

Content Manager OnDemand Backup, Recovery, and High Availability

Introducing basic concepts, strategies,
options, and procedures

Covering multiplatforms,
iSeries, and z/OS

Including real world case
studies



Wei-Dong Zhu
Monti Abrahams
Doris Ming Ming Ngai
Sandi Pond
Hernán Schiavi
Hassan A. Shazly
Ed Stonesifer
Vanessa Stonesifer



International Technical Support Organization

**Content Manager OnDemand Backup, Recovery,
and High Availability**

October 2005

Archived

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

First Edition (October 2005)

This edition applies to Version 8, Release 3 of IBM DB2 Content Manager OnDemand for Multiplatforms (product number 5724-J33); Version 7, Release 1 of IBM DB2 Content Manager OnDemand for z/OS and OS/390 (Program Number 5655-H39); Version 5, Release 3 of IBM DB2 Content Manager OnDemand for iSeries OS/400 (i5/OS) (product number 5722-RD1).

© Copyright International Business Machines Corporation 2005. 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

Figures	ix
Tables	xi
Notices	xiii
Trademarks	xiv
Preface	xv
The team that wrote this redbook	xvi
Become a published author	xviii
Comments welcome	xviii
Part 1. Introduction	1
Chapter 1. Basic concepts	3
1.1 Backup and recovery concept	4
1.1.1 Nature of failure	4
1.1.2 Time for recovery	6
1.1.3 Backup types	8
1.1.4 Backup window considerations	9
1.1.5 Space requirement	9
1.1.6 Recovery points	10
1.2 High availability	11
1.2.1 Planned versus unplanned outages	11
1.2.2 High availability versus continuous availability	12
1.2.3 Levels of availability	13
1.2.4 Measuring availability	16
1.3 Disaster recovery	16
1.3.1 Disaster recovery plan	16
1.3.2 Objectives and benefits	17
1.3.3 The seven tiers of disaster recovery	18
1.3.4 Trends in disaster recovery planning	21
1.4 Business continuity strategies and options	22
1.4.1 Cost versus loss	23
1.4.2 Solution design	25
1.4.3 Critical considerations and success factors	25
Chapter 2. Content Manager OnDemand overview	27
2.1 Introduction	28

2.1.1 OnDemand features and functions	28
2.2 System overview	29
2.2.1 Library Server	31
2.2.2 Object Server	31
2.2.3 Standard OnDemand system	32
2.2.4 Standard OnDemand system with TSM	33
2.2.5 Distributed OnDemand system	33
2.2.6 Distributed OnDemand system with TSM	34
2.3 OnDemand terminology and concepts	34
2.3.1 Application.	35
2.3.2 Application group	36
2.3.3 Folder	36
2.4 Database management.	36
2.4.1 Overview and terms	36
2.4.2 Database maintenance	39
2.4.3 Database utility	41
2.5 Storage management	41
2.5.1 Overview	41
2.5.2 OnDemand storage management policy and objects	42
2.5.3 Cache storage and archive storage	43
2.5.4 Storage set and storage nodes.	44
2.6 Report migration and removal.	45
2.6.1 Migrating reports	45
2.6.2 Removing reports	45
2.6.3 Removing reports from cache storage	46
2.7 Tivoli Storage Manager	46
2.7.1 TSM storage management policy and objects	47
2.7.2 TSM storage devices and media.	48
Part 2. Multiplatforms	49
Chapter 3. Backup and recovery for OnDemand Multiplatforms	51
3.1 Backup strategies and options	52
3.1.1 Operating system backup	52
3.1.2 Cache backup	52
3.1.3 Database backup	53
3.1.4 TSM backup	57
3.1.5 OnDemand configuration and definition	58
3.2 Practical procedures	59
3.2.1 Operating system	59
3.2.2 DB2 backup procedures	60
3.2.3 Cache backup procedure	64
3.2.4 TSM backup procedure.	66

3.2.5 OnDemand configuration and definition backup	70
3.3 Recovery plans under different scenarios	75
3.3.1 Human factor.	75
3.3.2 Hardware failure	78
3.3.3 Transaction failure.	78
3.3.4 Disaster.	78
3.4 Recovery procedures	78
3.4.1 Recovery of operating system.	79
3.4.2 TSM recovery procedure.	79
3.4.3 Recovery of DB2 database	84
3.4.4 Recovery of cache directory	86
3.4.5 After the restoration.	87
3.5 Problem determination	88
Chapter 4. High availability and business continuity for OnDemand Multiplatforms	89
4.1 OnDemand high availability strategies and options	90
4.1.1 High availability: Clustering.	90
4.2 Practical procedures for high availability	96
4.2.1 Test case scenario	96
4.2.2 Steps to configure example 1	97
4.2.3 Steps to configure example 2	128
4.2.4 HACMP post-configuration procedures.	144
4.2.5 Failover tests and results	146
4.3 Business continuity strategies and options	148
4.3.1 Multi-site solutions overview	148
4.3.2 Business continuity configuration examples	150
4.3.3 eRCMF (enterprise Remote Copy Management Facility)	158
4.3.4 Business continuity summary chart.	159
Chapter 5. Case studies for OnDemand Multiplatforms	161
5.1 Global voice and data communications company	162
5.1.1 Background.	162
5.1.2 Backup, recovery, and high availability approach	162
5.2 International financial services company.	165
5.2.1 Background.	165
5.2.2 Backup, recovery, and high availability approach.	165
Part 3. iSeries	169
Chapter 6. iSeries architecture.	171
6.1 iSeries overview	172
6.1.1 iSeries success factors	172
6.1.2 iSeries architecture overview	176

6.1.3 Summary	180
Chapter 7. OnDemand for iSeries overview	183
7.1 Introduction	184
7.2 Installation and configuration	185
7.3 Administration	187
7.4 User interface	189
7.5 Summary	191
Chapter 8. Backup and recovery for OnDemand iSeries	193
8.1 Overview	194
8.2 Database and system files backup	195
8.2.1 System save	195
8.2.2 OnDemand libraries	197
8.2.3 OnDemand directories	198
8.3 Optical media backup	205
8.4 Optical media considerations	210
8.5 Database recovery	212
8.6 Backup, Recovery and Media Services (BRMS)	214
Chapter 9. High availability strategies and options for the OnDemand iSeries	215
9.1 Introduction	216
9.2 Journaling	216
9.3 Remote journaling	220
9.4 High Availability Business Partner solutions	222
Chapter 10. Case studies for OnDemand iSeries	223
10.1 Scenario 1	224
10.2 Scenario 2	224
10.3 Scenario 3	225
10.4 Scenario 4	227
10.5 Scenario 5	229
10.6 Scenario 6	230
10.7 Scenario 7	232
10.8 Conclusion	232
Part 4. z/OS	233
Chapter 11. OnDemand overview for z/OS	235
11.1 Introduction	236
11.1.1 OnDemand features and functionalities	236
11.2 System overview	237
11.2.1 Library Server	238

11.2.2	Object Server	239
11.2.3	Server configurations	239
11.3	OnDemand terminology and concept	242
11.3.1	Applications	242
11.3.2	Application groups	242
11.3.3	Folders	243
11.3.4	Resources	244
11.3.5	Storage sets and storage nodes	244
11.3.6	Cache and archive storage	245
11.4	OnDemand Web Enablement Kit (ODWEK)	245
11.4.1	11.4.1 OnDemand programming interface	246
11.4.2	Document viewing	248
Chapter 12. Backup and recovery for OnDemand z/OS		251
12.1	Backup and recovery overview	252
12.2	Library Server backup and recovery	253
12.2.1	OnDemand software	253
12.2.2	OnDemand server information	253
12.2.3	OnDemand database	253
12.2.4	Other configuration and product files	259
12.3	Object Server backup and recovery	260
12.3.1	Stored reports	260
12.3.2	Cache storage	260
12.3.3	Archive storage	260
12.4	Object Access Method (OAM)	261
12.4.1	OAM components and SMS terminologies	262
12.4.2	Establishing OAM recovery procedures	264
12.5	Virtual Storage Access Method (VSAM)	271
12.5.1	DFSMSHsm	271
12.6	Tivoli Storage Manager (TSM)	275
12.6.1	TSM Overview	275
12.6.2	TSM as the OnDemand z/OS archive manager	276
12.6.3	Backup methodologies	277
12.6.4	TSM supplied functionality	279
Chapter 13. High availability for OnDemand z/OS in a SYSPLEX environment		281
13.1	High availability on z/OS overview	282
13.1.1	z/OS the nucleus high availability component	282
13.1.2	High availability concept for OnDemand z/OS	283
13.2	HA strategies for an OnDemand z/OS application	286
13.2.1	A 4-tier logical model	286
13.2.2	Breaking out the data tier (Tier 4)	288

13.2.3 Intelligent routing of inbound traffic options	290
13.2.4 Achieving HA for OnDemand system	292
13.2.5 Availability strategy failure scenarios	295
13.3 Sysplex terminology	301
13.4 TCP/IP port sharing	303
13.5 The shared OnDemand server	304
Chapter 14. Case study for OnDemand z/OS	307
14.1 International financial services company	308
14.1.1 Background	308
14.1.2 Backup, recovery, and high availability approach	308
14.2 Communications services company	310
14.2.1 Background	310
14.2.2 Backup, recovery, and high availability approach	311
14.3 Manufacturing company	312
14.3.1 Background	313
14.3.2 Backup, recovery, and high availability approach	313
Part 5. Appendices	315
Appendix A. Sample scripts and programs for the high availability scenarios	317
A.1 HACMP standby configuration scripts	318
A.1.1 Standby configuration startup script	318
A.1.2 Standby configuration shutdown script	320
A.2 HACMP mutual takeover configuration scripts	321
A.2.1 Mutual takeover configuration startup scripts	321
A.2.2 Mutual takeover configuration shutdown scripts	326
A.3 OnDemand load daemon cleanup script	329
Appendix B. Sample PPRC scripts for failover	335
Sample PPRC scripts	336
Related publications	341
IBM Redbooks	341
Other publications	341
Online resources	342
How to get IBM Redbooks	343
Help from IBM	343
Index	345

