportunity for both nodes to mount the /sharefs file system at the same time, so the data integrity is maintained.

Example 10-54 shows that the PHA171 node wrote its node name into the PowerHA_NFS_Reserved file successfully.

Example 10-54 NFS file that is written with the node name

```
# hostname
PHA171

# pwd
/nfs_tiebreaker

# ls -l
total 8
-rw-r--r--    1 nobody    nobody          257 Nov 28 07:24 PowerHA_NFS_Reserve
drwxr-xr-x    2 nobody    nobody          256 Nov 28 04:06
PowerHA_NFS_ReservewviewFilesDir

# cat PowerHA_NFS_Reserve
PHA171
```

10.9.5 Cluster merge

The steps are similar to 10.8.5, “Cluster merge” on page 370.

After CAA services start successfully, the PowerHA_NFS_Reserve file is cleaned up for the next cluster split event. Example 10-55 shows that the size of PowerHA_NFS_Reserve file is changed to zero after the CAA service is restored.

Example 10-55 NFS file zeroed out after the CAA is restored

```
# ls -l
total 0
-rw-r--r--    1 nobody    nobody          0 Nov 28 09:05 PowerHA_NFS_Reserve
drwxr-xr-x    2 nobody    nobody        256 Nov 28 09:05 PowerHA_NFS_ReserveviewFilesDir
```

10.9.6 Scenario summary

When the NFS tiebreaker is set as a split and merge policy when a cluster split occurs, the TBGL has a higher priority to reserve NFS. Other nodes restart, and the RGs are online on the surviving node.

During the cluster merge process, the NFS tiebreaker reservations are released automatically.

10.10 Scenario: Split and merge policy is manual

This section presents a split and merge manual policy scenario.

10.10.1 Scenario description

Figure 10-38 shows the topology of this scenario.

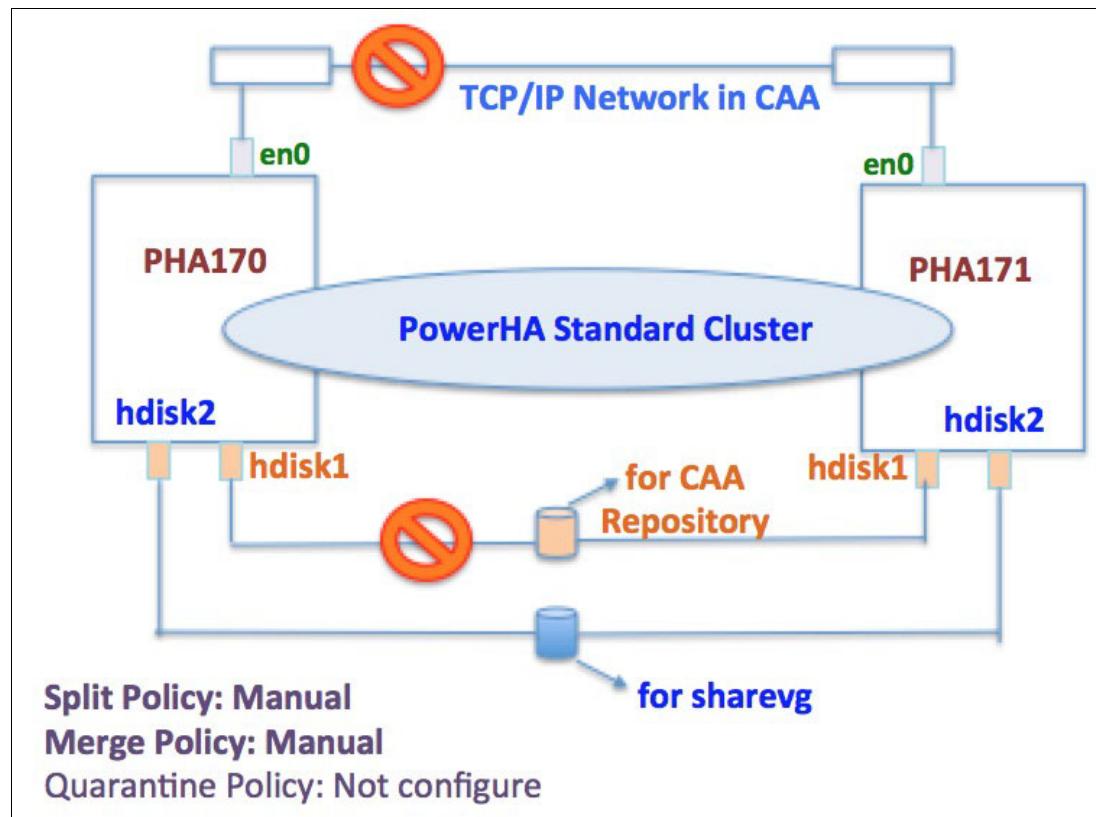


Figure 10-38 Manual split merge cluster topology

10.10.2 Split and merge configuration in PowerHA

The fast path to set the split and merge policy is `smitty cm_cluster_sm_policy_chk`. The full path is running `smitty sysmirror` and then selecting **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy**.

We select Manual for the split handling policy, as shown in Example 10-56.

Example 10-56 Manual split policy

Split Handling Policy

Move cursor to desired item and press Enter.

None
TieBreaker
Manual

After pressing Enter, the configuration panel opens, as shown in Example 10-57.

Example 10-57 Manual split and merge configuration menu

Split and Merge Management Policy

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

[Entry Fields]	
Split Handling Policy	Manual
Merge Handling Policy	Manual
Notify Method	[]
Notify Interval (seconds)	[]
Maximum Notifications	[]
Split and Merge Action Plan	Reboot

When selecting Manual as the split handling policy, the merge handling policy also is Manual. This setting is required and cannot be changed.

There are other options that can be changed. Table 10-3 shows the context-sensitive help for these items. This scenario keeps the default values.

Table 10-3 Information table to help explain the split handling policy

Name	Context-sensitive help (F1)	Associated list (F4)
Notify Method	A method to be invoked in addition to a message to /dev/console to inform the operator of the need to chose which site continues 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 merge to indicate the event.	None.
Notify Interval (seconds)	The frequency of the notification (time, in seconds, between messages) to inform the operator of the need to chose which site continues after a split or merge.	10..3600 Default is 30s, and then increases in frequency.
Maximum Notifications	The maximum number of times that PowerHA SystemMirror prompts the operator to chose which site continues after a split or merge.	3..1000 Default is infinite.

Name	Context-sensitive help (F1)	Associated list (F4)
Split and Merge Action Plan	<ol style="list-style-type: none"> 1. Reboot: Nodes on the loosing partition restart. 2. Disable Applications Auto-Start and Reboot: Nodes on the loosing partition restart. The RGs cannot be brought online until the merge finishes. 3. Disable Cluster Services Auto-Start and Reboot: Nodes on the loosing partition restart. CAA does not start. After the split condition is healed, you must run clenablepostsplit to bring the cluster back to a stable state. 	<ol style="list-style-type: none"> 1. Reboot. 2. Disable Applications Auto-Start and Reboot. 3. Disable Cluster Services Auto-Start and Reboot.

Example 10-58 shows the summary after confirming the manual policy configuration.

Example 10-58 Manual split merge configuration summary

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

The PowerHA SystemMirror split and merge policies have been updated.

Current policies are:

```

Split Handling Policy :                Manual
Merge Handling Policy :                Manual
Notify Method :
Notify Interval (seconds) :
Maximum Notifications :
Split and Merge Action Plan :        Reboot

```

The configuration must be synchronized to make this change known across the cluster.

The PowerHA **clmgr** command provides an option to display the cluster split and merge policy, as shown in Example 10-59.

Example 10-59 The clmgr output of split merge policies enabled

```

# clmgr view cluster SPLIT-MERGE
SPLIT_POLICY="manual"
MERGE_POLICY="manual"
ACTION_PLAN="reboot"
TIEBREAKER=""
NOTIFY_METHOD=""
NOTIFY_INTERVAL=""
MAXIMUM_NOTIFICATIONS=""
DEFAULT_SURVIVING_SITE=""
APPLY_TO_PPRC_TAKEOVER="n"

```

Synchronize the cluster. After the synchronization operation completes, the cluster can be activated.

10.10.3 Cluster split

Before simulating a cluster split, check its status, as described in 10.6.3, “Initial PowerHA service status for each scenario” on page 351.

In this case, we broke all communication between both nodes at 21:43:33.

Result and log on the PHA170 node

The following events occur:

- ▶ 21:43:33: All communication between the two nodes is broken.
- ▶ 21:43:43: The PHA170 node CAA marks ADAPTER_DOWN for the PHA171 node.
- ▶ 21:44:13: The PHA170 node CAA marks NODE_DOWN for the PHA171 node.
- ▶ 21:44:13: The PowerHA triggers a split_merge_prompt split event.

Then, every console on the PHA170 node receives the message that is shown in Example 10-60.

Example 10-60 Manual split console confirmation message on the PHA170

Broadcast message from root@PHA170 (tty) at 21:44:14 ...

A cluster split has been detected.

You must decide if this side of the partitioned cluster is to continue.
To have it continue, enter

/usr/es/sbin/cluster/utilities/cl_sm_continue

To have the recovery action - Reboot - taken on all nodes on this partition, enter

/usr/es/sbin/cluster/utilities/cl_sm_recover
LOCAL_PARTITION 1 PHA170 OTHER_PARTITION 2 PHA171

Also, in the hacmp.out log of the PHA170 node, there is a notification that is logged about a prompt for a split notification, as shown in Example 10-61.

Example 10-61 The hacmp.out log shows a split notification

```
Fri Dec 2 21:44:13 CST 2016 cl_sm_prompt (19136930): EVENT START: split_merge_prompt split LOCAL_PARTITION 1 PHA170 OTHER_PARTITION 2 PHA171 1
Fri Dec 2 21:44:14 CST 2016 cl_sm_prompt (19136930): split = Manual merge = Manual which = split split = Manual merge = Manual which = split
Fri Dec 2 21:44:14 CST 2016 cl_sm_prompt (19136930): Received a split notification for which a manual response is required.
Fri Dec 2 21:44:14 CST 2016 cl_sm_prompt (19136930): In manual for a split notification with Reboot
```

Result and log on the PHA171 node

The following events occur:

- ▶ 21:43:33: All communication between the two nodes is broken.
- ▶ 21:43:43: The PHA171 node CAA marks ADAPTER_DOWN for the PHA170 node.
- ▶ 21:44:13: The PHA171 node CAA marks NODE_DOWN for the PHA170 node.
- ▶ 21:44:13: PowerHA triggers the split_merge_prompt split event.

Every console of the PHA170 node also receives a message, as shown in Example 10-62.

Example 10-62 Manual split console confirmation message on PHA171

Broadcast message from root@PHA171 (tty) at 21:44:13 ...

A cluster split has been detected.

You must decide if this side of the partitioned cluster is to continue.
To have it continue, enter

```
/usr/es/sbin/cluster/utilities/cl_sm_continue
```

To have the recovery action - Reboot - taken on all nodes on this partition, enter

```
/usr/es/sbin/cluster/utilities/cl_sm_recover  
LOCAL_PARTITION 2 PHA171 OTHER_PARTITION 1 PHA170
```

Note: When the **cl_sm_continue** command is run on one node, this node continues to survive and takes over the RG if needed. Typically, this command is run on only one of the nodes.

When the **cl_sm_recover** command is run on one node, this node restarts. Typically, you do not want to run this command on both nodes.

This scenario runs the **cl_sm_recover** command on the PHA170 node, as shown in Example 10-63. We also run the **cl_sm_continue** command on the PHA171 node.

Example 10-63 Running cl_sm recover on PHA170

```
# date  
Fri Dec 2 21:44:25 CST 2016  
/usr/es/sbin/cluster/utilities/cl_sm_recover  
Resource Class Action Response for Resolve0pQuorumTie
```

Example 10-64 is the output of the **errpt -c** command. The PHA170 node restarts after running the **cl_sm_recover** command.

Example 10-64 The errpt output from the PHA170 post manual split

```
errpt -c  
4D91E3EA 1202214416 P S cluster0      A split has been detected.  
2B138850 1202214416 I O ConfigRM     ConfigRM received Subcluster Split event  
A098BF90 1202214416 P S ConfigRM     The operational quorum state of the acti  
<...>  
B80732E3 1202214416 P S ConfigRM     The operating system is being rebooted t  
<...>  
9DBCFCDEE 1202214616 T O errdemon     ERROR LOGGING TURNED ON  
69350832 1202214516 T S SYSPROC      SYSTEM SHUTDOWN BY USER  
<...>
```

The ConfigRM service log that is shown in Example 10-65 indicates that this node restarts at 21:44:48.

Example 10-65 ConfigRM service log from PHA170

```
[32] 12/02/16 _CFD 21:44:48.386539 !!!!!!!  
PeerDomainRcp::haltOSExecute (method=1). !!!!!!!  
[28] 12/02/16 _CFD 21:44:48.386540 ConfigRMUtils::log_error() Entered  
[32] 12/02/16 _CFD 21:44:48.386911 logerr: In  
File=../../../../src/rsct/rm/ConfigRM/PeerDomain.C (Version=1.99.22.299  
Line=23992) :  
CONFIGRM_REBOOTOS_ER
```

The operating system is being rebooted to ensure that critical resources are stopped so that another sub-domain that has operational quorum may recover these resources without causing corruption or conflict.

Note: To generate the IBM.ConfigRM service logs, run the following commands:

```
# cd /var/ct/IW/log/mc/IBM.ConfigRM  
# rpttr -o dct trace.* > ConfigRM.out
```

Then, check the ConfigRM.out file to get the relevant logs.

After the PHA170 node restarts, run the **c1_sm_continue** command operation on the PHA171 node, as shown in Example 10-66.

Example 10-66 The c1_sm_continue command on the PHA171 node

```
# date  
Fri Dec 2 21:45:08 CST 2016  
# /usr/es/sbin/cluster/utilities/c1_sm_continue  
Resource Class Action Response for Resolve0pQuorumTie
```

Then, the PHA171 node continues and proceeds to acquire the RG, as shown in the cluster.log file in Example 10-67.

Example 10-67 Cluster.log file from the PHA171 acquiring the resource group

```
Dec 2 21:45:26 PHA171 local0:crit clstrmgrES[10027332]: Fri Dec 2 21:45:26 Removing 1 from ml_idx  
Dec 2 21:45:26 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: split_merge_prompt quorum  
YES@SEQ@1450QRMNT@9@DE@11@NSEQ@8@OLD@1@NEW@0  
Dec 2 21:45:26 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: split_merge_prompt quorum  
YES@SEQ@1450QRMNT@9@DE@11@NSEQ@8@OLD@1@NEW@0  
0 0  
Dec 2 21:45:27 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: node_down PHA170  
Dec 2 21:45:27 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: node_down PHA170 0  
Dec 2 21:45:27 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move_release PHA171 1  
Dec 2 21:45:27 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move PHA171 1 RELEASE  
Dec 2 21:45:27 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move PHA171 1 RELEASE 0  
Dec 2 21:45:27 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move_release PHA171 1 0  
Dec 2 21:45:28 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move_fence PHA171 1  
Dec 2 21:45:28 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move_fence PHA171 1 0  
Dec 2 21:45:30 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move_fence PHA171 1  
Dec 2 21:45:30 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move_fence PHA171 1 0  
Dec 2 21:45:30 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move_acquire PHA171 1  
Dec 2 21:45:30 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move PHA171 1 ACQUIRE  
Dec 2 21:45:30 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: acquire_takeover_addr  
Dec 2 21:45:31 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: acquire_takeover_addr 0  
Dec 2 21:45:33 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move PHA171 1 ACQUIRE 0  
Dec 2 21:45:33 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move_acquire PHA171 1 0  
Dec 2 21:45:33 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: rg_move_complete PHA171 1  
Dec 2 21:45:34 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: rg_move_complete PHA171 1 0  
Dec 2 21:45:36 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT START: node_down_complete PHA170  
Dec 2 21:45:36 PHA171 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: node_down_complete PHA170 0
```

10.10.4 Cluster merge

In this case, the PHA170 restarts. After this restart operation completes, and when the heartbeat channel is restored, then you can merge this PowerHA cluster.