Figures

1-1	High availability and continuous availability	12
1-2	High availability tiers	14
1-3	Seven tiers of disaster recovery solutions	19
1-4	Cost versus lost	24
2-1	OnDemand distributed system architecture	29
2-2	Major OnDemand system components	30
2-3	System, instance, database, table, tablespace, SMS tablespace	38
2-4	Storage management overview	42
3-1	Add a server.	72
3-2	Add a server with local protocol.	72
3-3	Export application groups	73
3-4	Create summary report for an application group	74
4-1	High availability standby site configuration	93
4-2	High availability mutual takeover site configuration	95
4-3	Create a logical volume	100
4-4	Create a JFS2 log	102
4-5	Add an enhanced JFS over a predefined logical volume.	104
4-6	Deactivate a volume group	112
4-7	Configure nodes to an HACMP cluster (standard) smit panel	117
4-8	Add an IP-based network to the HACMP cluster smit panel	119
4-9	Add a communication interface smit panel	121
4-10	Add a service IP label/address configurable on multiple nodes.	122
4-11	Add an application server smit panel.	123
4-12	Add a cascading resource group (extended) smit panel	124
4-13	Change/show resources/attributes for a cascading resource group	125
4-14	Display HACMP configuration smit output	126
4-15	HACMP verification and synchronization (active cluster on local node)	128
4-16	Add a service IP label/address configurable on multiple nodes.	139
4-17	Add an application server smit panel	140
4-18	Display HACMP configuration smit output	143
4-19	Business continuity active/standby configuration.	151
4-20	Business continuity shared load, active/active configuration	154
4-21	Business continuity primary/secondary with disk replication	156
5-1	High availability configuration two-for-one standby	164
5-2	Disaster recovery configuration multi-site active/active	167
6-1	Technology-Independent Machine Interface (TIMI)	177
6-2	Object types and associated operations	178
6-3	Single-level storage architecture	179

6-4	Dynamic LPAR.	180
7-1	Some of the OnDemand product options.	186
7-2	OnDemand administration client	187
7-3	Report wizard.	188
7-4	iSeries Navigator OnDemand plug-in	189
7-5	Windows end-user client interface	190
7-6	OnDemand Web browser interface	191
8-1	OnDemand system directory	199
8-2	OnDemand instance directory	201
8-3	HTTP configuration directory	204
8-4	Migration policy backup option.	208
8-5	Optical storage groups.	209
8-6	Storage level backup	210
9-1	Journaling a database table	218
9-2	Journaling an IFS object	219
9-3	Remote journaling	221
10-1	Migration policy tape backup option.	228
10-2	Create backup optical storage group	230
10-3	Specify backup optical storage group	231
11-1	OnDemand for z/OS logical model	238
11-2	Standard Library Server and Object Server configuration	240
11-3	Distributed Library Server and Object Server configuration.	241
11-4	Application, application groups, and folders' relationship	244
11-5	Single instance using the Java interface	247
11-6	Three instance OWDEK topology	248
12-1	A hierarchy of DB2 structures	255
12-2	TSM in an OnDemand environment	277
13-1	End-to-end tier components diagram.	287
13-2	SYSPLEX basic structure	289
13-3	breakout of SYSPLEX basic structure	289
13-4	Intelligent routing of inbound traffic options	291
13-5	Sysplex and dynamic virtual IP addressing (VIPA)	293
13-6	Availability strategy for Library Server failure.	296
13-7	Availability strategy - Object Server failure	297
13-8	Availability strategy - Sysplex Distributor failure	298
13-9	Availability strategy - z/OS failure	299
13-10	Port sharing	304
14-1	High availability configuration using a parallel sysplex	310
14-2	High availability configuration using port sharing.	312
14-3	High availability configuration for a growing manufacturing company	314

Tables

3-1	OnDemand database backup commands	61
3-2	Different user types and their default rights	76
3-3	Sample configuration of TSM database and log volumes	80
4-1	High availability summary chart	96
4-2	OnDemand file system layout	98
4-3	Shared file system sizes	103
4-4	Library Server IDs	105
4-5	High availability IP addresses	115
4-6	Serial device communications	120
4-7	Configuration files summary	130
4-8	Resource groups detailed	139
4-9	Business continuity summary chart	159
5-1	Software installed on the active nodes and the standby node.	163
8-1	System save commands	196
9-1	Default OnDemand journals.	217

Notices

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

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

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

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

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

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

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

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


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

COPYRIGHT LICENSE:

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

Trademarks

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

@server®	DB2 Universal Database™	Lotus®
eServer™	DB2®	MVS™
iSeries™	DFSMS/MVS®	Operating System/400®
i5/OS™	DFSMSdfp™	OS/390®
pSeries®	DFSMSshsm™	OS/400®
z/OS®	Enterprise Storage Server®	Parallel Sysplex®
zSeries®	FlashCopy®	Redbooks™
Advanced Function Presentation™	Geographically Dispersed Parallel Sysplex™	Redbooks (logo)  ™
Advanced Function Printing™	GDPS®	RS/6000®
AFP™	HyperSwap™	Tivoli®
AIX 5L™	HACMP™	TotalStorage®
AIX®	Infoprint®	Virtualization Engine™
Domino®	IBM®	VTAM®
		WebSphere®

The following terms are trademarks of other companies:

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

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

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

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

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

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

Preface

This IBM® Redbook helps you understand backup, recovery, high availability, business continuity strategies, and options available for IBM DB2® Content Manager OnDemand for Multiplatforms, IBM @server® iSeries™, and z/OS®.

In Part 1 of the redbook, we introduce the basic concepts of backup and recovery, high availability, disaster recovery, and business continuity. In addition, we provide an overview of IBM DB2 Content Manager OnDemand.

In Part 2, we focus on OnDemand for multiplatforms. We describe the backup and recovery strategies and options, and high availability and business continuity strategies and options for OnDemand on Multiplatforms. We provide practical procedures and steps to accomplish the backup, recovery, and high availability with sample commands and scripts. In addition, two case studies are presented to show you how real-world businesses implement backup procedures, high availability configurations, and disaster recovery plans.

In Part 3, we focus on OnDemand for iSeries. We provide an introduction to the overall iSeries architecture and an overview of OnDemand for iSeries. In addition, we discuss backup and recovery strategies, and introduce some high availability options and strategies to assist you in selecting an appropriate solution to satisfy your business requirements and to suit your operating environment.

In Part 4, we focus on OnDemand for z/OS. We begin with an overview of OnDemand for z/OS. We discuss the backup and recovery strategies and options, and describe different options performing backup and recovery of OnDemand for z/OS systems. In addition, we provide options and strategies to achieve high availability and business continuity to assist you in meeting your business requirements and your operating environment.

This book is intended for IT architects, specialists, and OnDemand system administrators who are responsible for designing, implementing, and maintaining OnDemand systems for various platforms.

Note that IBM i5/OS™ is the next generation of OS/400®. Sections in this redbook may refer to i5/OS as OS/400.

The team that wrote this redbook

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

Wei-Dong Zhu (Jackie) is a Content Manager Project Leader with the International Technical Support Organization at the Almaden Research Center in San Jose, California. She has more than ten years of software development experience in accounting, image workflow processing, and digital media distribution. She holds a master's degree in Computer Science from the University of Southern California. Jackie joined IBM in 1996. She is a Certified Solution Designer for IBM DB2 Content Manager.

Monti Abrahams is an iSeries IT Specialist at IBM in South Africa. He is an IBM Certified Solutions Expert: DB2 Content Manager OnDemand iSeries and a Certified SAP Technical Consultant with more than 12 years experience in OS/400 and DB2 UDB for iSeries. Monti provides technical support to OnDemand and SAP customers on iSeries throughout South Africa and Namibia. He also conducts training courses for IBM IT Education Services and provides general iSeries technical customer support. Monti has co-authored a number of Redbooks™ on various iSeries-related topics.

Doris Ming Ming Ngai is an IT Availability Specialist at IBM Singapore. She holds a degree in Electrical Engineering from National University of Singapore. She has been working in Integrated Technology Services for six years, in the pSeries® Services and Support Team, spending most of her time implementing and supporting OnDemand on AIX® and Windows®. Her areas of expertise include AIX, OnDemand and TSM.

Sandi Pond is a Performance Analyst for the Content Manager OnDemand development team at IBM in the USA. She has six years of experience in the Content Management field. Her areas of expertise include AIX, DB2 UDB, and OnDemand. Sandi is a Certified Solution Expert for OnDemand for Multiplatforms.

Hernán Schiavi is an IT Specialist of Services and Support for pSeries AIX with IBM Global Services, Integrated Technology Services, IBM Argentina. He has five years of experience in installation, configuration, implementation, administration, problem resolution, and support for AIX and SP2 systems. He also worked in the pre-sales area for the RS/6000® servers, focused mainly on architecture and configurations. Hernán's areas of expertise include implementations of high availability (HACMP™) and performance analysis for large systems. He also provides support for Tivoli® software with the IBM Tivoli Storage Manager product.

Hassan A. Shazly is a Senior Software Engineer in the USA. He has 30 years of experience in Information Systems. He has worked at IBM for nine years. His current areas of expertise include Content Manager OnDemand for z/OS and OS/390®. He has written multiple articles and presented topics at several conferences, including client/server technology, image processing and systems performance.

Ed Stonesifer is a Certified Consulting IT Specialist with the Content Management East Team in the USA. Ed has 24 years of experience in Information Technology and 16 years of experience with IBM OnDemand specializing on the IBM @server zSeries® platform using OAM, DB2, and OnDemand z/OS.

Vanessa Stonesifer is a Consultant IT Specialist of Content Manager OnDemand for z/OS and OS/390 at IBM in the USA. She is the Technical Team Leader for Content Manager Services with more than six years of experience in Content Manager OnDemand for z/OS and OS/390. Vanessa provides implementation and migration services for Content Manager OnDemand for z/OS. Her areas of expertise include OAM, DB2, and OnDemand for z/OS.

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

Emma Jacobs
International Technical Support Organization, San Jose Center

Carol Allen
Benjamin Boltz
Darrell Bryant
Nelson Chen
Atul V. Gore
Richard Heffel
James Ilardi
Christopher Lewis
Nancy O'Brien
Leonora Wang
Kevin Van Winkle
IBM Software Group/Information Management, USA

Sanjoy Das
Vijayavenkatesh Yelanji
IBM Systems and Technology Group/Development, San Jose, CA, USA

Become a published author

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

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

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

ibm.com/redbooks/residencies.html

Comments welcome

Your comments are important to us!

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

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

ibm.com/redbooks

- ▶ Send your comments in an email to:

redbook@us.ibm.com

- ▶ Mail your comments to:

IBM Corporation, International Technical Support Organization
Dept. QXXE Building 80-E2
650 Harry Road
San Jose, California 95120-6099



Part 1

Introduction

In Part 1, we introduce the basic concepts of backup, recovery, high availability, disaster recovery, and business continuity. In addition, we provide an overview of IBM DB2 Content Manager OnDemand.

Archived

Basic concepts

In this chapter, we provide the basic concepts covered in this redbook for these areas:

- ▶ Backup and recovery
- ▶ High availability
- ▶ Disaster recovery
- ▶ Business continuity

1.1 Backup and recovery concept

Corporate data is extremely important for any company to function. A backup and recovery strategy of the data should be a part of an overall data management plan. With appropriate data backup practices and recovery plan in place, it is possible to recover data after a system failure.

Aside from data backup, we should also employ other means to safeguard our data to reduce the risk of a data loss. The following is a list of hardware redundancy that we recommend:

- ▶ Redundant array of inexpensive disk (RAID) devices
- ▶ Dual access paths
- ▶ Dual I/O controllers
- ▶ Dual power supplies
- ▶ Backup or standby processors
- ▶ Uninterruptable power supplies

None of these on their own can guarantee the availability of data. In combination, however, they can reduce the chance of a data failure.

There are many types of backups. Before considering what type of backup strategy to deploy, you have to perform an assessment on the user requirement as well as the environment setup. For example, if you prefer a hot backup site, investment of another server of comparable functionality is necessary.

Factors that need to be considered when defining the requirements for a backup strategy include:

- ▶ Nature of failure
- ▶ Time for recovery
- ▶ Backup types
- ▶ Backup windows
- ▶ Space required for backup and recovery
- ▶ Recovery points

1.1.1 Nature of failure

Your system might fail due to many causes. They include human errors, hardware failures, transaction failures, and disasters. In this section, we briefly explain what the causes are, how to handle them, and how to plan ahead.

Human factor

One of the most common causes of system failure is human error. For example, if an user accidentally deletes an application group in OnDemand, this would

trigger the removal of all data loaded into the application group. Although we might recover most of the data by restoring from available backups, some data might still be lost. This could be due to the fact that not every component of OnDemand was backed up when there were changes to the OnDemand system. Moreover, a lot of effort will be needed to recover from such errors and various types of backups, such as cache and DB2 database will be needed for restoration.

A proactive way is to minimize the risk of such incidents happening. For example, we can restrict the number of users who can perform deletion tasks. In addition, we can assign every user with a unique user ID and enforces users not sharing a common password, so that they will be solely responsible for their own action.

On the operating system level, user with the root access or system administrator authority could delete important data such as a database file. This could be prevented by restricting access of these users to the operating system.

OnDemand has incorporated some capabilities to reduce the risk or impact of human errors. This includes allowing you to setup user types with different authority and application groups with different level of permissions to restrict access.

Hardware failure

Hardware failure can happen to any part of the system. This includes the physical machine with the processors, memory, I/O, and all the peripherals such as storage devices. As a system administrator, always look out for errors in the error report and resolves the problem accordingly. For AIX, you can check `errpt`; for Windows, you can check the event logs. In the case of an iSeries server, the system operator message queue, job logs, system logs, and history logs contain important information to help you identify and correct failures.

Because OnDemand is an archival and retrieval system, there is a high demand for writing to and reading from the storage media. This increases the chance of media failure. TSM is commonly used to manage data for OnDemand. There is facility in TSM to recover media that is physically damaged or destroyed. When using TSM, you need to create a backup copy of the storage pools and perform backup storage pool regularly.

Note: Although OnDemand for iSeries does not currently support TSM, the iSeries operating system provides facilities for duplicating removable media such as optical volumes.

High availability implementation helps to contain hardware outages by reducing or eliminating many single point of failures. It does so by duplicating many

hardware resources which can be utilized in case the primary resources fail. We will discuss more on this topic in 1.2, “High availability” on page 11.

Transaction failure

If something happened to the server during an update of a transaction such as data loading to an OnDemand system, the database could be left in an inconsistent and unusable state. In this case, the data will need to be corrected to a consistent and usable state before users can access the OnDemand system again.

The database log files help correct this type of failure by allowing the transactions received before the failure to either be reapplied to the database or to be rolled-back. Rolling-back transactions is a way of returning the database to the state it was in before the failed transaction. OnDemand for iSeries uses the iSeries’ journaling feature to achieve this.

Disaster

There are disasters that would completely disrupt your day-to-day operations on your production site. They include:

- ▶ Hurricane
- ▶ Fires
- ▶ Accidents
- ▶ Earthquakes
- ▶ Terrorist attacks

Disaster normally involves a complete system failure in the entire location. Depends on how much data is lost and the provision for backup, different levels of recovery can be achieved. For example, if the data source can be reproduced easily within a week of its generation, and if backup is done weekly, the data lost would be minimized because the lost data can be easily reloaded. Note, in this scenario, the arrangements should be made to create the backups to a removable medium and store them off-site every week. We will discuss this further later in 1.3, “Disaster recovery” on page 16.

1.1.2 Time for recovery

The actual time required for a recovery depends on a number of factors: some are fixed (for example, the time needed to restore data from storage media), some are outside of the administrators’ control (for example, hardware might need to be repaired or replaced).

The recovery time objective (RTO) is the requirement for restoration of systems and applications, expressed in terms of how long a business can afford to have

systems and applications down after a disaster. For example, a business that could afford to be without systems for 8 hours has a RTO of 8 hours.

Important: It is very important to *set user expectations realistically* during the planning time so that everyone understands and agrees the estimated time it will take to recover a system after a disaster occurs.

To reduce the recovery time, there are several actions you can take:

- ▶ Document and backup system configuration files.
- ▶ Estimate the time required to execute the recovery procedures. This should include the time involved in identifying the problem and solution, and the time to restore the data from the storage media.
- ▶ Prioritize the recovery procedures. Generally, users have more tendency to look at the recent data; therefore, if both the cache and other storage method are used, restore the cache first. Users can access the data in cache while the rest of the data are being restored.
- ▶ Document the recovery procedures for different situations.
- ▶ Develop a strategy that strikes the right balance between the cost of backup and the speed of recovery.

OnDemand system configuration files

As mentioned earlier, one of the actions to take to reduce the recovery time is document and backup system configuration files. OnDemand uses DB2 and optionally TSM. Both of them have their own databases and their own commands to backup. There are configurations, however, are not backed up using the standard database and TSM backup commands.

The OnDemand system information and configuration files that should be documented and backed up are:

- ▶ OnDemand configuration files
- ▶ TSM configuration files
- ▶ DB2 configuration
- ▶ Operating system's password files
- ▶ Environment configurations and configuration files
- ▶ Network configuration

Note, only the first item from the list, OnDemand configuration files, is applicable to OnDemand for iSeries. The rest are automatically saved when a full system save (using Option 21 from the SAVE menu) is used. Again, TSM is currently not supported by OnDemand for iSeries.

These are external files that are not part of the databases, but they need to be accessible for reading, or even editing, when the system is down. The backup strategy must ensure that these files and configurations are also backed up using the operating system commands or tools such as TSM. We should also print them out on hard copies.

1.1.3 Backup types

There are two types of backup: online backup and offline backup. During backup, there are also full backup and incremental backup. You can have a combination of full online backup, full offline backup, incremental online backup, and incremental offline backup. In addition, depend on the data type, you can perform different types of backup for different types of data.

The decision as what type of backup to use and how often to perform the backup depend on factors such as how much data you have and the backup window you have.

Offline backup versus online backup

An *offline backup* is done when the system is not in operation and is inactive (quiescence). Users cannot connect to an application or the underlying database to perform any actions such as query; there will be no activities on the system except the backup process.

An *online backup* refers to a backup that is performed when the system is still in full operation. Users can access the application in question or the underlying database to perform normal actions such as data update and retrieval; the system is running as usual.

Full backup versus incremental backup

When you backup either online or offline, you can perform full backup or incremental backup.

A *full backup* is backing up of the complete data set. You can perform a full (or complete) backup of the all the databases involved in an application. You can also perform a full backup of all the files involved in the application.

An *incremental backup* is backup up of the changed data set. You can perform an incremental database backup that backups only the changed data since the last incremental backup or since the last full backup. You can also perform incremental (partial) backup of the files that have been changed since the last system's incremental or full backup.

1.1.4 Backup window considerations

A *backup window* is a period of time when a specific type of backup can be performed. Depends on the type of backup that needs to be done, in general, we recommend to schedule the backup window when system has the least activity or no activity.

In an OnDemand system, there is a lot of data that need to be backed up. Because some backups may take a longer time than others, we need to deploy different method to suit the different nature of backup. Some backups can be performed online while other backups should be done offline. For example, configuration files are usually static and can be backed up at anytime, offline or online. The window for this type of backup is limitless. On the other hand, because the database is always active, it requires special commands to backup in order to keep its integrity. In this case, we need a window during the off peak hours to perform this type of backup.

OnDemand database backup can be taken while the database is either online or offline. Although it is possible to perform online backup while the system is operational, you need to ensure that any load on processors, networks, or the storage manager libraries caused by the backup process does not result in performance or response degradation that is unacceptable to the end users. For example, if DB2 backup is done via TSM on the same tape library as the OnDemand object data, and at the same time, there are users accessing the system, then there will be less drives during the backup time to service user request. Depending on the number of drives in the tape library, an out of drives situation might occur if a lot of users are trying to retrieve data from the library. The situation could be worse if data is being loaded into OnDemand and needs to write into the same TSM libraries.

Note: Note, the only consideration for online backup on iSeries is that a Save-While-Active synchronization point is reached within the specified time. Backup issues related to TSM do not apply to iSeries.

Planning for a good backup window is important. The best time to perform a backup is when the system has least activity. For instance, database loading, database expiration, and runstat should not be active during the backup. At the same time, you must consider the time the backup is going to take and the window available for the backup.

1.1.5 Space requirement

Space is needed, whether in the hard disk or the storage libraries, to hold the backup copies and archived log files.

Depending on where the backup is, during restoration, you might need to allocate disk space to hold the backup copy of the database and the restored database. If transactions is needed to roll-forward, extra space is required to hold the backup copy of the database, the restored database, and all of the archived log files created between backup copies of the database. This is especially true for incremental or online backup.

Note: This does not apply to iSeries.

1.1.6 Recovery points

How near can the database be recovered to the time of failure? The *recovery point* is the point in time of the data which can be recovered. This could be the last transaction that the database has performed or the last backup of the data. You need to define recovery points from the beginning and have concurrence with the business users.

The recovery point objective (RPO) is the requirement for currency of data. It is also expressed in the amount of data that could acceptably be recreated after a disaster. For example, a business that could afford to lose 5 minutes worth of data has a RPO of 5 minutes.

Whatever your situation is, you need to consider recovery points and define a policy that is both achievable and acceptable to your user community. Sometimes, it could be too expensive and impractical to recover to the closest time.

In many situations, it may not be correct to simply restore databases from their latest backups; there may be damaged data that could not be restored to that point in time. In order to *maintain data integrity across the entire system*, it may be necessary to restore data from earlier backups so that all data can be properly restored at the same point in time.

Recovery of partial or full database

Database backup can be performed on the tablespace level or the full database level. With a tablespace backup, you can specify one or more tablespaces to be backed up rather than the entire database. If you specify tablespace backup, you can restore the selected tablespaces to a state identical to the time the backup was made. However, those tablespaces not selected at the time of the backup will not be in the same state as those that were restored.

Note: This does not apply to iSeries.

Backup of the operating system

It is a good practice to backup the operating system so that in case of need you have a good starting recovery point. This is especially true during the disaster recovery scenarios.

In AIX, this is done using the `mksysb` command. In Windows, you can use ghost or other similar software. With the operating system backup, you could restore all the installed software and configuration into a new machine without having to look for various installation media and go through the multiple software installation process. By performing tablespace level restore, it could probably shorten the RPO time as compare to a full database level restore.

In the case of iSeries, the full system save (using Option 21 from the SAVE menu) will save all components of the operating system and all user data.