The steps are similar to the one that are described in 10.8.5, “Cluster merge” on page 370.

10.10.5 Scenario summary

If you want to decide when a cluster split occurs, then use the Manual policy for split and merge.

10.11 Scenario: Active node halt policy quarantine

This section presents a scenario for an ANHP quarantine.

10.11.1 Scenario description

Figure 10-39 shows the topology of this scenario.

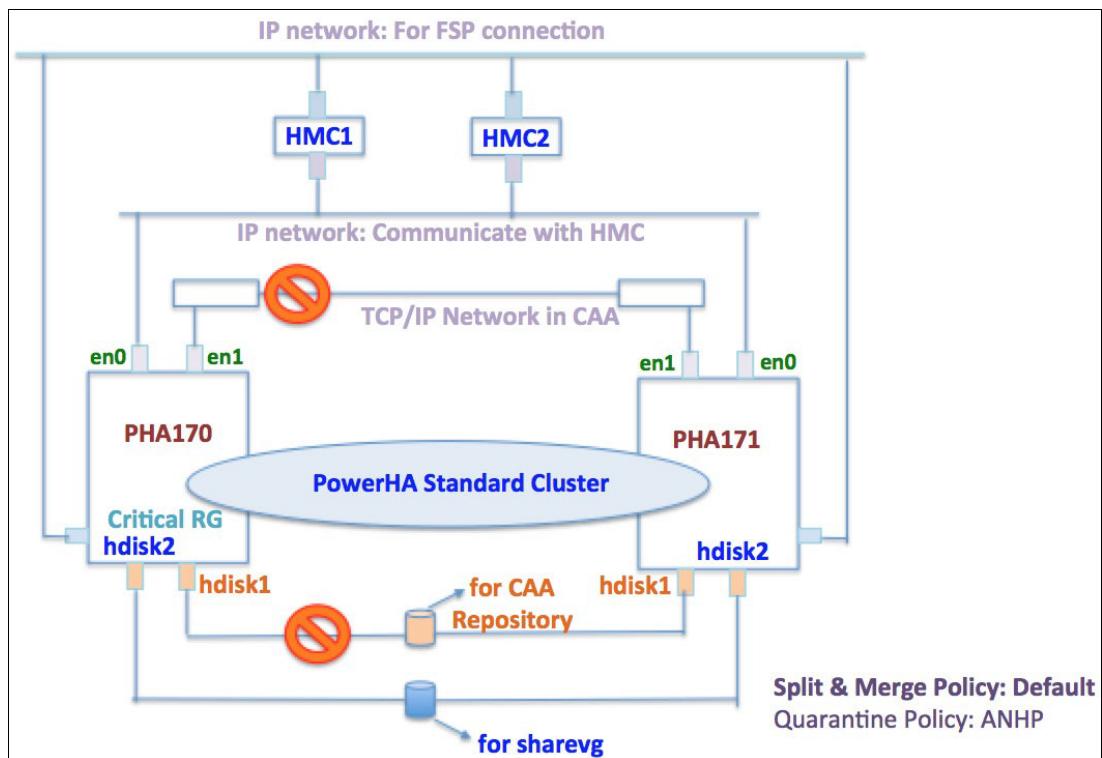


Figure 10-39 Active node halt policy quarantine

There are two HMCs in this scenario. Each HMC has two network interfaces: One is used to connect to the server's FSP adapter, and the other one is used to communicate with the PowerHA nodes. In this scenario, one node tries to shut down another node through the HMC by using the ssh protocol.

The two HMCs provide high availability functions. If one HMC fails, PowerHA uses another HMC to continue operations.

10.11.2 HMC password-less access configuration

Add the HMCs host names and their IP addresses into the /etc/hosts file on the PowerHA nodes:

```
172.16.15.55      HMC55
172.16.15.239     HMC239
```

Example 10-68 shows how to set up the HMC password-less access from the PHA170 node to one HMC.

Example 10-68 The ssh password-less setup of HMC55

```
# ssh-keygen -t rsa
Generating public/private rsa key pair.
Enter file in which to save the key (//.ssh/id_rsa):
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in //.ssh/id_rsa.
Your public key has been saved in //.ssh/id_rsa.pub.
The key fingerprint is:
64:f0:68:a0:9e:51:11:dc:e6:c5:fc:bf:74:36:72:cb root@PHA170
The key's randomart image is:
+--[ RSA 2048]----+
| .+=.o
| o..o++
| o  oo.+
| . o ..o .
| o      S .
|       + =
|       . B o
|       . E
+
# KEY=`cat ~/.ssh/id_rsa.pub` && ssh hscroot@HMC55 mkauthkeys -a \"$KEY\
Warning: Permanently added 'HMC55' (ECDSA) to the list of known hosts.
hscroot@HMC55's password: -> enter the password here

-> check if it is ok to access this HMC without password
# ssh hscroot@HMC55 lshmc -V
"version= Version: 8
 Release: 8.4.0
 Service Pack: 2
HMC Build level 20160816.1
", "base_version=V8R8.4.0
"
```

Example 10-69 shows how to set up HMC password-less access from the PHA170 node to another HMC.

Example 10-69 The ssh password-less setup of HMC239

```
# KEY=`cat ~/.ssh/id_rsa.pub` && ssh hscroot@HMC239 mkauthkeys -a \"$KEY\
Warning: Permanently added 'HMC239' (ECDSA) to the list of known hosts.
hscroot@HMC239's password: -> enter password here
```

```
(0) root @ PHA170: /ssh  
# ssh hscroot@HMC239 lshmc -V  
"version= Version: 8  
Release: 8.4.0  
Service Pack: 2  
HMC Build level 20160816.1  
","base_version=V8R8.4.0
```

Note: The operation that is shown in Example 10-69 on page 387 is also repeated for the PHA171 node.

10.11.3 HMC configuration in PowerHA

Complete the following steps:

1. The SMIT fast path is **smitty cm_cluster_quarantine_halt**. The full path is to run **smitty sysmirror** and then select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy** → **Quarantine Policy** → **Active Node Halt Policy**.

We choose the HMC Configuration, as shown in Example 10-70.

Example 10-70 Active node halt policy HMC configuration

Active Node Halt Policy

Move cursor to desired item and press Enter.

HMC Configuration
Configure Active Node Halt Policy

2. Select Add HMC Definition, as shown in Example 10-71 and press Enter. Then, the detailed definition menu opens, as shown in Example 10-72 on page 389.

Example 10-71 Adding an HMC

HMC Configuration

Move cursor to desired item and press Enter.

Add HMC Definition
Change/Show HMC Definition
Remove HMC Definition

Change/Show HMC List for a Node
Change/Show HMC List for a Site

Change/Show Default HMC Tunables
Change/Show Default HMC List

Example 10-72 HMC55 definition

Add HMC Definition

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

[Entry Fields]	
* HMC name	[HMC55]
DLPAR operations timeout (in minutes)	<input type="text"/>
Number of retries	<input type="text"/>
Delay between retries (in seconds)	<input type="text"/>
Nodes	[PHA171 PHA170]
Sites	<input type="text"/>
Check connectivity between HMC and nodes	Yes

Table 10-4 shows the help and information list for adding the HMC definition.

Table 10-4 Context-sensitive help and associated list for adding an HMC definition

Name	Context-sensitive help (F1)	Associated list (F4)
HMC name	Enter the host name for the HMC. An IP address is also accepted here. IPv4 and IPv6 addresses are supported.	Yes (single-selection). Obtained by running the following command: <code>/usr/sbin/rsct/bin/rmcd omainstatus -s ctrmc -a IP</code>
DLPAR operations timeout (in minutes)	Enter a timeout in minutes for DLPAR commands that are run on an HMC (use the <code>-w</code> parameter). This <code>-w</code> parameter exists only on the <code>chhwres</code> command when allocating or releasing resources. It is adjusted according to the type of resources (for memory, 1 minute per gigabytes 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 -1 is displayed in this field, it indicates that the default value is used.	None. This parameter is not used in an ANHP scenario.
Number of retries	Enter a number of times one HMC command is retried before the HMC is considered as non-responding. 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 -1 is displayed in this field, it indicates that the default value is used.	None. The default value is 5.
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 Change/Show Default HMC Tunables panel. When -1 is displayed in this field, it indicates that the default value is used.	None. The default value is 10s.

3. Add the first HMC55 for the two PowerHA nodes and keep the default value for the other items. Upon pressing Enter, PowerHA checks whether the current node can access HMC55 without a password, as shown in Example 10-73.

Example 10-73 HMC connectivity verification

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

Checking HMC connectivity between "PHA171" node and "HMC55" HMC : success!
Checking HMC connectivity between "PHA170" node and "HMC55" HMC : success!

4. Then, add another HMC, HMC239, as shown in Example 10-74.

Example 10-74 HMC239 definition

Add HMC Definition

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

[Entry Fields]	
*	HMC name [HMC239]
DLPAR operations timeout (in minutes)	<input type="text"/>
Number of retries	<input type="text"/>
Delay between retries (in seconds)	<input type="text"/>
Nodes	[PHA171 PHA170]
Sites	<input type="text"/>
Check connectivity between HMC and nodes	Yes

You can use the **clmgrn** commands to show the current setting of the HMC, as shown in Example 10-75.

Example 10-75 The clmgrn command displaying the HMC configurations

```
(0) root @ PHA170: /  
# clmgr query hmc -v  
NAME="HMC55"  
TIMEOUT="-1" -> '-1' means use default value  
RETRY_COUNT="-1" -> '-1' means use default value  
RETRY_DELAY="-1" -> '-1' means use default value  
NODES="PHA171 PHA170"  
STATUS="UP"  
VERSION="V8R8.4.0.2"  
  
NAME="HMC239"  
TIMEOUT="-1"  
RETRY_COUNT="-1"  
RETRY_DELAY="-1"  
NODES="PHA171 PHA170"  
STATUS="UP"  
VERSION="V8R8.6.0.0"
```

```
(0) root @ PHA170: /  
# clmgr query cluster hmc  
DEFAULT_HMC_TIMEOUT="10"  
DEFAULT_HMC_RETRY_COUNT="5"  
DEFAULT_HMC_RETRY_DELAY="10"  
DEFAULT_HMCS_LIST="HMC55 HMC239"
```

10.11.4 Quarantine policy configuration in PowerHA

Complete the following steps:

1. The SMIT fast path is **smitty cm_cluster_quarantine_halt**. The full path is to run **smitty sysmirror** and then select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy** → **Quarantine Policy** → **Quarantine Policy**.

The panel that is shown in Example 10-76 opens. Select the Configure Active Node Halt Policy.

Example 10-76 Configuring the active node halt policy

Active Node Halt Policy

Move cursor to desired item and press Enter.

HMC Configuration
Configure Active Node Halt Policy

2. The window in Example 10-77 is shown. Enable the Active Node Halt Policy and set the RG testRG as the Critical Resource Group.

Example 10-77 Enabling the active node halt policy and setting and critical resource group

Active Node Halt Policy

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

[Entry Fields]
* Active Node Halt Policy
* Critical Resource Group Yes +
 [testRG] +

In this scenario, there is only one RG, so we set it as the critical RG. For a description about the critical RG, see 10.3.1, “Active node halt quarantine policy” on page 332.

Example 10-78 shows the summary after pressing Enter.

Example 10-78 Cluster status summary

COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

The PowerHA SystemMirror split and merge policies have been updated.
Current policies are:

Split Handling Policy :	None
Merge Handling Policy :	Majority
Split and Merge Action Plan :	Reboot
The configuration must be synchronized to make this change known across the cluster.	
Active Node Halt Policy :	Yes
Critical Resource Group :	testRG

Note: If the split and merge policy is tiebreaker or manual, then the ANHP policy does not take effect. Make sure to set the Split Handling Policy to None before setting the ANHP policy.

3. Use the **clmgr** command to check the current configuration, as shown in Example 10-79.

Example 10-79 Checking the current cluster configuration

```
# clmgr view cluster|egrep -i "quarantine|critical"
QUARANTINE_POLICY="halt"
CRITICAL_RG="testRG"

# clmgr q cluster SPLIT-MERGE
SPLIT_POLICY="none"
MERGE_POLICY="majority"
ACTION_PLAN="reboot"
```

4. When the HMC and ANHP configuration is complete, verify and synchronize the cluster.

During the verification and synchronization process, the LPAR name and system information of the PowerHA nodes are added into the HACMPdynresop ODM database. They are used when ANHP is triggered, as shown in Example 10-80.

Example 10-80 Information that is stored in the HACMPdynresop

```
# odmget HACMPdynresop

HACMPdynresop:
    key = "PHA170_LPAR_NAME"
    value = "T_PHA170" -> LPAR name can be different with hostname,
    hostname is PHA170

HACMPdynresop:
    key = "PHA170_MANAGED_SYSTEM"
    value = "8284-22A*844B4EW" -> This value is System Model * Machine
    Serial Number

HACMPdynresop:
    key = "PHA171_LPAR_NAME"
    value = "T_PHA171"

HACMPdynresop:
    key = "PHA171_MANAGED_SYSTEM"
    value = "8408-E8E*842342W"
```

Note: You can obtain the LPAR name from AIX by running either `uname -L` or `lparstat -i`.

The requirements are as follows:

- ▶ Hardware firmware level 840 or later
- ▶ AIX 7.1 TL4 or 7.2 or later
- ▶ HMC V8 R8.4.0 (PTF MH01559) with a mandatory interim fix (PTF MH01560)

Here is an example output:

```
(0) root @ PHA170: /  
# hostname  
PHA170  
# uname -L  
5 T_PHA170
```

10.11.5 Simulating a cluster split

Before simulating a cluster split, check the cluster's status, as described in 10.6.3, "Initial PowerHA service status for each scenario" on page 351.

This scenario sets the Split Handling Policy to None and sets the Quarantine Policy to ANHP. The Critical Resource Group is testRG and is online on the PHA170 node at this time. When the cluster split occurs, it is expected that a backup node of this RG (PHA171) takes over the RG. During this process, PowerHA tries to shut down the PHA170 node through the HMC.

In this scenario, we broke all communication between two nodes at 02:44:04.

The main steps of CAA and PowerHA on the PHA171 node

The following events occur:

- ▶ 02:44:04: All communication between the two nodes is broken.
- ▶ 02:44:17: The PHA171 node CAA marks ADAPTER_DOWN for the PHA170 node.
- ▶ 02:44:47: The PHA171 node CAA marks NODE_DOWN for the PHA170 node.
- ▶ 02:44:47: PowerHA triggers the `split_merge_prompt` split event.
- ▶ 02:44:52: PowerHA triggers the `split_merge_prompt` quorum event, and then PHA171 takes over the RG.
- ▶ 02:44:55: In the `rg_move_acquire` event, PowerHA shuts down PHA170 through the HMC.
- ▶ 02:46:35: The PHA171 node completes the RG takeover.

The main steps of CAA and PowerHA on PHA170 node

The following events occur:

- ▶ 02:44:04: All communication between the two nodes is broken.
- ▶ 02:44:17: The PHA170 node marks REP_DOWN for the repository disk.
- ▶ 02:44:17: The PHA170 node CAA marks ADAPTER_DOWN for the PHA171 node.
- ▶ 02:44:47: The PHA170 node CAA marks NODE_DOWN for the PHA171 node.
- ▶ 02:44:47: PowerHA triggers a `split_merge_prompt` split event.
- ▶ 02:44:52: PowerHA triggers a `split_merge_prompt` quorum event.
- ▶ 02:44:55: The PHA170 node halts.

Example 10-81 shows the PowerHA cluster.log file of the PHA171 node.

Example 10-81 PHA171 node cluster.log file information

```
Dec 3 02:44:47 PHA171 EVENT START: split_merge_prompt split
Dec 3 02:44:47 PHA171 EVENT COMPLETED: split_merge_prompt split
Dec 3 02:44:52 PHA171 local0:crit clstrmgrES[7471396]: Sat Dec 3 02:44:52
Removing 1 from ml_idx
Dec 3 02:44:52 PHA171 EVENT START: split_merge_prompt quorum
Dec 3 02:44:52 PHA171 EVENT COMPLETED: split_merge_prompt quorum
Dec 3 02:44:52 PHA171 EVENT START: node_down PHA170
Dec 3 02:44:52 PHA171 EVENT COMPLETED: node_down PHA170 0
Dec 3 02:44:52 PHA171 EVENT START: rg_move_release PHA171 1
Dec 3 02:44:53 PHA171 EVENT START: rg_move PHA171 1 RELEASE
Dec 3 02:44:53 PHA171 EVENT COMPLETED: rg_move PHA171 1 RELEASE 0
Dec 3 02:44:53 PHA171 EVENT COMPLETED: rg_move_release PHA171 1 0
Dec 3 02:44:53 PHA171 EVENT START: rg_move_fence PHA171 1
Dec 3 02:44:53 PHA171 EVENT COMPLETED: rg_move_fence PHA171 1 0
Dec 3 02:44:55 PHA171 EVENT START: rg_move_fence PHA171 1
Dec 3 02:44:55 PHA171 EVENT COMPLETED: rg_move_fence PHA171 1 0
Dec 3 02:44:55 PHA171 EVENT START: rg_move_acquire PHA171 1
-> At 02:44:58, PowerHA triggered HMC to shutdown PHA170 node
Dec 3 02:46:28 PHA171 EVENT START: rg_move PHA171 1 ACQUIRE
Dec 3 02:46:28 PHA171 EVENT START: acquire_takeover_addr
Dec 3 02:46:29 PHA171 EVENT COMPLETED: acquire_takeover_addr 0
Dec 3 02:46:31 PHA171 EVENT COMPLETED: rg_move PHA171 1 ACQUIRE 0
Dec 3 02:46:31 PHA171 EVENT COMPLETED: rg_move_acquire PHA171 1 0
Dec 3 02:46:31 PHA171 EVENT START: rg_move_complete PHA171 1
Dec 3 02:46:33 PHA171 EVENT COMPLETED: rg_move_complete PHA171 1 0
Dec 3 02:46:35 PHA171 EVENT START: node_down_complete PHA170
Dec 3 02:46:35 PHA171 EVENT COMPLETED: node_down_complete PHA170 0
```

Example 10-82 shows the PowerHA hacmp.out file on the PHA171 node. The log indicates that PowerHA triggers a shutdown of the PHA170 node command at 02:44:55. This operation is in the PowerHA rg_move_acquire event.

Example 10-82 The PHA171 node hacmp.out file

```
Dec 3 2016 02:44:55 GMT -06:00 EVENT START: rg_move_acquire PHA171 1
<...>
:c1hmccmd[hmccmdexec:3707] : Start ssh command at Sat Dec 3 02:44:58 CST 2016
:c1hmccmd[hmccmdexec:1] ssh <...> hscroot@HMC55 'chsysstate -m
SVRP8-S822-08-SN844B4EW -r lpar -o shutdown --immed -n T_PHA170 2>&1
<...>
```

Note: PowerHA on the PHA171 node shuts down the PHA170 node before acquiring the service IP and varyonvg share VG. Only when this operation completes successfully does PowerHA continue other operations. If this operation fails, PowerHA is in the error state and does not continue. So, the data in the share VG is safe.

10.11.6 Cluster merge occurs

In this case, the PHA170 node halts after the cluster split occurs. When resolving cluster split issues, start PHA170 manually. After checking that the CAA service is up by running the **lsccluster -m** command, you can start the PowerHA service on the PHA170 node.

The steps are similar to what is described in 10.8.5, “Cluster merge” on page 370.

10.11.7 Scenario summary

Except for the cluster split and merge policies, PowerHA provides the ANHP quarantine policy to keep high availability and data safe in the case of a cluster split scenario. The policy also takes effect in case of a sick but not dead node. For more information, see 10.1.1, “Causes of a partitioned cluster” on page 321.

10.12 Scenario: Enabling the disk fencing quarantine policy

This section describes the scenario when disk fencing is enabled as the quarantine policy.

10.12.1 Scenario description

Figure 10-40 shows the topology of this scenario.

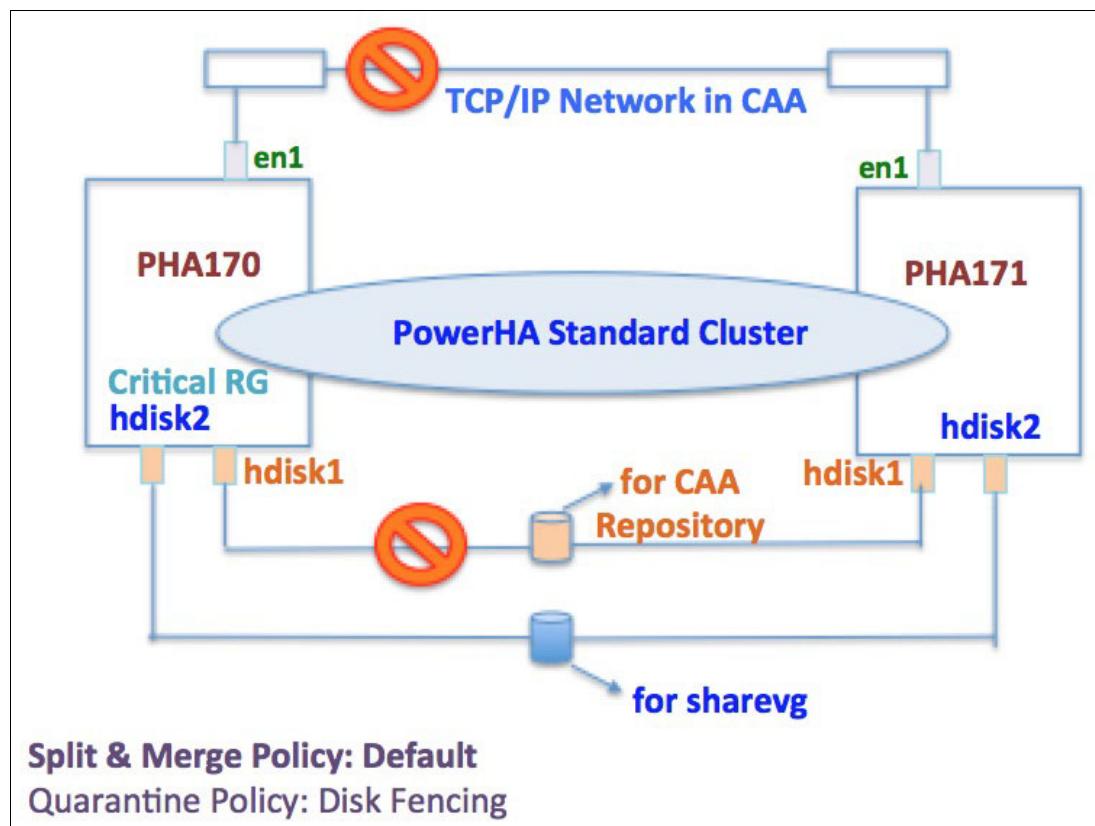


Figure 10-40 Topology scenario for the quarantine policy

In this scenario, the quarantine policy is disk fencing. There is one RG (testRG) in this cluster, so this RG is also marked as a Critical in Disk Fencing in the configuration.

There is one VG (sharevg) in this RG, and there is one hdisk in this VG. You must set the parameter `reserve_policy` to `no_reserve` for all the disks if you want to enable the disk fencing policy. In our case, hdisk2 is used, so you must run the following command on each PowerHA node:

```
chdev -l hdisk2 -a reserve_policy=no_reserve
```

10.12.2 Quarantine policy configuration in PowerHA

This section describes the quarantine policy configuration in a PowerHA cluster.

Ensuring that the active node halt policy is disabled

Note: If the ANHP policy is also enabled, in case of a cluster split, ANHP takes effect first.

Complete the following steps:

1. Use the SMIT fast path `smitty cm_cluster_quarantine_halt`, or run `smitty sysmirror` and then select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy** → **Quarantine Policy** → **Active Node Halt Policy**.
2. Example 10-83 shows the window. Select Configure Active Node Halt Policy.

Example 10-83 Configure the active node halt policy

Active Node Halt Policy

Move cursor to desired item and press Enter.

HMC Configuration
Configure Active Node Halt Policy

3. Example 10-84 shows where you can disable the ANHP.

Example 10-84 Disable the active node halt policy

Active Node Halt Policy

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

[Entry Fields]
* Active Node Halt Policy
* Critical Resource Group
No +
[testRG]

Enabling the disk fencing quarantine policy

Use the SMIT fast path `smitty cm_cluster_quarantine_disk_dialog`, or you can run `smitty sysmirror` and select **Custom Cluster Configuration** → **Cluster Nodes and Networks** → **Initial Cluster Setup (Custom)** → **Configure Cluster Split and Merge Policy** → **Split and Merge Management Policy** → **Quarantine Policy** → **Disk Fencing**.

Example 10-85 on page 397 shows that disk fencing is enabled and the Critical Resource Group is testRG.

Example 10-85 Disk fencing enabled and critical resource group selection

Disk Fencing

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

* Disk Fencing	[Entry Fields] Yes + [testRG]
* Critical Resource Group	

After pressing Enter, Example 10-86 shows the summary of the split and merge policy setting.

Example 10-86 Split and merge policy setting summary

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

The PowerHA SystemMirror split and merge policies have been updated.

Current policies are:

Split Handling Policy :	None
Merge Handling Policy :	Majority
Split and Merge Action Plan :	Reboot

The configuration must be synchronized to make this change known across the cluster.

Disk Fencing :	Yes
Critical Resource Group :	testRG

Note: If you want to enable only the disk fencing policy, you also must set the split handling policy to None.

Check the current settings

You can use the **clmgr** or the **odmget** command to check the current settings, as shown in Example 10-87 and Example 10-88.

Example 10-87 Checking the current cluster settings

```
# clmgr view cluster|egrep -i "quarantine|critical"
QUARANTINE_POLICY="fencing"
CRITICAL_RG="testRG"
```

Example 10-88 Checking the split and merge cluster settings

```
# odmget HACMPsplitmerge
```

```
HACMPsplitmerge:
    id = 0
    policy = "split"
    value = "None"
```

```
HACMPsplitmerge:
    id = 0
    policy = "merge"
    value = "Majority"
```

```

HACMPsplitmerge:
    id = 0
    policy = "action"
    value = "Reboot"

HACMPsplitmerge:
    id = 0
    policy = "anhp"
    value = "No" --> Important, make sure ANHP is disable.

HACMPsplitmerge:
    id = 0
    policy = "critical_rg"
    value = "testRG"

HACMPsplitmerge:
    id = 0
    policy = "scsi"
    value = "Yes"

```

Performing a PowerHA cluster verification and synchronization

Note: Before you perform a cluster verification and synchronization, check whether the reserve_policy for the shared disks are set to no_reserve.

After the verification and synchronization, you can see that the reserve_policy of hdisk2 changed to PR_shared and also generated one PR_key_value on each node.

Example 10-89 shows the PR_key_value and reserve_policy setting in the PHA170 node.

Example 10-89 The PR_key_value and reserve_policy settings on PHA170 node

```

# hostname
PHA170

# lsattr -El hdisk2|egrep "PR|reserve_policy"
PR_key_value      0x10001472090686                                Persistant Reserve Key
Value      True+
reserve_policy   PR_shared                                         Reserve Policy
True+

# devrsrv -c query -l hdisk2
Device Reservation State Information
=====
Device Name          : hdisk2
Device Open On Current Host? : NO
ODM Reservation Policy : PR SHARED
ODM PR Key Value   : 4503687439910534
Device Reservation State : NO RESERVE
Registered PR Keys  : No Keys Registered
PR Capabilities Byte[2] : 0x11 CRH PTPL_C
PR Capabilities Byte[3] : 0x81 PTPL_A
PR Types Supported   : PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

-> [HEX]0x10001472090686 = [DEC]4503687439910534

```

Example 10-90 shows the PR_key_value and the reserve_policy setting on the PHA171 node.

Example 10-90 PR_key_value and reserve_policy settings on the PHA171 node

```
# hostname
PHA171

# lsattr -El hdisk2|egrep "PR|reserve_policy"
PR_key_value      0x20001472090686                                Persistant Reserve Key
Value      True+
reserve_policy   PR_shared                                     Reserve Policy
True+

# devrsrv -c query -l hdisk2
Device Reservation State Information
=====
Device Name          : hdisk2
Device Open On Current Host? : NO
ODM Reservation Policy : PR_SHARED
ODM PR Key Value    : 9007287067281030
Device Reservation State : NO RESERVE
Registered PR Keys   : No Keys Registered
PR Capabilities Byte[2] : 0x11 CRH  PTPL_C
PR Capabilities Byte[3] : 0x81 PTPL_A
PR Types Supported    : PR_WE_AR  PR_EA_RO  PR_WE_RO  PR_EA  PR_WE  PR_EA_AR

-> [HEX]0x20001472090686 = [DEC]9007287067281030
```

10.12.3 Simulating a cluster split

Before simulating a cluster split, check the cluster's status, as described in 10.6.3, “Initial PowerHA service status for each scenario” on page 351.

This scenario sets the split handling policy to None and sets the quarantine policy to disk fencing. The Critical Resource Group is testRG and is online on the PHA170 node at this time. When the cluster split occurs, it is expected that the backup node of this RG (PHA171) takes over the RG. During this process, PowerHA on the PHA171 node fences out PHA170 node from accessing the disk and allows itself to access it. PowerHA tries to use this method to keep the data safe.