1.2 High availability

Availability is a measure of the time that a server or process is functioning normally, as well as a measure of the time the recovery process requires after a component failure. It is the *downtime* that defines system availability. Availability requires that the system provides some degree of redundancy in order to eliminate single points of failure (SPOF). Vertical scalability provides redundancy by using multiple processes; however, the physical machine is still a single point of failure. For this reason, a high availability system topology typically involves horizontal scaling such as redundancy across multiple machines.

The concept of *high availability* roughly equates to a system and its data available almost all the time, 24 hours a day, 7 days a week, and 365 days a year. We know that 100% availability is not a cost-effective reality today for the large majority of implementations; rather, it is a goal. Our goals are to design and to build systems that are highly available by minimizing both planned and unplanned outages that can be caused by single points of failure.

Many organizations need almost continuous availability of their mission-critical applications and server resources. Loss of service (also called an *outage*) of an important application often translates directly into lost revenue or business.

1.2.1 Planned versus unplanned outages

Outages can be broadly classified into two categories, planned and unplanned outages. The *planned outages* take place when the operation staff takes a server offline to perform backups, upgrades, maintenance, and other *scheduled* events. The *unplanned outages* occur due to *unforeseen* events such as power loss, hardware or software failure, human errors, security breaches, or natural

disasters. As a result, a system downtime can be caused by both planned and unplanned events.

When addressing high availability, we concentrate on how to minimize the unplanned downtime, because nobody knows when the unplanned downtime occurs and businesses require their systems to be up during normal operating hours.

Failure as a result of unplanned outages may spread across the different components that make up an OnDemand system. The components include hardware, software, and applications. It is important to understand that a redundant hardware-only solution would clearly help to prevent unplanned outages, but a true, highly available system must take into account of all the components of an end-to-end solution. You must also consider the human factor, which can be a major contributor to downtime. Although education is important, it is perhaps even more important to design easy-to-use system management facilities with well-documented and executed policies and procedures to help minimize any potential human factors contributing to downtime.

1.2.2 High availability versus continuous availability

Confusion often occurs between high availability and continuous availability. High availability is a component of continuous availability. High availability focuses on reducing the downtime for unplanned outages. *Continuous operations* focus on providing a “never stop” set of applications. The sum of high availability and continuous operations equals to *continuous availability*. Figure 1-1 shows these concepts graphically.

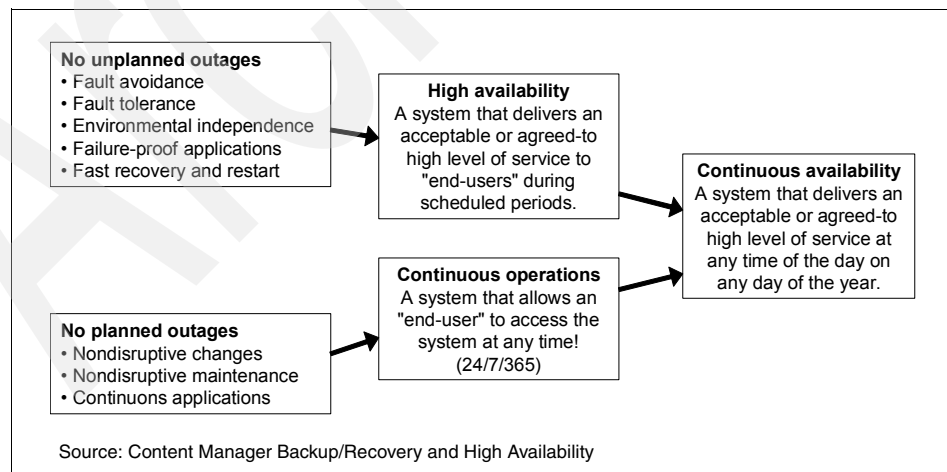


Figure 1-1 High availability and continuous availability

1.2.3 Levels of availability

There are many different levels of availability. When considering to what extent or level your system should be made available, it is not only important to balance downtime with costs, it is also very important to evaluate the damage to the organization if the service is temporarily unavailable.

Several levels of high availability can be deployed. The objective once again should be to provide an *affordable level* of availability that supports the business requirements and goals.

In OnDemand, there are several components that need to be taken into consideration when designing an end-to-end high availability system. Some of the components are:

- ▶ Library Server and its database
- ▶ Object Servers
- ▶ TSM process and its database
- ▶ Disk subsystem
- ▶ Tape and optical libraries
- ▶ Operating system processes

Note: For iSeries, we do not distinguish between Library Server, database, and Object Server. TSM is currently not supported by OnDemand for iSeries.

To meet business requirements, the technologies used to achieve high availability must be weighed against the cost for a particular implementation. There are several levels of technologies that can be deployed today to achieve high availability for systems in general. Figure 1-2 on page 14 depicts some of the more commonly used technologies to achieve high availability.

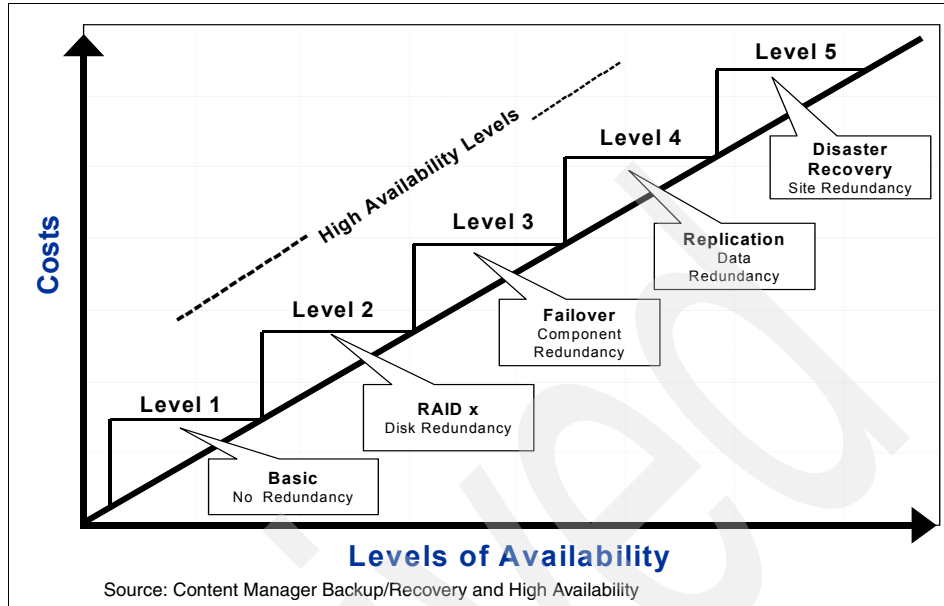


Figure 1-2 High availability tiers

As shown in Figure 1-2, there are five levels of availability:

- ▶ Level 1: Basic systems, no redundancy
- ▶ Level 2: RAID x, disk redundancy
- ▶ Level 3: Failover, component redundancy
- ▶ Level 4: Replication, data redundancy
- ▶ Level 5: Disaster recovery, site redundancy

Level 1: Basic systems, no redundancy

Although backup would most likely be taken on a regular basis, *basic systems* do not employ any special measures to protect data or services. When an outage occurs, support personnel need to restore the system from backup (usually tape). We will discuss this topic in more details in Chapter 3, “Backup and recovery for OnDemand Multiplatforms” on page 51, Chapter 8, “Backup and recovery for OnDemand iSeries” on page 193, and Chapter 12, “Backup and recovery for OnDemand z/OS” on page 251.

Level 2: RAID x, disk redundancy

Disk redundancy, either RAID or disk mirroring, is used to protect data against the loss of a disk. Technologies have also improved the reliability of hard disks. New storage devices such as IBM Enterprise Storage Server® (ESS) provide excellent protection and at the same time gives very good performance.

Level 3: Failover, component redundancy

Most systems have multiple components that may become a single point of failure (SPOF). An outage in any single component can result in service interruption to end users. Multiple instances or redundancy for any single component should be deployed for availability purposes. OnDemand does not provide any functionality to perform clustering natively. It is possible, however, to use software such as HACMP to protect an OnDemand system against server failure.

Note: i5/OS (OS/400) and z/OS do not use HACMP.

We discuss failover strategies and techniques for OnDemand implementations in later chapters of this redbook.

Level 4: Replication, data redundancy

This type of high availability for OnDemand implementations extends the protection by duplicating the database content (metadata and control tables) and file system content to another machine (server) in the event of a hardware, software, disk, or data failure. This would provide a higher level of protection and high availability in the event of a failure compared to a shared disk/failover strategy previously discussed. This type of a high availability implementation (replication) can also be used as a disaster recovery strategy. The difference is whether the servers are located within the same location or are geographically separated. We discuss this topic in more detail in later chapters of this redbook.

Level 5: Disaster recovery

To handle disasters such as tornados, hurricanes, and tsunamis that disrupt the entire business operation at a physical location, the highest level of availability, disaster recovery, implies that we maintain systems in different sites. When the primary site becomes unavailable due to disasters, the backup site becomes operational within a reasonable time. This can be done manually through regular data backups stored *off site*, and automatically by geographical clustering, replication, or mirroring software. We will discuss this topic further in 1.3, “Disaster recovery” on page 16.

A high availability solution normally does not just use one specific technology. The solutions can incorporate a variety of strategies and technologies. It is common to combine multiple high availability levels within a single solution. For example, we can design a system to have a failover (level 3) strategy for the OnDemand database, with disk redundancy strategy (level 2) and a disaster recovery server (level 5) at another location.

1.2.4 Measuring availability

Availability can be subjective. To a user who only uses the system 8 hours a day, the user would consider the system is 100% available if the system is up and running during this 8 hour time frame, even if this system is shutdown every night for routine maintenance. To a user who accesses the system 24 hours a day, the availability of the same system would not be considered as 100%. In addition, because of the multiple components involved to make a system functional, system availability is also a combination of the availability of all these component (such as the hardware, the network, the server, and the database).

The *total availability* of a system can be calculated by multiplying the availability of each component. For example, if a system is comprised of three components, one has an availability of 98.5%, one has 98.5%, and the last has 99.5%, then the total availability of the system is $98.5\% \times 98.5\% \times 99.5\%$. It is important to balance the availability of all components in a production environment to increase availability as perceived by the end users. This means that true availability is the product of the components or the weakest link comprising the end-to-end solution.

Availability is not free. There is a lot of work involved to integrate the many diverse components, people, and processes into a stable, highly available system and it needs serious management attention. High availability starts with reliable products and relies on an infrastructure and application design that includes availability techniques and careful system integration. A lack of, or failure to follow, careful management and effective systems management procedures is one of the most common causes of an outage. Effective management systems that employ defined and repeatable processes contribute significantly to higher levels of availability, and in the long run, decrease the cost of Information Technology (IT) services through more effective and efficient use of IT resources.

In later chapters of this redbook, we describe ways to improve system availability.

1.3 Disaster recovery

Every company should have a disaster recovery plan. In this section, we discuss the plan, the disaster recovery objectives and their benefits.