In this case, we broke all communication between two nodes at 04:14:12.

Main steps of CAA and PowerHA on the PHA171 node

The following events occur:

- ▶ 04:14:12: All communication between the two nodes is broken.
- ▶ 04:14:24: The PHA171 node CAA marks ADAPTER_DOWN for the PHA170 node.
- ▶ 04:14:54: The PHA171 node CAA marks NODE_DOWN for the PHA170 node.
- ▶ 04:14:54" PowerHA triggers a split_merge_prompt split event.
- ▶ 04:15:04: PowerHA triggers a split_merge_prompt quorum event, and then PHA171 took over the RG.
- ▶ 04:15:07: In the rg_move_acquire event, PowerHA preempts the PHA170 node from Volume Group sharevg.
- ▶ 04:15:14: The PHA171 node completes the RG takeover.

Example 10-91 shows the output of the PowerHA cluster.log file.

Example 10-91 PowerHA cluster.log output

```
Dec 3 04:14:54 PHA171 EVENT START: split_merge_prompt split
Dec 3 04:15:04 PHA171 EVENT COMPLETED: split_merge_prompt split
Dec 3 04:15:04 PHA171 local0:crit clstrmgrES[19530020]: Sat Dec 3 04:15:04 Removing 1
from ml_idx
Dec 3 04:15:04 PHA171 EVENT START: split_merge_prompt quorum
Dec 3 04:15:04 PHA171 EVENT COMPLETED: split_merge_prompt quorum
Dec 3 04:15:04 PHA171 EVENT START: node_down PHA170
Dec 3 04:15:04 PHA171 EVENT COMPLETED: node_down PHA170 0
Dec 3 04:15:05 PHA171 EVENT START: rg_move_release PHA171 1
Dec 3 04:15:05 PHA171 EVENT START: rg_move PHA171 1 RELEASE
Dec 3 04:15:05 PHA171 EVENT COMPLETED: rg_move PHA171 1 RELEASE 0
Dec 3 04:15:05 PHA171 EVENT COMPLETED: rg_move_release PHA171 1 0
Dec 3 04:15:05 PHA171 EVENT START: rg_move_fence PHA171 1
Dec 3 04:15:05 PHA171 EVENT COMPLETED: rg_move_fence PHA171 1 0
Dec 3 04:15:07 PHA171 EVENT START: rg_move_fence PHA171 1
Dec 3 04:15:07 PHA171 EVENT COMPLETED: rg_move_fence PHA171 1 0
Dec 3 04:15:07 PHA171 EVENT START: rg_move_acquire PHA171 1
-> At 04:15:07, PowerHA preempted PHA170 node from Volume Group sharevg, and continue
Dec 3 04:15:08 PHA171 EVENT START: rg_move PHA171 1 ACQUIRE
Dec 3 04:15:08 PHA171 EVENT START: acquire_takeover_addr
Dec 3 04:15:08 PHA171 EVENT COMPLETED: acquire_takeover_addr 0
Dec 3 04:15:10 PHA171 EVENT COMPLETED: rg_move PHA171 1 ACQUIRE 0
Dec 3 04:15:10 PHA171 EVENT COMPLETED: rg_move_acquire PHA171 1 0
Dec 3 04:15:10 PHA171 EVENT START: rg_move_complete PHA171 1
Dec 3 04:15:11 PHA171 EVENT COMPLETED: rg_move_complete PHA171 1 0
Dec 3 04:15:13 PHA171 EVENT START: node_down_complete PHA170
Dec 3 04:15:14 PHA171 EVENT COMPLETED: node down complete PHA170 0
```

Example 10-92 shows the output of the PowerHA hacmp.out file. It indicates that PowerHA triggers the preempt operation in the cl_scsipr preempt script.

Example 10-92 PowerHA hacmp.out file output

```
Dec  3 2016 04:15:07 GMT -06:00 EVENT START: rg_move_acquire PHA171 1
...
:cl_scsipr_preempt[85] PR_Key=0x10001472090686
:cl_scsipr_preempt[106] : Node PHA170 is down, preempt PHA170 from the Volume Groups,
:cl_scsipr_preempt[107] : which are part of any Resource Group.
:cl_scsipr_preempt[109] odmget HACMPgroup
:cl_scsipr_preempt[109] sed -n '$/group =/{ s/.*\\"(.*)\\\"/\\"1/; h; }\\n\\t\\t\\t\\t\\t/nodes =/{ /[
"]PHA170[ "]/{ g; p; }\\n\\t\\t\\t\\t\\t}'"
:cl_scsipr_preempt[109] ResGrps=testRG
:cl_scsipr_preempt[109] typeset ResGrps
:cl_scsipr_preempt[115] clodmget -n -q group='testRG' and name like '*VOLUME_GROUP' -f value
HACMPresource
:cl_scsipr_preempt[115] VolGrps=sharevg
:cl_scsipr_preempt[115] typeset VolGrps
:cl_scsipr_preempt[118] clpr_ReadRes_vg sharevg
Number of disks in VG sharevg: 1
hdisk2
:cl_scsipr_preempt[120] clpr_verifyKey_vg sharevg 0x20001472090686
Number of disks in VG sharevg: 1
hdisk2
:cl_scsipr_preempt[124] : Node PHA170 is down, preempting that node from Volume Group sharevg.
:cl_scsipr preempt[126] clpr preempt abort vg sharevg 0x10001472090686
```

```
Number of disks in VG sharevg: 1
```

```
hdisk2
```

```
...
```

Main steps of CAA and PowerHA on the PHA170 node

The following events occur:

- ▶ 04:14:12: All communication between the two nodes is broken.
- ▶ 04:14:21: The PHA171 node CAA marks ADAPTER_DOWN for the PHA170 node.
- ▶ 04:14:51: The PHA171 node CAA marks NODE_DOWN for the PHA170 node.
- ▶ 04:14:51: PowerHA triggers the split_merge_prompt split event.
- ▶ 04:14:56: Removing 2 from m1_idx.
- ▶ 04:14:56: PowerHA triggers a split_merge_prompt quorum event.
- ▶ 04:14:58: EVENT START: node_down PHA171.
- ▶ 04:14:58: EVENT COMPLETED: node_down PHA171.

No other events occur on the PHA170 node.

After some time, at 04:15:16, the /sharefs file system is fenced out and the application on the PHA170 node cannot perform an update operation to it, but the application can still perform read operations from it.

Example 10-93 shows the PowerHA cluster.log file of the PHA171 node.

Example 10-93 PowerHA cluster.log file of the PHA171 node

PHA170:		
4D91E3EA	1203041416 P S cluster0	A split has been detected.
2B138850	1203041416 I O ConfigRM	ConfigRM received Subcluster Split event
...		
A098BF90	1203041416 P S ConfigRM	The operational quorum state of the acti
4BDDFBCC	1203041416 I S ConfigRM	The operational quorum state of the acti
AB59ABFF	1203041416 U U LIBLVM	Remote node Concurrent Volume Group fail
AB59ABFF	1203041416 U U LIBLVM	Remote node Concurrent Volume Group fail
...		
65DE6DE3	1203041516 P S hdisk2	REQUESTED OPERATION CANNOT BE PERFORMED
E86653C3	1203041516 P H LVDD	I/O ERROR DETECTED BY LVM
EA88F829	1203041516 I O SYSJ2	USER DATA I/O ERROR
65DE6DE3	1203041516 P S hdisk2	REQUESTED OPERATION CANNOT BE PERFORMED
65DE6DE3	1203041516 P S hdisk2	REQUESTED OPERATION CANNOT BE PERFORMED
E86653C3	1203041516 P H LVDD	I/O ERROR DETECTED BY LVM
52715FA5	1203041516 U H LVDD	FAILED TO WRITE VOLUME GROUP STATUS AREA
F7DDA124	1203041516 U H LVDD	PHYSICAL VOLUME DECLARED MISSING
CAD234BE	1203041516 U H LVDD	QUORUM LOST, VOLUME GROUP CLOSING
E86653C3	1203041516 P H LVDD	I/O ERROR DETECTED BY LVM
52715FA5	1203041516 U H LVDD	FAILED TO WRITE VOLUME GROUP STATUS AREA
CAD234BE	1203041516 U H LVDD	QUORUM LOST, VOLUME GROUP CLOSING
65DE6DE3	1203041516 P S hdisk2	REQUESTED OPERATION CANNOT BE PERFORMED
65DE6DE3	1203041516 P S hdisk2	REQUESTED OPERATION CANNOT BE PERFORMED
E86653C3	1203041516 P H LVDD	I/O ERROR DETECTED BY LVM
E86653C3	1203041516 P H LVDD	I/O ERROR DETECTED BY LVM
78ABDDEB	1203041516 I O SYSJ2	META-DATA I/O ERROR
78ABDDEB	1203041516 I O SYSJ2	META-DATA I/O ERROR
65DE6DE3	1203041516 P S hdisk2	REQUESTED OPERATION CANNOT BE PERFORMED
E86653C3	1203041516 P H LVDD	I/O ERROR DETECTED BY LVM

C1348779	1203041516	I O	SYSJ2	LOG I/O ERROR
B6DB68E0	1203041516	I O	SYSJ2	FILE SYSTEM RECOVERY REQUIRED

Example 10-94 shows detailed information about event EA88F829.

Example 10-94 Showing event EA88F829

LABEL: J2_USERDATA_EIO
IDENTIFIER: EA88F829

Date/Time: Mon Dec 3 04:15:16 CST 2016
Sequence Number: 12629
Machine Id: 00FA4B4E4C00
Node Id: PHA170
Class: 0
Type: INFO
WPAR: Global
Resource Name: SYSJ2

Description
USER DATA I/O ERROR

Probable Causes

ADAPTER HARDWARE OR MICROCODE
DISK DRIVE HARDWARE OR MICROCODE
SOFTWARE DEVICE DRIVER
STORAGE CABLE LOOSE, DEFECTIVE, OR UNTERMINATED

Recommended Actions

CHECK CABLES AND THEIR CONNECTIONS
INSTALL LATEST ADAPTER AND DRIVE MICROCODE
INSTALL LATEST STORAGE DEVICE DRIVERS
IF PROBLEM PERSISTS, CONTACT APPROPRIATE SERVICE REPRESENTATIVE

Detail Data

JFS2 MAJOR/MINOR DEVICE NUMBER
0064 0001
FILE SYSTEM DEVICE AND MOUNT POINT
</dev/sharelv>, </sharefs>

Example 10-95 shows the output of the **devrsrv** command on the PHA170 node. It indicates that hdisk2 was held by the 9007287067281030 PR key, and this key belongs to the PHA171 node.

Example 10-95 The devrsrv command output of the PHA170 node

```
# hostname
PHA170

# devrsrv -c query -l hdisk2
Device Reservation State Information
=====
Device Name          : hdisk2
Device Open On Current Host?   : YES
ODM Reservation Policy      : PR SHARED
ODM PR Key Value        : 4503687439910534
Device Reservation State    : PR SHARED
```

PR Generation Value	:	34
PR Type	:	PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value	:	0
Registered PR Keys	:	9007287067281030 9007287067281030
PR Capabilities Byte[2]	:	0x11 CRH PTPL_C
PR Capabilities Byte[3]	:	0x81 PTPL_A
PR Types Supported	:	PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

Example 10-96 shows the output of the **devrsrv** command on the PHA171 node.

Example 10-96 The devrsrv command output of the PHA171 node

# hostname		
PHA170		
# devrsrv -c query -l hdisk2		
Device Reservation State Information		
=====		
Device Name	:	hdisk2
Device Open On Current Host?	:	YES
ODM Reservation Policy	:	PR SHARED
ODM PR Key Value	:	9007287067281030
Device Reservation State	:	PR SHARED
PR Generation Value	:	34
PR Type	:	PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)
PR Holder Key Value	:	0
Registered PR Keys	:	9007287067281030 9007287067281030
PR Capabilities Byte[2]	:	0x11 CRH PTPL_C
PR Capabilities Byte[3]	:	0x81 PTPL_A
PR Types Supported	:	PR_WE_AR PR_EA_RO PR_WE_RO PR_EA PR_WE PR_EA_AR

Note: From the above description, you can see that the PHA171 node takes over the RG and the data in /sharefs file system is safe, and the service IP is attached on PHA171 node too. But the service IP is also online in the PHA170 node. So there is a risk that there is an IP conflict. So, you need to do some manual operations to avoid this risk, including rebooting the PHA170 node manually.

10.12.4 Simulating a cluster merge

Restarting or shutting down the PHA170 node is one method to avoid a service IP conflict.

In this scenario, restart the PHA170 node and restore all communication between the two nodes. After checking that the CAA service is up by running the **1scluster -m** command, start the PowerHA service on the PHA170 node.

The steps are similar to 10.8.5, “Cluster merge” on page 370.

During the start of the PowerHA service, in the node_up event, PowerHA on the PHA170 node resets the reservation for the shared disks.

Example 10-97 shows the output of the PowerHA cluster.log file on the PHA170 node.

Example 10-97 PowerHA cluster.log file on the PHA170 node

```
Dec 3 04:41:05 PHA170 local0:crit clstrmgrES[10486088]: Sat Dec 3 04:41:05 HACMP: clstrmgrES: VRMF fix  
level in product ODM = 0  
Dec 3 04:41:05 PHA170 local0:crit clstrmgrES[10486088]: Sat Dec 3 04:41:05 CLSTR_JOIN_AUTO_START - This  
is the normal start request  
Dec 3 04:41:18 PHA170 user:notice PowerHA SystemMirror for AIX: EVENT START: node_up PHA170  
-> PowerHA reseted reservation for shared disks  
Dec 3 04:41:20 PHA170 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: node_up PHA170 0  
Dec 3 04:41:22 PHA170 user:notice PowerHA SystemMirror for AIX: EVENT START: node_up_complete PHA170  
Dec 3 04:41:22 PHA170 user:notice PowerHA SystemMirror for AIX: EVENT COMPLETED: node_up_complete PHA170 0
```

Example 10-98 shows the output of the node_up event in PHA170. The log indicates that PowerHA registers its key into the shared disks of the sharevg.

Example 10-98 The node_up event output of the PHA170 node

```
Dec 3 2016 04:41:18 GMT -06:00 EVENT START: node_up PHA170  
...  
:node_up[node_up_scsipr_init:122] clpr_reg_res_vg sharevg 0x10001472090686  
Number of disks in VG sharevg: 1  
hdisk2  
:node_up[node_up_scsipr_init:123] (( 0 != 0 ))  
:node_up[node_up_scsipr_init:139] : Checking if reservation succeeded  
:node_up[node_up_scsipr_init:141] clpr_verifyKey_vg sharevg 0x10001472090686  
Number of disks in VG sharevg: 1  
hdisk2  
:node_up[node_up_scsipr_init:142] RC1=0  
:node_up[node_up_scsipr_init:143] (( 0 == 1 ))  
:node_up[node_up_scsipr_init:149] (( 0 == 0 ))  
:node_up[node_up_scsipr_init:153] : Reservation success
```

Example 10-99 shows that the PR key value of PHA170 node is registered to hdisk2. Thus, it is ready for the next cluster split event.