1.3.1 Disaster recovery plan

Disaster recovery is becoming an increasingly important aspect of enterprise computing. This is because interruption of service or loss of data can have serious financial impact, whether directly or through loss of customer confidence.

A *disaster recovery plan* (DRP), sometimes referred to as a *business continuity plan* (BCP), or *business process contingency plan* (BPCP), describes how an organization is to deal with potential disasters. Just as a disaster is an event that makes the continuation of normal functions impossible, a disaster recovery plan consists of the precautions taken so that the effects of a disaster will be minimized, and the organization will be able to either maintain or quickly resume mission-critical functions.

As devices, systems, and networks become ever more complex, there are simply more things that can go wrong. As a consequence, recovery plans have also become more complex. Typically, disaster recovery planning involves an analysis of business processes and continuity needs; it can and should also include a significant focus on disaster prevention.

Regulatory compliance is the main reason for implementing a disaster recovery plan. In the last 30 years, the legal requirements for data protection and recovery have been consistently growing to involve today most business sectors. In addition, the demand for business continuity has greatly increased in the past two years. Major catastrophic events have been a “wake up call” for the entire business community and, in particular, for the public sector. The trend of BCP-DRP started in the late 1990s with the preparation for the end of the millennium. Fixing the year-2000 bug prompted many organizations to rethink their systems in terms of how best to protect their critical data in case of a system failure on December 31, 1999 at 12:00 midnight. Distributed work environments and transaction-based processing are the types of activities that usually demand the highest level of data protection and BCP-DRP.

1.3.2 Objectives and benefits

Disaster recovery planning varies from one enterprise to another, depending on variables such as the type of business, the processes involved, and the level of security needed. Disaster recovery plans can be developed within an organization or purchased as a software application or a service. It is not unusual for an enterprise to spend 25% of its information technology budget on disaster recovery planning and testing.

The purpose of DRP is to provide users continual access to the system or data should some major, catastrophic event or disruption which threaten the operation or deny access of the production system. A DRP does not usually address minor disruptions that do not require relocation.

There are additional benefits through implementation of disaster recovery solution. To recover as much data as possible, people will need to rethink the operational process, the actual storage device where the business data is stored. Is there any way to streamline the operational process? How can data be easily

replicated to a different site? Or will there be a need to move data to a more reliable storage device with capability of duplicating data to another location? Making use of more streamlined processes, including more reliable storage devices, will also improve efficiency and reduce down time due to storage failure.

It is possible to simplify storage management by using a global strategy (holistic approach) to address common disaster recovery requirements across the organization. Following this principle, all applications and systems (departmental and organization-wide) should comply with the same disaster recovery requirements and guidelines based on the different type of data processed by those systems.

1.3.3 The seven tiers of disaster recovery

In 1992, the SHARE user group in the United States, in combination with IBM, defined a set of disaster recovery tier levels. This was done to address the need to properly describe and quantify various methodologies for successful mission-critical computer systems' disaster recovery implementations. Accordingly, within the IT Business Continuity industry, the tier concept continues to be used, and it is very useful for describing today's disaster recovery capabilities.

The seven tiers of disaster recovery solutions offer a simple methodology of defining your current service level, the current risk, the target service level, and the target environment. Figure 1-3 on page 19 shows the seven tiers of disaster recovery solutions in terms of cost and time to recovery:

- ▶ Tier 0 - No off-site data
- ▶ Tier 1 - Data backup with no hot site
- ▶ Tier 2 - Data backup with a hot site
- ▶ Tier 3 - Electronic vaulting
- ▶ Tier 4 - Point-in-time copies
- ▶ Tier 5 - Transaction integrity
- ▶ Tier 6 - Zero or little data loss
- ▶ Tier 7 - Highly automated, business-integrated solution

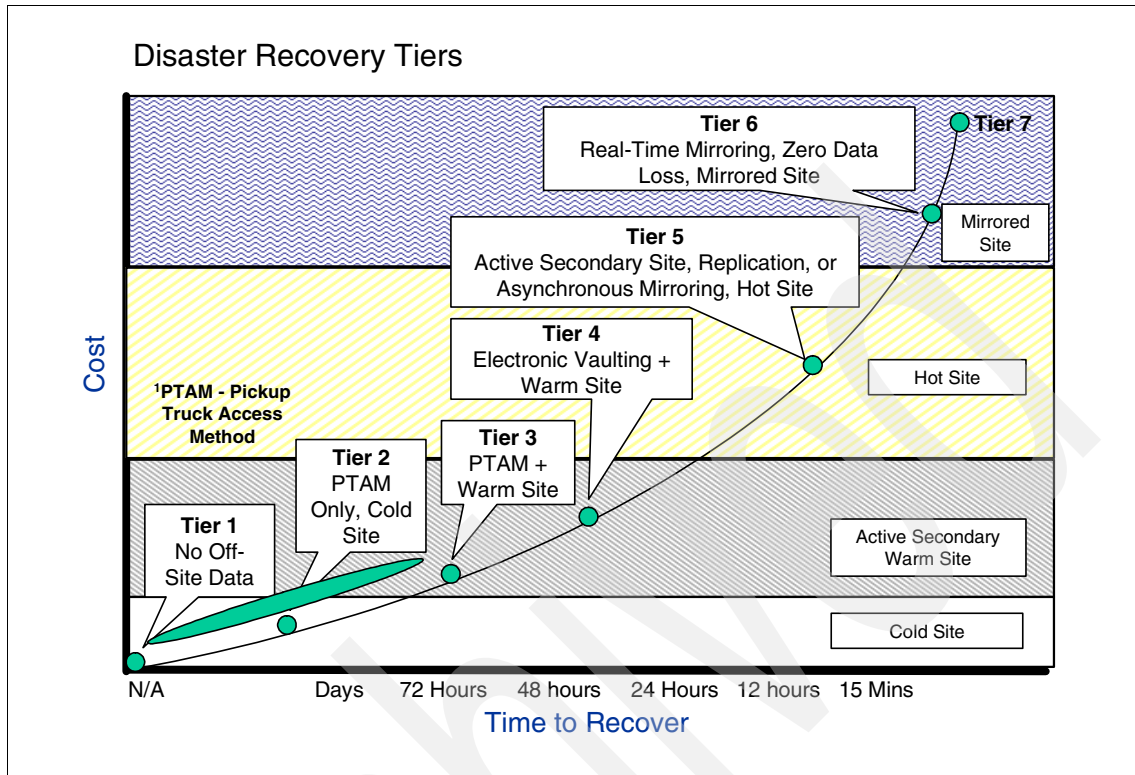


Figure 1-3 Seven tiers of disaster recovery solutions

Tier 0 - No off-site data

Businesses with a tier 0 disaster recovery solution have no disaster recovery plan. There is no saved information, no documentation, no backup hardware, and no contingency plan.

The length of recovery time in this instance is unpredictable. It may not be possible to recover at all.

Tier 1 - Data backup with no hot site

Businesses that use tier 1 disaster recovery solutions back up their data at an off-site facility without a hot site. There is no facility to restore the system. Depending on how often backups are made, the company must be prepared to accept several days to weeks of data loss. Although the backups are secure off-site, this tier lacks the systems on which to restore data.

Examples of tier 1 disaster recovery solutions include PTAM¹, disk subsystem or tape-based mirroring to locations without processors and IBM Tivoli Storage Manager.

Tier 2 - Data backup with a hot site

Businesses using tier 2 disaster recovery solutions make regular backups on tape. This is combined with an off-site facility and infrastructure (known as a *hot site*) in which to restore systems from those tapes in the event of a disaster. This tier solution will still result in the need to recreate several hours to days worth of data, but it is more predictable in recovery time.

Examples of tier 2 disaster recovery solutions include PTAM with hot site available and IBM Tivoli Storage Manager.

Tier 3 - Electronic vaulting

Tier 3 solutions utilize components of tier 2 and some mission-critical data is electronically vaulted so that the backups are electronically transmitted to a secure facility. This electronically vaulted data is typically more current than that which is shipped via PTAM. As a result, there is less data recreation or loss after a disaster occurs.

Example of tier 3 disaster recovery solutions include electronic vaulting of data and IBM Tivoli Storage Manager - Disaster Recovery Manager (DRM).

Tier 4 - Point-in-time copies

Tier 4 solutions are used by businesses that require both greater data currency and faster recovery than users of lower tiers. Rather than relying largely on shipping tape, as is common in the lower tiers, tier 4 solutions begin to incorporate more disk-based solutions. Several hours of data loss is still possible, but it is easier to make such point-in-time (PIT) copies with greater frequency than the data that can be replicated through tape-based solutions.

Example of tier 4 disaster recovery solutions include Batch/Online Database Shadowing and Journaling, Peer-to Peer Remote Copy Extended Distance (PPRC-XD), FlashCopy®, FlashCopy Manager, Peer-to-Peer Virtual Tape Server, Asynchronous Cascading PPRC, IBM Tivoli Storage Manager - Disaster Recovery Manager, eRCMF, and iSeries IASPs with FlashCopy.

¹ Pick-up Truck Access Method, shipping of backup tapes from production data centers to backup facilities via ground transportation

Note: OnDemand for iSeries currently does not support TSM. OnDemand for iSeries currently does not support IASP. For tier 4 and tier 6, journaling is applicable to iSeries. For more information about journaling, refer to 9.2, “Journaling” on page 216 and 9.3, “Remote journaling” on page 220.

Tier 5 - Transaction integrity

Tier 5 disaster recovery solutions are used by businesses with a requirement for consistency of data between production and recovery data centers. There is little to no data loss in such solutions; however, the presence of this functionality is entirely dependent on the application in use.

Example of tier 5 disaster recovery solutions include software and two-phase commit.

Tier 6 - Zero or little data loss

Tier 6 disaster recovery solutions maintain the highest levels of data currency. They are used by businesses with little or no tolerance for data loss and who need to restore data to applications rapidly. These solutions have no dependence on the applications to provide data consistency.

Example of tier 6 disaster recovery solutions include Peer-to Peer Remote Copy (PPRC), XRC, GDPS/PPRC Storage Manager, Peer-to-Peer VTS, Asynchronous Cascading PPRC, PPRC Migration Manager, eRCMF, GeoRM, AIX Logical Volume Mirroring, and iSeries IASPs with PPRC.

Tier 7 - Highly automated, business-integrated solution

Tier 7 disaster recovery solutions include all the major components being used for a tier 6 solution with the additional integration of automation. This allows a tier 7 solution to ensure consistency of data above that of which is granted by tier 6 solutions. Additionally, recovery of the applications is automated, allowing for restoration of systems and applications much faster and more reliably than would be possible through manual disaster recovery procedures.

Example of tier 7 disaster recovery solutions include GDPS/PPRC, GDPS/XRC, GDPS/PPRC with Open LUN Management, GDPS/PPRC with HyperSwap™, HACMP/XD, ESS support of GDS for MSCS, and iSeries High Availability Business Partner software.

1.3.4 Trends in disaster recovery planning

Large organizations are beginning to show more interest in disaster recovery. Skill transfer and training are usually underestimated. Simulations or rehearsals are needed for gap identification. Spending, however, remains moderate.

New facts about business continuity

Business are setting the service-level bar higher. Some of the new facts about business continuity include:

- ▶ Traditional 72-hour recovery periods for business-critical processes are no longer good enough.
- ▶ A new 4 to 24 hour recovery time and recovery point objectives are generally used.
- ▶ A need for a larger goal of ensuring resumption and recovery of end-to-end enterprise business processes.
- ▶ Active/passive configuration between two sites for 30-60 minute recovery.
- ▶ The 24x7 continuous availability being designed into most critical applications.

Geographic diversity is imperative

One of the new trends is the goal of geographically diversifying the primary and the backup sites and different infrastructure components. The 9/11 experience expanded the definition of “significant” distance between sites. The U.S. Securities and Exchange Commission (SEC) is now mandating same day recovery for firms that play critical roles in financial markets. Other compliance regulations favor a network approach.

The physical distance between the sites can be categorized as follows:

- ▶ Campus/local: Distance is less than or equal to 10 kilometers (km).
- ▶ Medium area network (MAN): Distance is between 80 to 100 kilometers.
- ▶ Wide Area Network (WAN): Distance is greater than or equal to 1,000 kilometers.

1.4 Business continuity strategies and options

Business continuity is a management process that relies on people facilities, business processes, infrastructure and applications to sustain operations at all times and under any circumstances. A business continuity plan focuses on sustaining an organization’s business functions during and after a disruption. A business continuity plan (BCP) may be written for a specific business process or may address all key business processes. The IT infrastructure addressed by the business continuity plan is only based on its support for business processes.

A business continuity plan is a combination of backup, recovery, high availability as well as a disaster recovery plan. When working on a business continuity plan,

you need to consider the cost of implementing a solution, the design of the solution, and other critical considerations and success factors.

1.4.1 Cost versus loss

When designing a solution, we need to balance the cost of implementation versus the loss incurred should the system become unavailable. Although highly available system is desirable, we need to strike a optimum balance between the cost of availability and the cost of unavailability. It is crucial to understand the business impact of a failure and what the loss to the business in case of an unplanned outage. The consequences of outages for different businesses varies. For financial services, there could be loss of money and business for each minute of downtime during business hours. For some other business, however, it could merely means inconvenience to users; it might have cost more to implement the high availability or disaster recovery solution. Other than lost of business or money, sometimes the lost is intangible. This includes a company's reputation and customer confidence.

The bottom line is, always invest wisely and effectively to minimize the outages. Figure 1-4 on page 24 shows when the investment in availability could reach the point of diminishing returns.

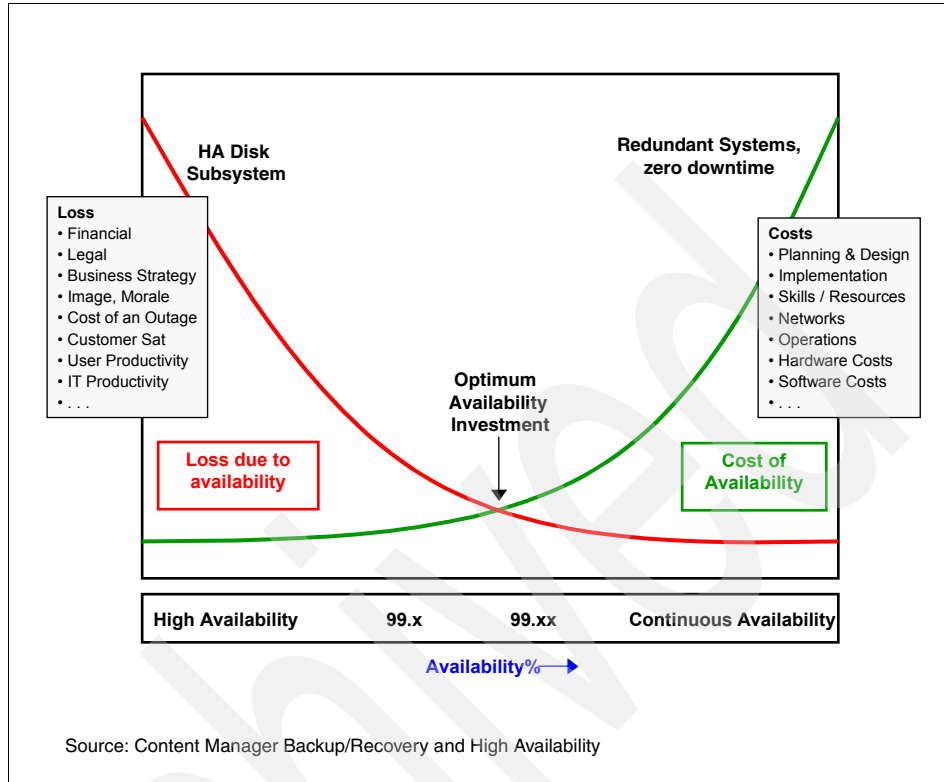


Figure 1-4 Cost versus lost

At some point in Figure 1-4, the cost of availability and the loss due to availability cross to reach an optimum availability investment point. The optimum availability investment point can be a difficult equation to calculate with quantifying the losses being the more challenging of the two variables. The idea is to strive for an optimum balance by designing cost-effective, high availability solution to support the business environment. Also, the cost of implementation becomes increasingly expensive as you approach higher continuous availability (100%).

Data protection cost

The additional cost associated with data protection varies from one organization to another and from one type of solution to another. The key questions to address when planning for business continuity are:

- ▶ How much data can the organization afford to lose?
- ▶ What is the organization's recovery point objective (RPO)?
- ▶ How long can the organization afford to have the system offline?
- ▶ What is the organization's recovery time objective (RTO)?

Factors contributing to higher costs

There are many factors contributing to higher costs. They include:

- ▶ More complex IT operating environment as a result of exponential growth of storage capacity and diversification of operating systems (strong growth of Windows NT® in the last two decades and the emergence of Linux® in the late 1990s).
- ▶ The new era of e-business requires solutions operating at 24 hours a day, 7 days a week, 365 days a year (7x24x365), and on a more global basis of information exchange between dispersed sites.
- ▶ Digital data continues to expand dramatically, and more data is critical to the business.
- ▶ Increased complexity with new data protection and regulatory requirements.

1.4.2 Solution design

In designing a suitable solution, the following factors need to be considered:

- ▶ Categorize requirements.
- ▶ Identify critical applications and data.
- ▶ Determine cost of downtime.
- ▶ Develop solution with need and cost in mind.
- ▶ Implementation time.
- ▶ Provision for periodic testing.
- ▶ For disaster recovery, compliance to data center disaster recovery strategy within overall corporate business continuity objectives.
- ▶ Which tiers of the 7 tiers in disaster recovery would you like to follow?
- ▶ In High availability, which of the 5 levels of availability would you like to achieve?

As discussed earlier, there can be a combination of high availability and disaster recovery. Keep in mind the cost would go up as well. One strategy may be using of the high availability server or the disaster recovery server to handle distributed workload for your current system.

1.4.3 Critical considerations and success factors

When planning for business continuity, you should consider the following (although they may not directly relate to OnDemand):

- ▶ Information-based business model

This business model is highly digital versus paper-based business. 24x7 access to information is critical. Security of information is critical. Some OnDemand customers can be grouped under this category.

- ▶ Transaction-based (versus batch-based) processing
This implies the operations stop when IT is down.
- ▶ Distributed work environment
This implies a mobile workforce. Remote offices need to connect to headquarters.
- ▶ People-based business
This implies productivity of employees is closely tied to access to technology and information. This applies to customers who use OnDemand in a call center.

Success factor of a business continuity plan

One of the main success factors of a business continuity plan is a holistic approach to solution design; it puts emphasis on integrating islands of automation. For example, an OnDemand system has a very good disaster recovery plan, but there is no plan for the host system which originates the data for the OnDemand system. In case of a disaster, although OnDemand can be recovered, the company still cannot function due to lack of disaster recovery plan for the host system. Another point to consider is that it might be more cost effective to have a disaster recovery plan for selected group of important systems in the company than devising a disaster recovery plan for each of the systems.

We will discuss more on the strategies and options used for high availability and business continuity in later chapters of this redbook.

Content Manager OnDemand overview

In this chapter, we provide an overview of the IBM DB2 Content Manager OnDemand (OnDemand) system. Note, this is a general description of OnDemand. It is mostly applicable to multiplatforms. When different from other platforms such as iSeries, we add special notes to inform you of the differences.

The following topics are covered in this chapter:

- ▶ OnDemand features and functions
- ▶ System overview
- ▶ OnDemand terminology and concepts
- ▶ Database management
- ▶ Storage management
- ▶ Report migration and removal
- ▶ Tivoli Storage Manager

2.1 Introduction

IBM DB2 Content Manager OnDemand (OnDemand) supports any organization that can benefit from hard copy or microfiche replacement and instant access to information. An OnDemand system can support small office environments and large enterprise installations with hundreds of system users. OnDemand can dramatically improve productivity and customer service in many businesses by providing fast access to information stored in the system.

OnDemand processes the print output of application programs, extracts index fields from the data, stores the index information in a relational database, and stores one or more copies of the data in the system. OnDemand can archive newly created and frequently accessed reports on high speed, disk storage volumes and automatically migrate them to other types of storage volumes as they age.

OnDemand fully integrates the capabilities of Advanced Function Presentation™ (AFP™) including management of resources, indexes, and annotations, and supports full fidelity reprinting and FAXing of documents to devices attached to a PC, OnDemand server, or other server in the network. OnDemand provides administrators with tools to manage OnDemand servers, authorize users to access OnDemand servers and data stored on the server, and backup the database and data storage.

OnDemand provides users the ability to view documents, print, send and FAX copies of documents, and attach electronic notes to documents.

2.1.1 OnDemand features and functions

OnDemand provides an information management tool that can increase the users' effectiveness when working with customers. It does the following:

- ▶ Integrates data created by application programs into an online, electronic information archive and retrieval system.
- ▶ Provides the controlled and reliable access to all of an organization's reports.
- ▶ Retrieves data that is needed when it is needed.
- ▶ Provides a standard, intuitive client with features such as thumbnails, bookmarks, notes, and shortcuts.

Using OnDemand, there are advantages of:

- ▶ Easily locating data without specifying the exact report
- ▶ Retrieving the pages of the report that are needed without processing the entire report.

- Viewing selected data from within a report.

These features mean that OnDemand can help customers quickly retrieve the specific page of a report that they need to provide fast customer service.

2.2 System overview

OnDemand consists of client programs and server programs that communicate over a network running the TCP/IP protocol, a database manager that maintains objects, index data, and server control information, and storage managers that maintain documents on various types of storage devices. See Figure 2-1 for the OnDemand distributed system architecture.