Example 10-99 PHA170 PR key value

```
# hostname  
PHA171  
  
# devrsrv -c query -l hdisk2  
Device Reservation State Information  
=====  
Device Name : hdisk2  
Device Open On Current Host? : YES  
ODM Reservation Policy : PR SHARED  
ODM PR Key Value : 9007287067281030  
Device Reservation State : PR SHARED  
PR Generation Value : 38  
PR Type : PR_WE_AR (WRITE EXCLUSIVE, ALL REGISTRANTS)  
PR Holder Key Value : 0  
Registered PR Keys : 4503687439910534 9007287067281030  
9007287067281030 4503687439910534  
PR Capabilities Byte[2] : 0x11 CRH PTPL_C  
PR Capabilities Byte[3] : 0x81 PTPL_A
```

```
PR Types Supported      : PR_WE_AR  PR_EA_RO  PR_WE_RO  PR_EA  PR_WE
PR_EA_AR
Sat Dec  3 04:41:22 CST 2016
```

10.12.5 Scenario summary

Except for the cluster split and merge policies, PowerHA provides a disk fencing quarantine policy to keep high availability and data safe in case of cluster split scenarios. It also takes effect in the case of sick but not dead. For more information, see 10.1.1, “Causes of a partitioned cluster” on page 321.



IBM PowerHA SystemMirror special features

This chapter covers specific features that are new to IBM PowerHA SystemMirror for IBM AIX for Version 7.2 and Version 7.2.1.

This chapter covers the following topic:

- ▶ New option for starting PowerHA by using the clmgr command

11.1 New option for starting PowerHA by using the clmgr command

Starting with of PowerHA V7.2, there is an additional management option to start the cluster. The new argument for the option **manage** is named **delayed**.

Note: This new option is backlevel ported to PowerHA V7.2 and 7.1.3. At the time of writing, you can obtain it by opening a Problem Management Report (PMR) and asking for an interim fix for defect 1008628.

Here is the syntax of the new option:

```
clmgr online cluster manage=delayed
```

11.1.1 PowerHA Resource Group dependency settings

Starting with PowerHA V7.1.0, the *Start After* and the *Stop After* dependencies are added. The *Start After* is used often. The Resource Group (RG) dependencies support up to three levels. Figure 11-1 shows a proposed setup that cannot be configured. If you encounter such a case, you must find another solution. In this example, RG1 is at level 1, RG2 is at level 2, RG3 and RG4 are at level 3, and RG5 and RG6 are at level 4.

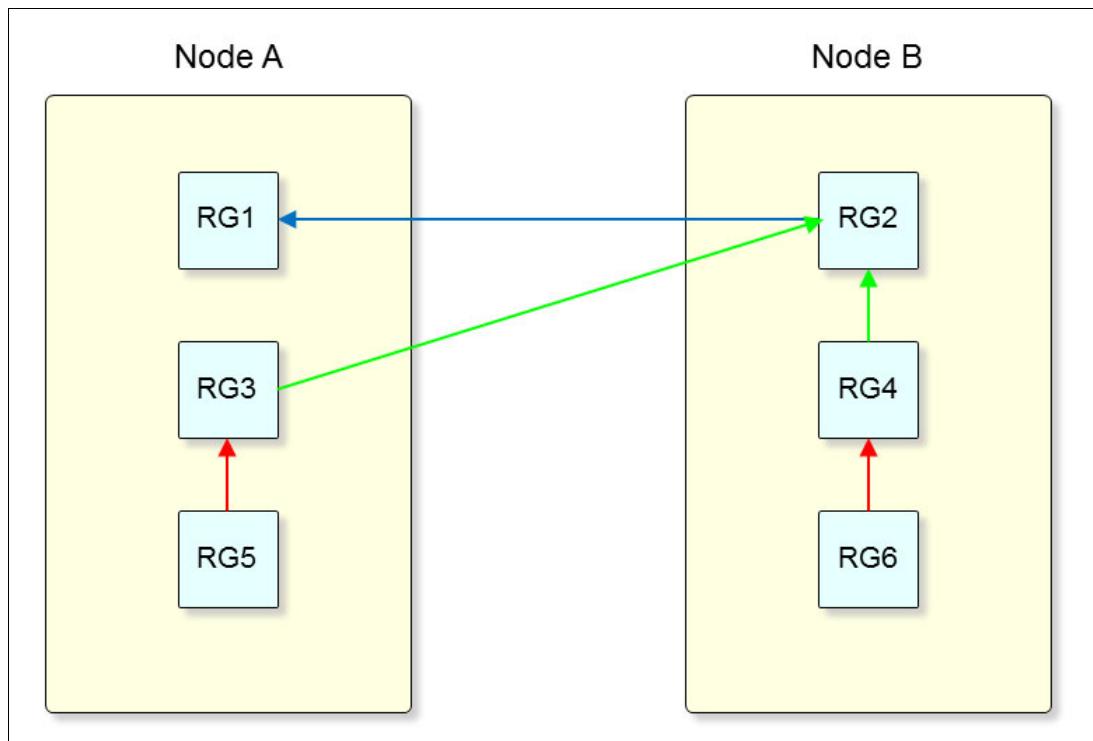


Figure 11-1 Start After with more than three levels

A supported setup example is shown in Figure 11-2. It has the same number of RGs, but it is using three dependency levels. The challenges that you can get with such request are described in 11.1.2, “Use case for using manage=delayed” on page 409.

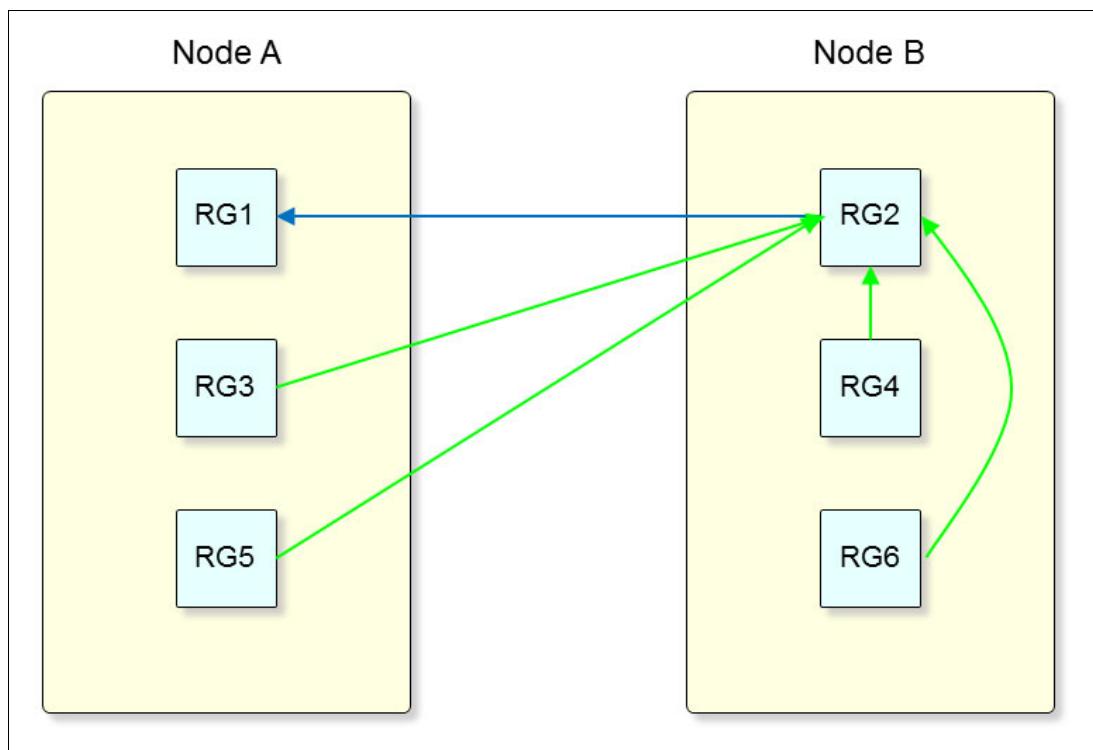


Figure 11-2 Start After with three levels

11.1.2 Use case for using manage=delayed

When you have multiple RGs with a similar dependency as the one that is shown in Figure 11-2, there is a challenge when the cluster is started. This section describes this behavior in more details.

Starting the entire cluster with the default settings

When starting the cluster in one go, for example, by using `clmgr on cluster`, then one of the two nodes always is the first one. In fact, it does not matter if you use `clmgr` or `smitty`. The sequence can be different, but that is the only difference.

The following section illustrates an example and does not describe the exact internal behavior. It is a description of what you can experience. Assume that Node A is the first one. Then, you get the startup situation that is shown illustrated in Figure 11-3:

1. (A) Cluster Manager on Node A is initialized.
2. (1) RG1 starts.
3. (2) The RG3 start fails due to the Start After dependency.
4. (3) The RG5 start fails due to the Start After dependency.
5. (B) The cluster manager on Node B is initialized
6. (4) RG2 starts.
7. (5) RG4 and RG6 start.
8. (6) RG3 and RG5 recover from the error status and start.

The startup takes longer than expected. Depending on your timeout settings, the start can have a significant time delay.

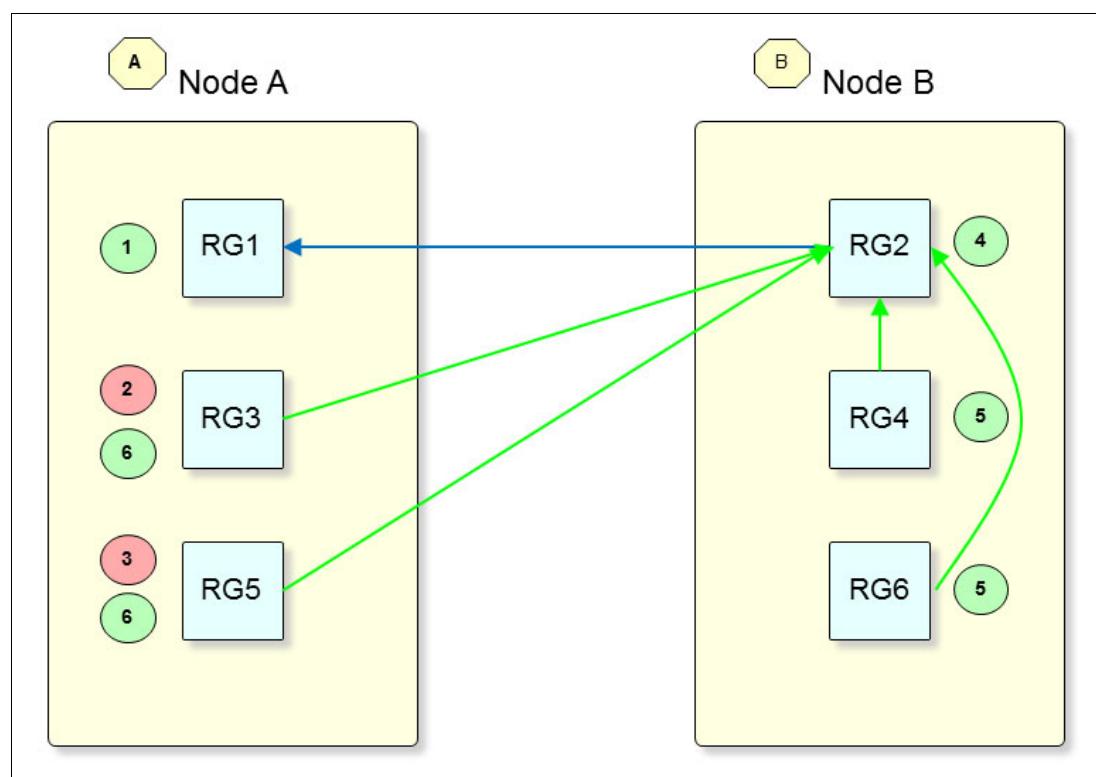


Figure 11-3 Start sequence (clmgr on cluster)

This behavior happens due to the fact that the first cluster manager does not see the partner at its initialization. This situation always happens even if there is a small time delay. Therefore, the second cluster manager must wait until all RGs are processed by the first cluster manager.

Starting the whole cluster with the first available node settings

A solution might be to change all the RGs to start on the *first available node*. The Start After settings are still the same as shown in Figure 11-2 on page 409. The assumption is that you still use the command `clmgr on cluster` command.

As shown in Figure 11-4, the start sequence is as defined in the Start After dependencies. But, now all RGs are running on Node A, which is not the outcome that you want.

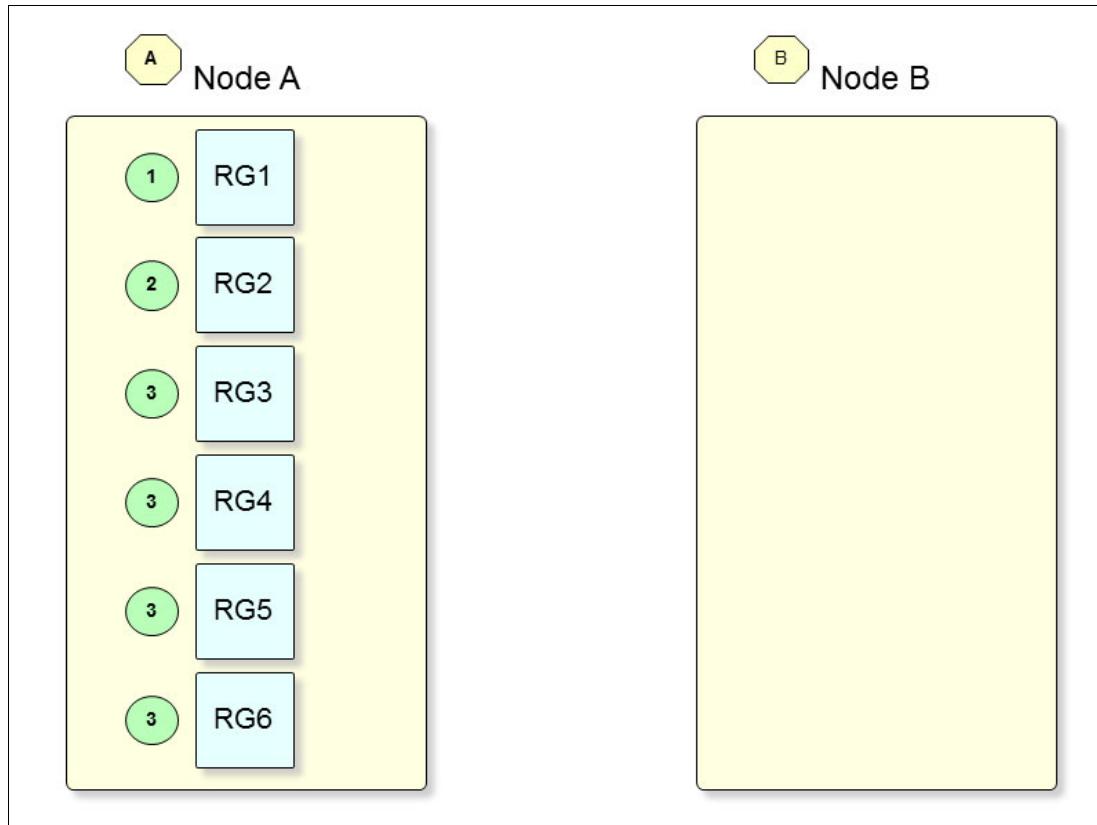


Figure 11-4 Start sequence with the setting on first available node

Starting the whole cluster with the manual option

In the past, the only way to correct this situation was to use the manual option. For the example here, start with the original settings that are described in “Starting the entire cluster with the default settings” on page 409. Run the `clmgr on cluster manage=manual`. This command starts both cluster managers but not the RG.

Now, the cluster managers are running in a stable state, as shown in Figure 11-5. You can start all your RGs by using the **c1mgr** command.

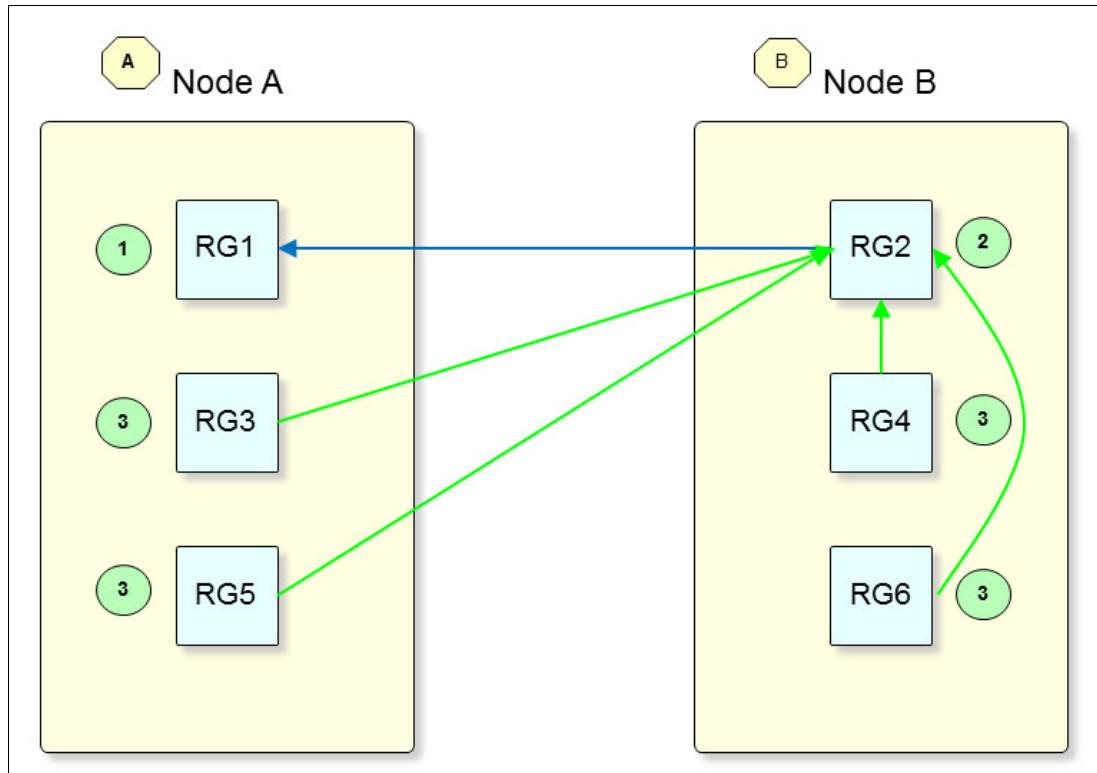


Figure 11-5 RG start sequence when all the cluster managers are running



A

SCSI reservations

This appendix describes SCSI reservation, and how it can be used to provide faster disk failover times when the underlying storage supports this feature. For example, SCSI 3 Persistent Reservation allows the stripe group manager, also known as file system manager, to “fence” disks during node failover by removing the reservation keys for that node. In contrast, non-PR disk failover causes the system to wait until the disk lease expires.

Attention: You do *not* run these commands in your systems. By running these commands, this section shows you how disk reservations work, especially in a clustered environment, which demands more care while managing disk reservations.

This appendix describes SCSI reservations. This appendix covers the following topics:

- ▶ SCSI reservations
- ▶ Persistent Reserve IN
- ▶ Storage
- ▶ More about PR reservations
- ▶ Persistent reservation commands

SCSI reservations

SCSI 2 reservations provide a mechanism to reserve and control access to a SCSI device from a node. An initiator obtains ownership of the device by using the *reserve* system call and works as a lock against any I/O attempt from other initiators. Another initiator trying to access this reserved disk gets a reservation conflict error code. Only the original initiator can release this reservation by issuing a **release** or **reset** system call.

SCSI 3 Persistent Reservations provide the mechanism to control access to a shared device from multiple nodes. The reservation persists even if the bus is reset for error recovery, which is not the case with the SCSI 2 command, where device reservations do not survive after a node restarts. Also, SCSI 3 PR supports multiple paths to a host, where SCSI 2 works only with one path from host to a disk. The scope of a persistent reservation is the entire logical unit.

SCSI 3 Persistent Reservations use the concept of *register* and *reserve*. Multiple nodes can register their reservation keys (also known as PR_Key) with the shared device and establish a reservation in any of the following modes, as shown in Table A-1.

Table A-1 Types of SCSI reservations

Types	Code
Write exclusive	1h
Exclusive access	3h
Write exclusive - Registrants only	5h
Exclusive Access - Registrants only	6h
Write Exclusive - All registrants	7h
Exclusive Access - All registrants	8h

In All Registrants type of reservations (WEAR and EAAR), each registered node is a Persistent Reservation (PR) Holder. The PR Holder value is set to zero. The All registrants type is an optimization that makes all cluster members equal, so if any member fails, the others continue.

In all other types of reservation, there is a single reservation holder, which is one of the following I_T nexus examples:

- ▶ The nexus for which the reservation was established with a **PERSISTENT RESERVE OUT** command with the **RESERVE** service action, the **PREEMPT** service action, the **PREEMPT AND ABORT** service action, or the **REPLACE LOST RESERVATION** service action.
- ▶ The nexus to which the reservation was moved by a **PERSISTENT RESERVE OUT** command with the **REGISTER AND MOVE** service action.

An I_T nexus refers to the combination of the initiator port on the host with the target port on the server:

- ▶ 1h Write Exclusive (WE)

Only the Persistent reservation holder shall be permitted to perform write operations to the device. Only one persistent reservation holder at a time.

- ▶ 3h Exclusive Access (EA)

Only the Persistent reservation holder shall be permitted to access (includes read/write operations) the device. Only one persistent reservation holder at a time.

- ▶ 5h Write Exclusive Registrants only (WERO)

Write access commands are permitted only to registered nodes. A cluster that is designed around this type must declare one cluster owner (the persistent reservation holder) at a time. If the owner fails, another must be elected. The PR_key_Holder value is pointing to the PR_Key of the I_T nexus that holds the reservation of the disk. Only one persistent reservation holder at a time, but all registered I_T nexuses are allowed to do write operations on the disk.

- ▶ 6h Exclusive Access Registrants only (EARO)

Access to the device is limited only to the registered nodes, and as in WERO, if the current owner fails, the reservation must be established again to gain access to the device. Only one persistent reservation holder at a time is permitted, but all registered I_T nexuses may do read/write operations on the disk.

- ▶ 7h Write exclusive All Registrants (WEAR)

While this reservation is active, only the registered initiators may perform write operations to the indicated extent. This reservation shall not inhibit read operations from any initiator or conflict with a read exclusive reservation from any initiator. Each registered I_T nexus is a reservation holder, and may write to the disk.

- ▶ 8h Exclusive access All Registrant (EAAR)

While this reservation is active, no other initiator shall be permitted any access to the indicated extent apart from registered nodes. Each registered I_T nexus is a reservation holder, and is allowed to read/write to the disk.

Table A-2 shows the read/write operations with the type of All Registrants.

Table A-2 Read and write operations with the All Registrants type

Type	WEAR (7h)	WERO (5h)	EAAR (8h)	EARO (6h)
Registered?	Not registered	Registered	Not registered	Registered
WRITE	Not allowed	Allowed	Not allowed	Allowed
READ	Allowed	Allowed	Not allowed	Allowed

In the Registrants Only (RO) type, a reservation is exclusive to one of the registrants. The reservation of the device is lost if the current PR holder removes this PR Key from the device. To avoid losing the reservation, any other registrant can replace themselves (known as a *preempt*) as the Persistent Reservation Holder. Alternatively, in the All Registrants (AR) type, the reservation is shared among all registrants.

ODM reserve policy

The AIX ODM device reserve_policy attribute must be set to open the device in any of the previous reservation types. The values are the current valid values of the reserve_policy attribute, which can be seen by using **lsattr** with the -R option, as shown in Example A-1.

Example A-1 Current valid values of the reserve_policy attribute

```
#lsattr -Rl <hdisk#> -a reserve_policy
no_reserve
single_path
PR_exclusive
PR_shared
```

Note: The values that are shown in Example A-1 on page 415 can change according to the ODM definitions or host attachment scripts that are provided by the disk or storage vendors.

The following attribute values are valid:

- ▶ The no_reserve value does not apply a reservation methodology for the device. The device can be accessed by any initiators.
- ▶ The single_path value applies a SCSI 2 reserve methodology.
- ▶ The PR_exclusive value applies SCSI 3 persistent reserve, which is an exclusive host methodology. Write Exclusive Registrants Only type of reservations require the reserve_policy attribute to be set to PR_exclusive.
- ▶ The PR_shared value applies a SCSI 3 persistent reserve shared host methodology. Write Exclusive All Registrants type of reservations require the reserve_policy attribute to be set to PR_shared.

This attribute can be set and read as shown in Example A-2.

Example A-2 Setting the disk attribute to PR_shared

```
# chdev -l hdisk1 -a reserve_policy=PR_shared  
hdisk1 changed  
  
# lsattr -El hdisk1 -a reserve_policy  
reserve_policy PR_shared Reserve Policy True+
```

The **lsattr** command with the **-E** option displays the effective policy for the disk in the AIX ODM. The **-P** option displays the policy when the device was last configured. This is the reservation information about the AIX kernel that is used to enforce the reservation during disk opens.

Setting these attributes by using the **chdev** command can fail if the resource is busy, as shown in Example A-3.

Example A-3 Setting the disk attribute with the chdev command

```
# chdev -l hdisk1 -a reserve_policy=PR_shared  
Method error (/usr/lib/methods/chgdisk):  
0514-062 Cannot perform the requested function because the specified device is busy.
```

When the device is in use, you can use the **-P** flag to **chdev** to change the effective policy only. The change is made to the database and the changes are applied to the device when the system is restarted. Another method is to use the **-U** flag where the reservation information is updated with the AIX ODM and the AIX kernel. However, not all devices support the **-U** flag. One of the ways to determine this support is to look for the **True+** value in the **lsattr** output, as shown in Example A-4.

Example A-4 Checking whether the device supports the U flag by using the lsattr command output

```
# lsattr -Pl hdisk1 -a reserve_policy  
reserve_policy PR_shared Reserve Policy True+
```

Persistent Reserve IN

Attention: You do *not* run these commands in your systems. By running these commands, this section shows you how disk reservations work, especially in a clustered environment, which demands more care while managing disk reservations.

Persistent Reserve IN (PRIN) commands are used to obtain information about active reservations and registrations on a device. The following PRIN service actions are commonly used:

- | | |
|----------------------------|---|
| Read keys | To read PR Keys of all registrants of the device. |
| Read reservation | To obtain information about the Persistent Reservation Holder. The PR Holder value is zero if the All Registrants type of reservation exists on the device; otherwise, it is the PR Key of the node holding the reservation of the device exclusively. |
| Report capabilities | To read the capability information of the device. The capability bits indicate whether the device supports persistent reservations and the types of reservation that are supported by the device. A devrsrv implementation of this service action is shown in Example A-5. |

Example A-5 Output of the devrsrv implementation

```
# devrsrv -c prin -s 2 -l hdisk1
PR Capabilities Byte[2]      : 0x1  PTPL_C
PR Capabilities Byte[3]      : 0x81 PTPL_A
PR Types Supported          : PR_WE_AR  PR_EA_RO  PR_WE_RO  PR_EA  PR_WE  PR_EA_AR
```

Persistent Preserve OUT

Attention: You do *not* run these commands in your systems. By running these commands, this section shows you how disk reservations work, especially in a clustered environment, which demands more care while managing disk reservations.

Persistent Preserve OUT (PROUT) commands are used to reserve, register, and remove the reservations and reservation keys. The following PROUT service actions are commonly used:

- | | |
|--------------------------|--|
| Register | To register and unregister a PR key with a device. |
| Reserve | To create a persistent reservation for the device. |
| Release | To release the selected persistent reservation and not remove any registrations. |
| Clear | To release any persistent reservations and remove all registrations on the device. |
| Preempt | To replace the persistent reservation or remove registrations. |
| Preempt and abort | Along with preempting, to abort all tasks for one or more preempted nodes. |

The value of the service action key and the reservation type matters when Preempt or Preempt and Abort actions are performed. Therefore, a little more information about these service actions is necessary.

A PROUT command with **PREEMPT** or **PREEMPT AND ABORT** is used to perform one of the following actions:

- ▶ Preempt (for example, replace) the persistent reservation and remove registrations.
- ▶ Remove registrations.

The **PREEMPT AND ABORT** service action is identical to the responses to a **PREEMPT** service action except that all tasks from the device that is associated with the persistent reservations or registrations being preempted (but not the task containing the **PROUT** command itself) shall be aborted. See Table A-3.

Table A-3 Effects of preempt and abort under different reservation types

Reservation type	Service action reservation key	Action
All registrants	Zero	Preempt the persistent reservation and remove registrations.
	Not zero	Remove registrations.
All other types	Zero	Invalid request.
	Reservation holder's reservation key	Preempt the persistent reservation and remove registrations.
	Any other, nonzero reservation key	Remove registrations.

Understanding register, reserve, and preempt

As an example, we have a cluster of four systems with shared access to disk, as shown in Figure A-1. Assign PR_key_value from each node, and also set the reserve_policy of the target disk to PR_shared or PR_exclusive. The unique PR_key of each device is registered with the disk and the reserved disk with SCSIPR reservation, which gives access to registered devices only.

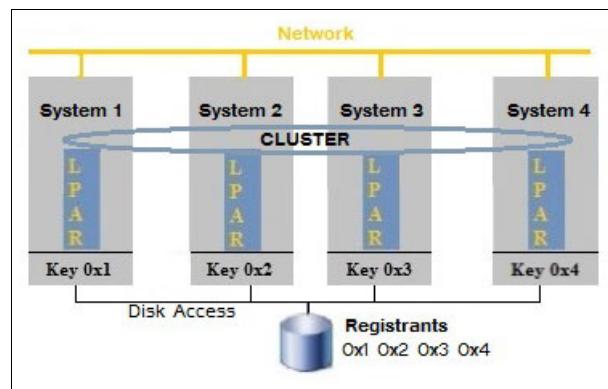


Figure A-1 Four-node cluster setup with shared disk

Perform the register action from each system (1 - 4) to register its reservation key with the disk and reserve action to establish the reservation. The PR_Holder_key value represents the current reservation holder of the disk. As shown in Table A-4 on page 419, in the RO type only one system can hold the reservation of the disk at a time (key 0x1 in our example).

However, all of the four registrant systems hold the reservation of the disk under the AR type, so you see that the PR_Holder_key value is zero.