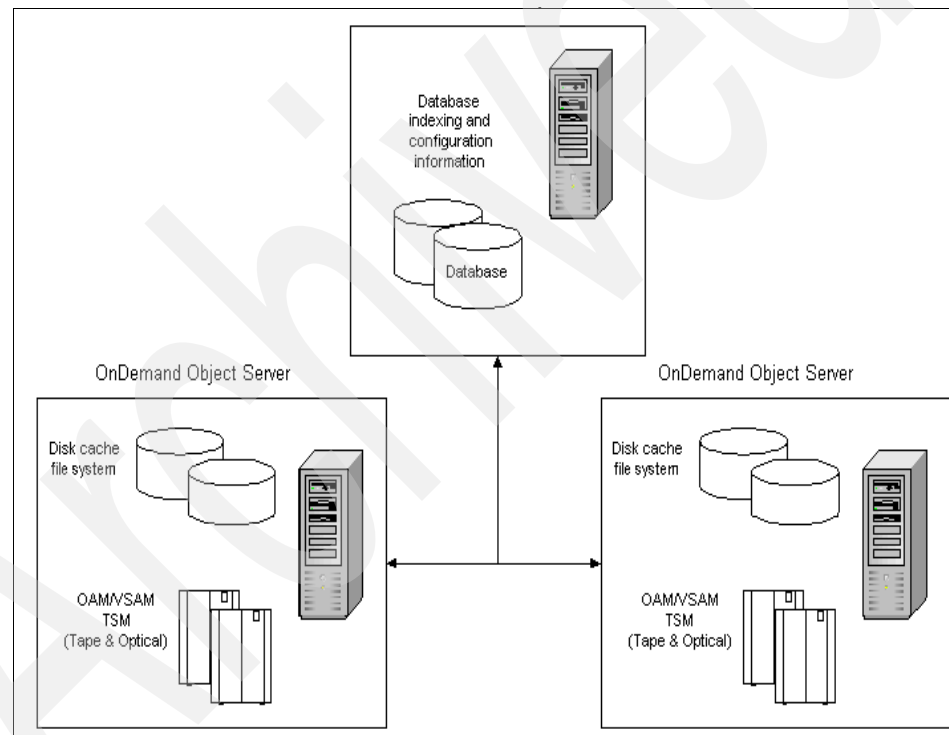


Figure 2-1 OnDemand distributed system architecture

An OnDemand system provides a single system image to the end-user. The end-user uses a Windows client or a Web browser (if using ODWEK) to log into the OnDemand system (which is the Library Server). The user then initiates queries. The Library Server processes these queries and returns the results of the queries to the client. When the user selects a document for viewing,

OnDemand retrieves the document from the Object Server on which the document was loaded. The client can then view, print, or perform other types of actions to the retrieved document.

Note: Figure 2-1 is not representative of an iSeries implementation. For OnDemand on iSeries, there is *no separate* Library Server and Object Server, although it occurs technically under the cover. If you work with iSeries, read this system overview section as reference only. Refer to Chapter 6, “iSeries architecture” on page 171 and Chapter 7, “OnDemand for iSeries overview” on page 183 for specific information about OnDemand for iSeries.

The Library Server manages information such as users, groups, and index information about the reports that are stored on the system. The Object Server manages the reports on disk, optical, and tape storage devices. An OnDemand system has one Library Server and one or more Object Servers. An Object Server can operate on the same workstation or node as the Library Server or on a different workstation or node than the Library Server. Figure 2-2 shows the major components of an OnDemand system including the Windows client, OnDemand Web Enablement Kit (ODWEK), Library Server, Object Server, database manager, storage manager, AG index tables, system tables, system log, and document resources.

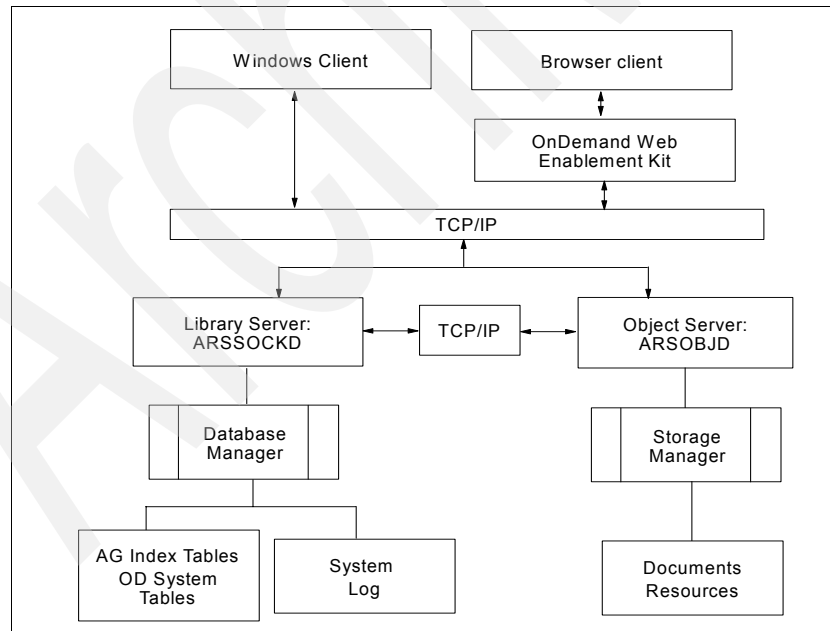


Figure 2-2 Major OnDemand system components

2.2.1 Library Server

The *Library Server* is the central component of the OnDemand system. It maintains the central database of the objects on the system. The objects it maintains include users, user groups, storage sets, storage nodes, printers, applications, application groups, and folders. The database management products that can be used with OnDemand include DB2 (provided with OnDemand), Oracle, and SQL Server (for Windows servers only).

The Library Server is an OnDemand instance, which represents a logical server environment consisting of a single system image using its own database and cache storage. Each OnDemand instance (the server, database, and cache storage):

- ▶ Has its own folders, application groups, applications, users, user groups, storage sets, and printers.
- ▶ Must run in a single code page.
- ▶ Has different security for users, groups, folder and application group permissions.
- ▶ Must have its name specified on commands if it is not the default instance.
- ▶ Has its own system log.

The standard Library Server and Object Server configuration includes the database and cache storage on one workstation or node. The Library Server processes the logins from the clients, handles the queries from the clients, and maintains the database.

2.2.2 Object Server

The *Object Server* is the component of an OnDemand system that holds the reports that are accessed by users. An Object Server belongs to an OnDemand instance. An instance is a logical server environment consisting of a Library Server, one or more Object Servers, a database, and cache storage. An Object Server:

- ▶ Has its own storage nodes.
- ▶ Must run in the same code page as the Library Server.
- ▶ Uses the security from the Library Server.

Some reasons to have more than one Object Server are:

- ▶ To distribute the storage of data across workstations or locations
- ▶ To be able to load data into more than one storage node at a time

The Object Server maintains reports that are stored in cache storage. If an Object Server is configured with archive storage, the Object Server works with Tivoli Storage Manager (TSM) on multiplatforms, a built-in storage manager on OS/400, and/or Object Access Method (OAM) or VSAM on OS/390 to manage reports in archive storage. The Object Server processes data loading operations, document retrieval requests from the clients, migration, and expiration processing.

Note: IBM i5/OS is the next generation of OS/400. Sections in this redbook may refer to i5/OS as OS/400.

2.2.3 Standard OnDemand system

The standard OnDemand system includes a database manager (such as DB2), a cache storage manager, the programs that are required to index reports and load data on the system, and ODWEK. Reports are staged on temporary storage volumes for the data indexing and loading programs.

Note: OnDemand on iSeries is installed as an s licensed program and does not have a separate database manager.

This environment is ideal for customers who do not require backup copies of data on archive media or for customers who only need to run OnDemand on a single workstation or node.

The software necessary for a standard OnDemand system are:

- ▶ **Base product package:** This includes the core system components, such as system log, enhanced ACIF indexer, generic indexer, cache storage manager, data loading, migration, expiration programs, and other data processing utilities.
- ▶ **Database manager:** This includes a database engine and is used for database administration. DB2 is included with the OnDemand base product package.
- ▶ **OnDemand clients:** There is a Windows client program for end-users, and an administrative client program for system administrators. Both are included with the OnDemand base product package. For Windows servers, the OnDemand configurator program is also included with OnDemand to maintain instances of OnDemand that run under Windows.

2.2.4 Standard OnDemand system with TSM

The standard OnDemand system with TSM provides the same functions and works with the same software as the standard OnDemand system that is described in 2.2.3, “Standard OnDemand system” on page 32.

Note: OnDemand for iSeries currently does not support TSM.

In addition to the standard system, TSM is included with the OnDemand base product package. TSM is the software that is required to maintain OnDemand data in archive storage, on media such as optical and tape storage volumes. TSM is used primarily to maintain a backup or long-term copy of OnDemand documents. TSM can also be used to maintain index data that has been migrated to archive storage and to maintain DB2 archived log files and DB2 backup image files.

This environment is ideal for customers who require backup copies of data on archive media and need to run OnDemand on a single workstation or node.

2.2.5 Distributed OnDemand system

OnDemand supports storing data on and retrieving data from more than one physical server. In a distributed environment, users submit queries to the Library Server and OnDemand retrieves documents from the Object Server on which the data is stored. Reports can be loaded on any of the Object Servers that are part of the system. The index data is always stored on the Library Server.

Note: For iSeries, the OnDemand system *must* be installed on a single iSeries server or LPAR.

The distributed OnDemand system consists of a standard OnDemand system and one or more additional Object Servers that are running on different workstations or nodes than the Library Server. Each additional Object Server requires a copy of the base OnDemand software.

This environment is ideal for customers who need to distribute the loading and accessing of reports over more than one server. The servers can reside on nodes in one physical machine, such as an SP processor, or on separate systems in different physical locations.

2.2.6 Distributed OnDemand system with TSM

The distributed OnDemand system with TSM provides the same functions and works with the same software as the distributed OnDemand system.

In addition, TSM is included with the OnDemand base product package. TSM on an Object Server is ideal for customers who want to move the archive storage part of the system off of the Library Server and for customers who need to distribute the loading and accessing of reports over more than one workstation or node. The servers can reside on nodes in one physical machine, such as an SP processor, on separate workstations, and on separate systems in different physical locations.

2.3 OnDemand terminology and concepts

An OnDemand system contains and interacts with many different components including:

- ▶ A *database manager* that maintains the index data for the reports that are loaded into the system. The database manager is a relational management product, such as DB2. The database manager resides on the Library Server.
- ▶ A *database* that contains control information about the users, groups, applications, application groups, folders, storage sets, and printers that are added to the system. The control information determines who can access the system, the folders that a user can open, and the application group data that a user can query and retrieve. The database resides on the Library Server.
- ▶ A *cache storage manager* that maintains reports and resources in cache storage. Cache storage is designed for high-speed, short-term access to the most frequently used documents.
- ▶ An *archive storage manager*, an optional component of the system, that is for the long-term storage of one or more copies of reports and resources on archive media, such as optical or tape storage libraries. TSM is an example of an archive storage manager product. TSM can also be used to maintain DB2 archived log files and backup image files. OnDemand for iSeries currently does not support TSM, but it does have its own unique archive storage manager.
- ▶ *Data indexing and conversion programs* that create index data, collect required resources, and optionally convert input data to some other format for storage and retrieval. OnDemand provides several indexing programs:
 - The AFP Conversion and Indexing Facility (ACIF) can be used to index OS/390 line data, ASCII data, and AFP files, collect resources required to view the reports, and convert line data files to AFP data.