Table A-4 Differences with RO and AR

Type	All registrant (Types 7h/8h)	Registrant only (Types 5h/6h)
Registrants	0x1 0x2 0x3 0x4	0x1 0x2 0x3 0x4
PR_Holder_Key	0	0x1

A **read key** command displays all of the reservation keys that are registered with the disk (0x1, 0x2, 0x3, and 0x4). The **read reservation** command gives the value of PR_Holder_Key, which varies per reservation type. If there is a network or any other failure such that system 1 and the rest of the systems cannot communicate with each other for a certain period, a split-brain or split-cluster situation results, as shown in Figure A-2.

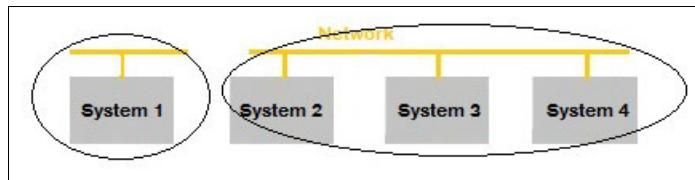


Figure A-2 Split-cluster situation

Suppose that your cluster manager decides on system 2 to take ownership (or the subcluster with system 2); then, the system can issue a **PROUT** command with a **preempt** or **preempt and abort** option and remove the PR_Key 0x1 registration from the disk. The result is that the reservation is moved away from system 1, as shown in Table A-5, and is denied access to the shared disk.

Table A-5 Differences with RO and AR

Type	All registrant (Types 7h/8h)	Registrants only (Types 5h/6h)
PR_Holder_Key	0	0x2

Preempt or preempt and abort functions can take the following arguments:

- | | |
|--------------------|--|
| Current_key | PR_key of nodes issuing the command, for example, 0x2. |
| Disk | The shared disk in discussion. |
| Action_key | PR_key on which the action must be taken. |

The action_key is 0x1 with the RO type of reservation. The action_key can be either 0 or 0x1 with the AR type of reservation. The two methods of preempting in case of an AR type are explained as follows:

- ▶ Method 1: Zero action key

If the action key is zero, the following actions take place:

- Registration of systems 1, 3, and 4 are removed.
- The persistent reservation is removed.
- A reservation from system 2 is created.

These actions result in access only to system 2, as shown in Figure A-3.

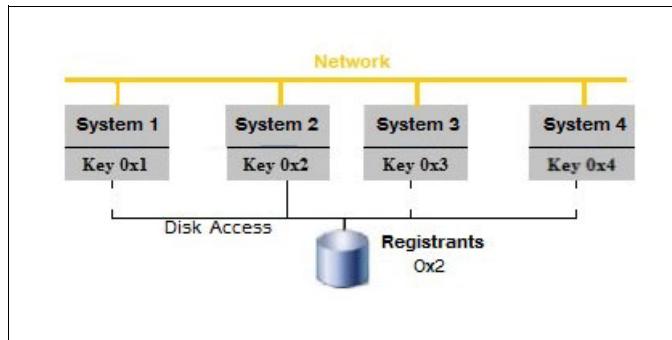


Figure A-3 Result of the preempt with action key zero

If the access to the rest of the system in active subclusters must be regained, perform an event to reregister the keys of the systems of the active cluster (systems 3 and 4).

- ▶ Method 2: Nonzero action key

If the action key is nonzero, in our case, the key of the system, there is no release of the persistent reservation, but the registration of the PR_Key 0x1 is removed, which achieves fencing, as shown in Figure A-4.

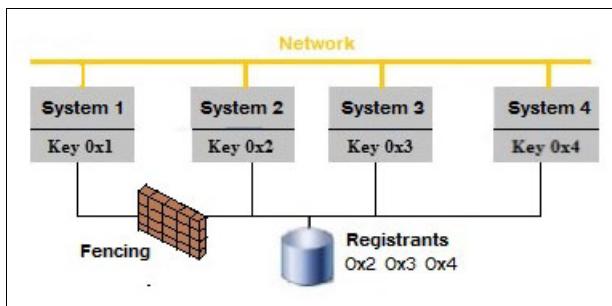


Figure A-4 Disk fencing

Table A-6 shows the result of **prin** commands after preempting system 1.

Table A-6 Difference with RO and AR

The scsipr command	All registrants (Types 7h/8h)		Registrants only (Types 5h/6h)
	Method 1	Method 2	
Read key	0x2	0x2 0x3 0x4	0x2 0x3 0x4
Read reservation	0	0	0x2

Unregister request

A registered PR_key can be removed by issuing a **register** or **register and ignore** command through that node. The service action key must be set to zero to unregister a reservation key. The list of registrants and PR_key_holder are shown in Table A-7.

Table A-7 Differences with RO and AR

Type	All registrants (Types 7h/8h)	Registrants only (Types 5h/6h)
Registrants	0x2 0x3 0x4	0x2 0x3 0x4
PR_Holder_Key	0	0x2

If the unregistered key is the PR_Holder_key (0x2) in the RO type of reservation, along with the PR_key, the reservation to the disk is also lost. Removing Key 0x2 has no impact on the reservation in the case of the AR reservation type. The same is true when other keys are removed.

Any preempt attempt by system 1 fails with a conflict because its key is not registered with the disk.

Release request

A **release** request from persistent reservation holder node releases the reservation of the disk only, and the pr_keys remains registered. Referring to Table A-7, with AR type of reservation, a release command from any of the registrants (0x2 0x3 0x4) results in the reservation being removed. In the case of RO type, a **release** command from non pr_holders (0x3 0x4) returns successfully, but with no impact on the reservation or registration. Release requests come from PR_holder (0x2) in this case.

Clear request

Referring again to Table A-7, if a **clear** request is made to the target device from any of the nodes, the persistent reservation of the disk is lost, and all of the pr_keys that are registered with the disk (0x2 0x3 0x4) are removed. As the T10 document suggests, the **clear** action must be restricted to recovery operations because it defeats the persistent reservation feature that protects data integrity.

Note: When a node opens the disk or a **register** action is performed, it registers with the PR_key value through each path to the disk. Therefore, you can see multiple registrations (I_T nexuses) with the same key. The number of registrations is equal to the number of active paths from the host to the target because each path represents an I_T nexus.

Storage

Contact your storage vendor to discover whether your device or multipathing driver is capable of SCSI Persistent Reservation, and the types of reservations it supports. Your storage vendor can also provide you the minimum firmware level, driver version that is needed, and the flags that are required to enable support for persistent reservations.

The following configurations provide examples of support for persistent reservations:

- ▶ IBM XIV®, IBM DS8000, and IBM SAN Volume Controller storages with native AIX MPIO supports¹ SCSI PR Exclusive and Shared reservations by default, as shown in Example A-6.

Example A-6 IBM storage support with native AIX MPIO of the SCSI PR Exclusive

```
# lsattr -Rl hdiskx -a reserve_policy | grep PR  
PR_exclusive  
PR_shared
```

The **devsrv** utility verifies the capability of your disks.

- ▶ Hitachi disks with native AIX MPIO support² all SCSI PR reservation types if the Host Mode Options (HMOs) 2 and 72 are set. The minimum code to support HMO72 is 70-04-31-00/00.
- ▶ EMC disks support³ PR Shared reservations and not Exclusive reservation with powerpath V6.0, as shown in Example A-7.

Example A-7 EMC disk reservation support with powerpath V6.0

```
# lsattr -Rl hdiskpowerX -a reserve_policy | grep PR  
PR_shared
```

Director bits SCSI3 Interface (SC3) and SCSI Primary Commands (SC2) must be enabled. Flag SCSI3_persist_reserv must also be enabled to use persistent reservation on powerpath devices.

More about PR reservations

During the reset sequence of the disk through a path, send a **PR IN** command with service action READ RESERVATION(01h), which returns the current reserved key on the disk if any persistent reservation exists. If an All Registrant type reservation is on the disk, the reserved key is zero.

In the case of a PR_exclusive type of reservation, the following actions occur:

- ▶ If the current reservation key is same as the node's key as in ODM, register the key by using the **PR OUT** command with the Register and Ignore Existing Key service action.
- ▶ If the current reservation key is zero and the TYPE field (persistent reservation type, as shown in Table A-1 on page 414) is also 0, which means no persistent reservation is on the disk, complete the following steps:
 - a. Register the key on the disk by using a **PR OUT** command with the **Register and Ignore Existing Key** service action.

¹ Confirm with the storage and driver vendors.

² Confirm with the storage and driver vendors.

³ Confirm with the storage and driver vendors.

- b. If not reserved already by this host, reserve it by using a **PR OUT** command with the **Reserve** service action and a type of Write Exclusive Registrants Only (5h).
- ▶ If the current reservation key is different from the current host's key, then it means that some other host holds the reservation. If you are not trying to open the disk with the **-force** flag, the open call fails. If you are trying to open the disk with the **-force** flag, complete the following steps:
 - a. Register the disk with your key running a **PR OUT** command with the **Register and Ignore Existing Key** service action.
 - b. Preempt the current reservation by running a **PR OUT** command with the **Preempt and Abort** service action to remove the registration and reservation of the current reservation holder. The key of the current reservation holder is given in the Service Action Reservation Key field.

In the case of a PR_shared reservation, the following actions occur:

If the current reservation key is zero and the TYPE field (persistent reservation type as shown in Table A-1 on page 414) is also 0, this means that there is no persistent reservation on the disk. If the TYPE field is Write Exclusive All Registrants(7h), then some other host is already registered for shared access. In either case, complete the following actions:

1. Register our key on to the disk by using a **PR OUT** command with the **Register and Ignore Existing Key** service action.
2. Reserve the disk by using a **PR OUT** command with the **RESERVE** service action and the type of Write Exclusive All Registrants (7h).

While closing the disk, for PR_exclusive reservations alone, send a **PR OUT** command with the **Clear** service action to the disk to clear all of the existing reservations and registration. This command is sent through any one of the good paths of the disk (the I_T nexus where registration was done successfully).

While changing the **reserve_policy** by using **chdev** from PR_shared to PR_exclusive, from PR_shared or PR_exclusive to single_path (or no_reserve if the key in ODM is one of the registered keys on the disk), send a **PR OUT** command with the **Clear** service action to the disk to clear all of the existing reservations and registration.

Persistent reservation commands

The **devrsrv** command of AIX queries, and can even break, persistent reservations on the device. For more information about the usage of the **devrsrv** command, see the [IBM Knowledge Center](#).

Use the following syntax for the **devrsrv** command:

```
devrsrv -c query | release | prin -s sa | (prout -s sa -r rkey -k sa_key -t
prtype) -l devicename
```

The **clrsrvmgr** command of PowerHA V7.2 lists and clears the reservation of a disk or a group of disks in a volume group (VG).

Use the following syntax for the **clrsrvmgr** command:

```
clrsrvmgr -r {[ -l DiskName] | [-g VGname]} [-v]
clrsrvmgr -c {[ -l DiskName] | [-g VGname]} [-v]
clrsrvmgr -h
```

This command lists or clears the reservation status of a disk or a VG. The command displays the following key attributes that are related to disk reservations:

Configured Reserve Policy The reservation information in the AIX kernel that is used to enforce the reservation during disk opens.

Effective Reserve Policy Reservation policy for the disk in the AIX ODM.

Reservation Status The status of the actual reservation on the storage disk itself.

The options are mostly self explanatory:

-r	Read
-c	Clear
-h	Help
-v	Verbose
-l	Expects a disk name
-g	Expects a volume group name

The manager does not guarantee the operation because disk operations depend on the accessibility of the device. However, it tries to show the reason for failure when used with the **-v** option. The utility does not support operations at both the disk and VG levels together. Therefore, the **-l** and **-g** options cannot coexist. At the VG level, the number of disks in the VG, and each target disk name, are displayed as shown in the following code:

```
# clrsrvmgr -rg PRABVG  
Number of disks in PRABVG: 2
```

hdisk1011

```
Configured Reserve Policy : no_reserve  
Effective Reserve Policy : no_reserve  
Reservation Status : No reservation
```

hdisk1012

```
Configured Reserve Policy : no_reserve  
Effective Reserve Policy : no_reserve  
Reservation Status : No reservation
```

At disk level, the disk name is not mentioned because the target device is known:

```
# clrsrvmgr -rl hdisk1015 -v  
Configured Reserve Policy : PR_shared  
Effective Reserve Policy : PR_shared  
Reservation Status : No reservation
```



B

IBM PowerHA: Live kernel update support

This appendix provides details about the PowerHA live kernel update (LKU) support.

This appendix covers the following topics:

- ▶ Live kernel update support
- ▶ Example of live kernel update patching a kernel interim fix in a PowerHA environment

Live kernel update support

Starting with AIX 7.2, the AIX operating system provides the AIX Live Update function, which eliminates the downtime that is associated with patching the AIX operating system.

PowerHA V7.2 recognizes and supports live update of cluster member nodes:

- ▶ PowerHA switches to an unmanage mode during the operation.
- ▶ It allows workload and storage activities continue to be run without interruption.
- ▶ Live update can be performed on one node in the cluster at a time.

The hardware requirements are as follows:

- ▶ All devices in the node are virtual.
- ▶ Each disk has multi-path.
- ▶ Four spare disks for LKU (disks for mirrorvg, new rootvg, temporary paging space, and temporary dump device).

Tip: This [AIX 7.2 Live Kernel update YouTube video](#) provides a demonstration of LKU.

Example of live kernel update patching a kernel interim fix in a PowerHA environment

The test environment that is used has the following configuration:

- ▶ Two-node cluster environment
- ▶ AIX 7.2.0.0 with the following filesets:
 - bos.mp64 7.2.0.0
 - bos.cluster.rte 7.2.0.0
 - bos.liveupdate.rte 7.2.0.0
- ▶ PowerHA V7.2 SP1:
 - cluster.es.server.rte 7.2.1.0

Check the environment by completing the following steps:

1. Check that the PowerHA cluster service is running and in a stable state on both nodes:

```
# clcmd lssrc -ls clstrmgrES | egrep "Current state"
  Current state: ST_STABLE
  Current state: ST_STABLE
```

2. Check that the CAA cluster is running active:

```
# lscluster -c | grep Cluster
  Cluster Name: CL102_103
  Cluster UUID: 8e1409c6-a407-11e5-8002-c6d7ab283702
  Number of nodes in cluster = 2
    Cluster ID for node kern102.aus.stglabs.ibm.com: 1
    Cluster ID for node kern103.aus.stglabs.ibm.com: 2
```

3. Check that the PowerHA RGs are online and available:

```
# clcmd c1RGinfo -m
-----
NODE kern103.aus.stglabs.ibm.com
-----
Group Name  Group State   Application state   Node
-----
RG1          ONLINE           montest        kern102
                                         Node
                                         ONLINE MONITORED
-----
NODE kern102.aus.stglabs.ibm.com
-----
Group Name  Group State   Application state   Node
-----
RG1          ONLINE           montest        kern102
                                         Node
                                         ONLINE MONITORED
```

Then, to perform the LKU, complete the following steps:

1. The HMC authentication is required to perform an LKU.

The **hmcauth** command is used to authenticate with a Hardware Management Console (HMC). For example, run the following command:

```
# hmcauth
  Enter HMC URI: dsolab134
  Enter HMC user name: hscroot
  Enter HMC password:
```

To list all the known HMC authentication tokens, run the following command:

```
# hmcauth -l
Address  : 9.3.4.134
User name: hscroot
port      : 12443
TTL       : 23:59:55 left
```

2. The **geninstall** command is used to install this kernel interim fix. For more information about the command, see the [IBM Knowledge Center](#).

The flags that are used in the **geninstall** command are explained as follows:

- p Performs a preview of an action by running all the preinstallation checks for the specified action.
- d The device or directory that specifies the device or directory containing the images to install.
- k Specifies that the AIX Live Update operation is to be performed. This is a new flag and for LKU.

3. Use the **-p** flag to preview. The output shows whether any action must be corrected before installing this interim fix package. For example, run the following command:

```
# geninstall -p -k -d /home/ dummy.150813.epkg.Z
```

```
Validating live update input data.
```

```
Computing the estimated time for the live update operation:
```

```
-----  
LPAR: kern102
```

```
Blackout_time(s): 37
```

```
Global_time(s): 939
```

```
Checking mirror vg device size:
```

```
-----  
Required device size: 15104 MB
```

```
Given device size: 32767 MB
```

```
PASSED: device size is sufficient.
```

```
Checking new root vg device size:
```

```
-----  
Required device size: 15104 MB
```

```
Given device size: 32767 MB
```

```
PASSED: device size is sufficient.
```

```
Checking temporary storage size for original LPAR:
```

```
-----  
Required device size: 1024 MB
```

```
Given device size: 32767 MB
```

```
PASSED: device size is sufficient.
```

```
Checking temporary storage size for surrogate LPAR:
```

```
-----  
Required device size: 1024 MB
```

```
Given device size: 20479 MB
```

```
PASSED: device size is sufficient.
```

```
Validating the adapters and their paths:
```

```
-----  
PASSED: adapters can be divided into two sets so that each has paths to all  
disks.
```

```
Checking lpar minimal memory size:
```

```
-----  
Required memory size: 2048 MB
```

```
Current memory size: 8192 MB
```

```
PASSED: memory size is sufficient.
```

```
Checking other requirements:
```

```
-----  
PASSED: sufficient space available in /var.
```

```
PASSED: sufficient space available in /.
```

```
PASSED: sufficient space available in /home.
```

```
PASSED: no existing altinst_rootvg.
```

```
PASSED: rootvg is not part of a snapshot.
```

```
PASSED: pkcs11 is not installed.
```

```
PASSED: DoD/DoDv2 profile is not applied.
```

PASSED: Advanced Accounting is not on.
PASSED: Virtual Trusted Platform Module is not on.
PASSED: multiple semid lists is not on.
PASSED: The trustchk Trusted Execution Policy is not on.
PASSED: The trustchk Trusted Library Policy is not on.
PASSED: The trustchk TSD_FILES_LOCK policy is not on.
PASSED: the boot disk is set to the current rootvg.
PASSED: the mirrorvg name is available.
PASSED: the rootvg is uniformly mirrored.
PASSED: the rootvg does not have the maximum number of mirror copies.
PASSED: the rootvg does not have stale logical volumes.
PASSED: all of the mounted file systems are of a supported type.
PASSED: this AIX instance is not diskless.
PASSED: no Kerberos configured for NFS mounts.
PASSED: multibos environment not present.
PASSED: Trusted Computing Base not defined.
PASSED: no local tape devices found.
PASSED: live update not executed from console.
PASSED: the execution environment is valid.
PASSED: enough available space for /var to dump Component Trace buffers.
PASSED: enough available space for /var to dump Light weight memory Trace buffers.
PASSED: all devices are virtual devices.
PASSED: No active workload partition found.
PASSED: nfs configuration supported.
PASSED: HMC token is present.
PASSED: HMC token is valid.
PASSED: HMC requests successful.
PASSED: A virtual slot is available.
PASSED: RSCT daemons are active.
PASSED: no Kerberos configuration.
PASSED: lpar is not remote restart capable.
PASSED: no virtual log device configured.
PASSED: lpar is using dedicated memory.
PASSED: the disk configuration is supported.
PASSED: no Generic Routing Encapsulation (GRE) tunnel configured.
PASSED: Firmware level is supported.
PASSED: vNIC resources available.
PASSED: Consolidated system trace buffers size is within the limit of 64 MB.
PASSED: SMT number is valid.
INFO: Any system dumps present in the current dump logical volumes will not be available after live update is complete.

4. Update the /var/adm/ras/liveupdate/lvupdate.data file:

```
# cat /var/adm/ras/liveupdate/lvupdate.data
    --- start ---
    software:

        single = /home/dummy.150813.epkg.Z
    --- EOF ---
```

5. Edit this file and add the following fields:

```
general:  
    kext_check =  
  
disks:  
    nhdisk = <hdisk#>  
    mhdisk = <hdisk#>  
    tohdisk = <hdisk#>  
    tshdisk = <hdisk#>  
  
hmc:  
    lpar_id =  
    management_console = dsolab134  
    user = hscroot
```

Note: For the disks description, the /var/adm/ras/liveupdate/lvupdate.template file provides the following information:

```
disks:  
    nhdisk =  
    mhdisk =  
    tohdisk =  
    tshdisk =  
# disk:  
#     nhdisk = <disk1,disk2,...> The disk names to be used to make a copy  
#          of the original rootvg which will be used to boot disk the  
#          Surrogate  
#          (surr-boot-rootvg). The capacity needs to match the capacity of  
#          the “required” file systems (/, /var, /opt, /usr, /etc) from the  
#          orig-rootvg. (If preview mode, size checking will be performed)  
#     mhdisk = <disk1,disk2,...> The disk names to be used for the mirrored  
#          rootvg (surr-mir-rootvg) on the Surrogate. The capacity needs to  
#          match the capacity of orig-rootvg. (If preview mode, size checking  
#          will be performed.)  
#     tohdisk = <disk1,disk2,...> The name of disks to be used as temporary  
#          storage for the Original. This is only required if paging space is  
#          present on a non-rootvg disk, or if a dump device is present  
#          (either  
#          on rootvg or non-rootvg). The capacity needs to match the total  
#          capacity  
#          of paging devices and dump devices defined for the original  
#          partition.  
#          (If preview mode, size checking will be performed.)  
#     tshdisk = <disk1,disk2,...> The name of disks to be used as temporary  
#          storage for the Surrogate. This is only required if paging space  
#          is  
#          present on a non-rootvg disk, or if a dump device is present  
#          (either  
#          on rootvg or non-rootvg). It must have the same capacity as  
#          tohdisk.  
#          (If preview mode, size checking will be performed.)
```

For more information, see [IBM Knowledge Center](#).

For example, you might receive the following information:

```
general:  
    kext_check =  
  
disks:  
    nhdisk = hdisk1  
    mhdisk = hdisk2  
    tohdisk = hdisk3  
    tshdisk = hdisk7  
  
hmc:  
    lpar_id =  
    management_console = dsolab134  
    user = hscroot  
software:  
    single = /home/dummy.150813.epkg.Z
```

6. Install the interim fix by running the following command. The flags that are used in the commands are described as follows:

-d device or directory	Specifies the device or directory containing the images to install.
-k	Specifies that the AIX Live Update operation is to be performed. This is a new flag and for LKU.

```
# geninstall -k -d /home/ dummy.150813.epkg.Z  
Validating live update input data.  
Computing the estimated time for the liveupdate operation:
```

```
-----  
LPAR: kern102  
Blackout_time(s): 82  
Global_time(s): 415
```

```
Checking mirror vg device size:
```

```
-----  
Required device size: 7808 MB  
Given device size: 32767 MB  
PASSED: device size is sufficient.
```

```
Checking new root vg device size:
```

```
-----  
Required device size: 7808 MB  
Given device size: 32767 MB  
PASSED: device size is sufficient.
```

```
Checking temporary storage size for the original LPAR:
```

```
-----  
Required device size: 1024 MB  
Given device size: 32767 MB  
PASSED: device size is sufficient.
```

```
Checking temporary storage size for the surrogate LPAR:
```

```
-----  
Required device size: 1024 MB  
Given device size: 20479 MB  
PASSED: device size is sufficient.
```

Validating the adapters and their paths:

PASSED: adapters can be divided into two sets so that each has paths to all disks.

Checking lpar minimal memory size:

Required memory size: 2048 MB

Current memory size: 8192 MB

PASSED: memory size is sufficient.

Checking other requirements:

PASSED: sufficient space available in /var.

PASSED: sufficient space available in /.

PASSED: sufficient space available in /home.

PASSED: no existing altinst_rootvg.

PASSED: rootvg is not part of a snapshot.

PASSED: pkcs11 is not installed.

PASSED: DoD/DoDv2 profile is not applied.

PASSED: Advanced Accounting is not on.

PASSED: Virtual Trusted Platform Module is not on.

PASSED: The trustchk Trusted Execution Policy is not on.

PASSED: The trustchk Trusted Library Policy is not on.

PASSED: The trustchk TSD_FILES_LOCK policy is not on.

PASSED: the boot disk is set to the current rootvg.

PASSED: the mirrorvg name is available.

PASSED: the rootvg is uniformly mirrored.

PASSED: the rootvg does not have the maximum number of mirror copies.

PASSED: the rootvg does not have stale logical volumes.

PASSED: all of the mounted file systems are of a supported type.

PASSED: this AIX instance is not diskless.

PASSED: no Kerberos configured for NFS mounts.

PASSED: multibos environment not present.

PASSED: Trusted Computing Base not defined.

PASSED: no local tape devices found.

PASSED: live update not executed from console.

PASSED: the execution environment is valid.

PASSED: enough available space for /var to dump Component Trace buffers.

PASSED: enough available space for /var to dump Light weight memory Trace buffers.

PASSED: all devices are virtual devices.

PASSED: No active workload partition found.

PASSED: nfs configuration supported.

PASSED: HMC token is present.

PASSED: HMC token is valid.

PASSED: HMC requests successful.

PASSED: A virtual slot is available.

PASSED: RSCT services are active.

PASSED: no Kerberos configuration.

PASSED: lpar is not remote restart capable.

PASSED: no virtual log device configured.

PASSED: lpar is using dedicated memory.

PASSED: the disk configuration is supported.

PASSED: no Generic Routing Encapsulation (GRE) tunnel configured.

```
PASSED: Firmware level is supported.  
PASSED: vNIC resources available.  
PASSED: Consolidated system trace buffers size is within the limit of 64 MB.  
PASSED: SMT number is valid.  
INFO: Any system dumps present in the current dump logical volumes will not be  
available after live update is complete.  
  
Non-interruptable live update operation begins in 10 seconds.  
Broadcast message from root@kern102 (pts/3) at 22:20:18 ...  
  
Live AIX update in progress.  
  
.....  
Initializing live update on original LPAR.  
  
Validating original LPAR environment.  
  
Beginning live update operation on original LPAR.  
  
Requesting resources required for live update.  
.....  
Notifying applications of impending live update.  
....  
Creating rootvg for boot of surrogate.  
.....  
Starting the surrogate LPAR.  
.....  
  
Broadcast message from root@kern102 (tty) at 22:26:02 ...  
  
PowerHA SystemMirror on kern102 shutting down. Please exit any cluster  
applications...  
  
Creating mirror of original LPAR's rootvg.  
.....  
Moving workload to surrogate LPAR.  
.....  
    Blackout Time started.  
.....  
    Blackout Time end.  
  
Workload is running on surrogate LPAR.  
.....  
Shutting down the Original LPAR.  
.....The live update operation succeeded.  
  
Broadcast message from root@kern102 (pts/3) at 22:33:05 ...  
  
Live AIX update completed.  
  
File /etc/inittab has been modified.  
  
One or more of the files listed in /etc/check_config.files have changed.
```

See /var/adm/ras/config.diff for details.

During LKU, the PowerHA switches into an unmanaged mode:

```
# lssrc -ls clstrmgrES
Current state: ST_STABLE
sccsid = "@(#)36 1.135.1.125
src/43haes/usr/sbin/cluster/hacmpd/main.C,hacmp.pe,71haes_r721,1532A_hacmp721
7/31/15"
build = "Dec 2 2015 04:17:07 1549A_hacmp721"
i_local_nodeid 1, i_local_siteid -1, my_handle 2
m1_idx[1]=0 m1_idx[2]=1
Forced down node list: kern102
AIX Live Update operation in progress on node list: kern102
...
# c1RGinfo -m
-----
Group Name Group State Application state Node
-----
RG1 UNMANAGED kern102
montest OFFLINE
RG1 UNMANAGED kern103
montest OFFLINE
```

AIX Live Update is automatically enabled at PowerHA V7.2.0 and AIX 7.2.0 and later versions. AIX Live Update is not supported on AIX 7.1.X with PowerHA V7.2.0 installed. However, if you are upgrading AIX to Version 7.2.0 or later, you must enable the AIX Live Update function in PowerHA to use the Live update support of AIX.

Here is some additional information about this topic:

- ▶ AIX Live Update activation / deactivation can be performed by using the SMIT menu:

```
# smitty sysmirror
      Cluster Nodes and Networks
          Manage Nodes
              Change>Show a Node
```

A new field that is called Enable AIX Live Update operation can be set to Yes or No (to enable or disable the AIX Live update operation). This action must be performed on each node in the cluster, one node at a time.

- ▶ AIX Live Update and PowerHA tips and logs:

- The **lssrc -ls clstrmgrES** command shows the list of nodes in the cluster that are processing a Live Update operation.
- Here are the logs that are generated by a cluster script during the AIX Live Update operation:

```
/var/hacmp/log/lvupdate_orig.log
/var/hacmp/log/lvupdate_surr.log
```

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 topic in this document. Some publications referenced in this list might be available in softcopy only.

- ▶ *IBM PowerHA SystemMirror 7.1.2 Enterprise Edition for AIX*, SG24-8106
- ▶ *IBM PowerHA SystemMirror for AIX Cookbook*, SG24-7739
- ▶ *IBM PowerHA SystemMirror Standard Edition 7.1.1 for AIX Update*, SG24-8030
- ▶ *IBM PowerHA SystemMirror V7.2 for IBM AIX Updates*, SG24-8278
- ▶ *Power Enterprise Pools on IBM Power Systems*, REDP-5101

You can search for, view, download or order these documents and other Redbooks, Redpapers, web docs, draft, and additional materials, at the following website:

ibm.com/redbooks

Other publications

This publication is also relevant as a further information source:

- ▶ *IBM RSCT for AIX: Guide and Reference*, SA22-7889

Online resources

These websites are also relevant as further information sources:

- ▶ PowerHA SystemMirror Interim Fix Bundles information:
https://aix.software.ibm.com/aix/ifixes/PHA_Migration/ha_install_mig_fixes.htm
- ▶ PowerHA wiki:
<http://tinyurl.com/phawiki>
- ▶ PowerHA LinkedIn group:
<https://www.linkedin.com/grp/home?gid=8413388>

Help from IBM

IBM Support and downloads

ibm.com/support

IBM Global Services

ibm.com/services

Redbooks

IBM PowerHA SystemMirror V7.2.1 for IBM AIX Updates

SG24-8372-00

ISBN 0738442518



(0.5" spine)
0.475" <-> 0.875"
250 <-> 459 pages



SG24-8372-00

ISBN 0738442518

Printed in U.S.A.

Get connected