- The OnDemand PDF Indexer can be used to create index data for Adobe Acrobat PDF files.
- The OnDemand Generic Indexer can be used to create index data for almost any other type of data that is stored on the system, such as HTML documents, Lotus® WordPro documents, and TIFF images.

The indexing programs can run on any OnDemand server. ACIF can also run on an OS/390 system. OS/400 indexer on iSeries includes the same function as ACIF and more.

- ▶ *Data loading programs* that can be set up to automatically store report data into application groups and update the database. The data loading programs can run on any OnDemand server.
- ▶ *Management programs* that maintain the OnDemand database and reports in cache storage.
- ▶ A *system log* that provides administrators with tools to monitor server activity and respond to specific events as they occur. The interface to the system log is through the system log folder and system log user exit.

Note: Again, OnDemand for iSeries does not have separate Library Server and Object Server. In addition, it currently does not support TSM.

The terms *application*, *application group*, and *folder* represent how OnDemand stores, retrieves, views, and prints reports and index data. When defining a new report or type of data to the system, an administrator must create an application and assign the application to an application group. (If an application group does not exist, the administrator must create one first.) Before users can search for or retrieve documents, an administrator must create or update a folder to use the application group and application. We discuss these terms in more details in the following sections.

2.3.1 Application

An *application* provides a way to describe the physical characteristics of a report to the system. Most customers add an application for each program that produces output that will be stored in the system. The application includes information about the format of the data, the orientation of data on the page, the paper size, the record length, and the code page of the data. The application also includes parameters that the indexing program uses to locate and extract index data and processing instructions that the system uses to load index data in the database and documents on storage volumes.

2.3.2 Application group

An *application group* contains the storage management information and index fields for data that is loaded into the system. When a report is loaded into the system, an administrator must identify the application group in which the system loads the index data and store the documents. An application group is a collection of one or more OnDemand applications with common indexing and storage management attributes. You can put several different types of reports in an application group so that users can access the information contained in the reports with a single query. All of the applications in the application group must be indexed on the same fields, for example, customer name, account number, and date.

2.3.3 Folder

A *folder* provides users with a convenient way to find related information stored in the system, regardless of the source of the information or how the data was prepared. A folder allows an administrator to set up a common query screen for several application groups that may use different indexing schemes, so that a user can retrieve the data with a single query. For example, a folder called Student Information might contain transcripts, bills, and grades, which represents information stored in different application groups, defined in different applications, and created by different programs.

2.4 Database management

OnDemand uses a database management product, such as the IBM DB2 Universal Database™, to maintain index data for the report files that are stored in OnDemand. The database management product maintains the OnDemand database, with tables that describe the applications, application groups, storage sets, folders, users, groups, and printers that you define and statistics used to optimize the operation of OnDemand servers.

2.4.1 Overview and terms

OnDemand uses a relational database product to maintain the index data for the reports that are loaded into the system. The database manager also maintains the OnDemand system tables, that describe the applications, application groups, storage sets, folders, users, groups, and printers that are defined to the system. We recommend to periodically collect statistics on the tables in the database to optimize the operation of the OnDemand database. The database manager and the database reside on the Library Server.

Application group (AG) index tables

The *application group index tables* contain the index data for the reports that are maintained on the system. When a user submits a query, the system searches one or more application group index tables for documents that match the query. For most application groups, the system also maintains a segment table. A segment table can be used to improve the performance of queries by limiting a query to the table or tables that contain the date that is specified in the query.

OnDemand system tables

The *OnDemand system tables* contain control information about the users, groups, applications, application groups, folders, storage sets, and printers that are defined to the system. The control information determines who can access the system, the folders that a user can open, and the application group data that a user can query and retrieve.

Table

A *table* consists of data logically arranged in columns and rows. For example, when an application group is added to the system, the system creates a table definition that contains one column for each application group field. When a report is loaded into an application group, the system adds one row to an application group table for each document that is contained in the report.

Tablespace

A *tablespace* is a place to store tables and a database is organized into tablespaces. OnDemand supports System Managed Space (SMS) tablespaces. For an SMS tablespace, each container is a directory in the file space of the operating system. The operating system's file manager controls the storage space.

Note: iSeries does not use tablespaces.

OnDemand system and application group tables, by default, exist in a tablespace called *USERSPACE1*. If one or more tablespace file systems are defined to OnDemand, the application group tables can be stored in them instead. We strongly encourage you to define tablespace file systems. By storing tables in tablespaces other than the USERSPACE1 tablespace, the performance of the system can be improved, enabling more efficient backup and recovery options and providing a more flexible configuration.

Container

A *container* is a physical storage device. It can be identified by a directory name, a device name, or a file name. A container is assigned to a tablespace. All database and table data is assigned to tablespaces.

Note: iSeries does not use containers.

A single tablespace can span several containers, but each container can belong to only one tablespace. It is possible for multiple containers (from one or more tablespaces) to be created on the same physical disk. The database manager attempts to balance the load of data across the containers. For SMS tablespaces, OnDemand decides on the number and locations of the containers, the database manager controls their names, and the file system is responsible for managing them.

Figure 2-3 illustrates the relationship between tables, tablespaces, container, and database.

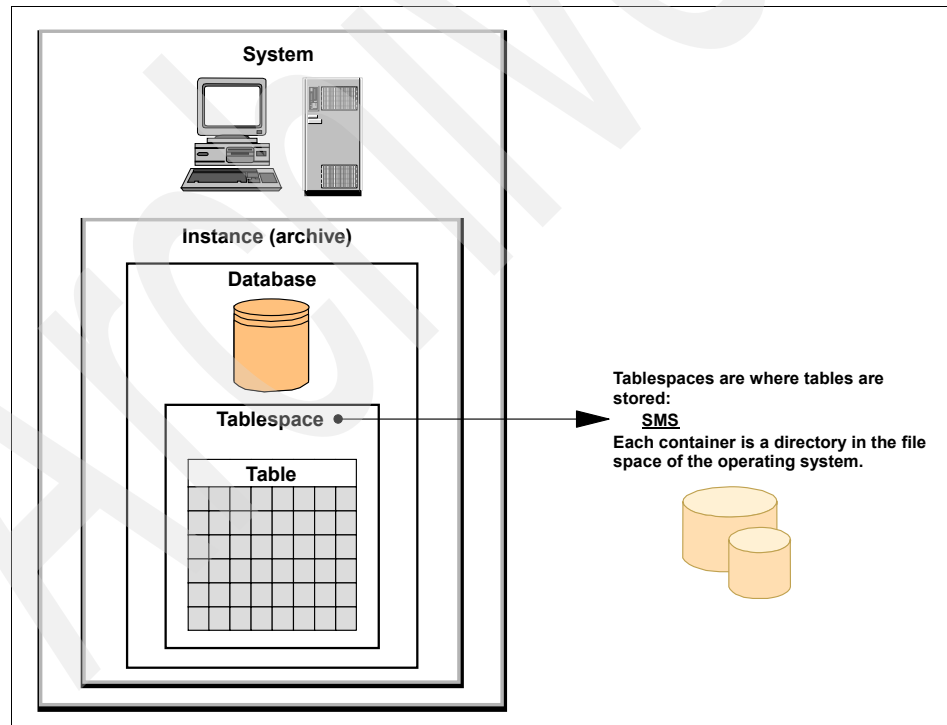


Figure 2-3 System, instance, database, table, tablespace, SMS tablespace

Note: Figure 2-3 is not relevant to iSeries. iSeries does not use tablespaces.

Index

In OnDemand, an *index* is a key that points to a document. An index allows more efficient access to documents by creating a direct path to a document through pointers. Indexes are defined when an application group is added to the system. The indexes should contain information that uniquely identifies a document, such as account number and customer name. Indexes are populated by values that are extracted from a report when a report is loaded on the system. Each row in an application group index table identifies one document.

Most application groups should not have too many indexes. There should be a good business reason to have an index. While indexes can help users find documents faster, having too many of them can slow the system down when reports are loaded on the system. Every time a new row (document) is added to an application group index table, the system has to add a row to each and every one of the indexes for that table. The more indexes that are defined, the longer it may take to load a report.

The SQL optimizer automatically chooses the most efficient way to access data in tables. The optimizer takes indexes into consideration when determining the fastest access path to data.

System catalog tables

Each database includes a set of *system catalog tables*, which describe the logical and physical structure of the data. The database manager creates and maintains an extensive set of system catalog tables for each database. These tables contain information about the definitions of the database objects, such as user tables, views, and indexes, as well as security information about the authority that users have for these objects. They are created when the database is created, and are updated in the course of normal operations. Users cannot explicitly create or drop them, but can query and view their contents by using the database manager utilities and catalog view.

2.4.2 Database maintenance

Customers should plan to maintain the database to keep it performing in an optimal manner. We recommend to run the following database maintenance tasks on a regular schedule:

- ▶ Collect statistics on tables to keep optimization information up-to-date.
- ▶ Remove index data that has reached its life of data and indexes period.
- ▶ Run database statistics to optimize index data.

Note: For iSeries, statistics collection is an automatic DB2 function.

Collecting statistics

OnDemand provides the ability to collect statistics for the application group index tables by using the ARSMaint program. The ARSMaint program collects statistics on all of the tables in the database that have changed since the last time that statistics were collected.

Removing index data

OnDemand provides the ability to remove application group index data from the database by using the ARSMaint program. Indexes *expire* (are eligible for removal) because their life of data period has passed. The indexes, and the documents that they point to, can then be removed from the system. When an index is removed, information about the document to which it points is removed from the database (the document can no longer be retrieved). However, because indexes are eligible to be removed does not mean that they will be deleted from the database. OnDemand does not delete expired index data from the database until expiration processing runs.

The application group expiration policy determines when index data is eligible for deletion from the database. The expiration policy is defined when the application group is added to the system. The following properties on the Storage Management page comprise the expiration policy:

- ▶ *Life of data and indexes.* The length of time in days to maintain index data and documents on the system. After the index data has been on the system for this number of days, it is eligible to be deleted.
- ▶ *Expiration type.* Determines whether individual indexes or an entire table of index data is deleted at a time. When OnDemand deletes index data, it either deletes a row (if the expiration type is Document) or drops a table (if the expiration type is Segment or Load and that the database organization is Single Load per database table).

Optimizing index data

OnDemand provides the ability to run database statistics, which enables the database manager to optimize application group index data and make access to information as efficient as possible.

For OnDemand for iSeries, the expiration of indexes and data is accomplished using the STRDSMOND command.

2.4.3 Database utility

OnDemand provides the ARSDB program as an interface to the database manager for the following database functions:

- ▶ Create and initialize the OnDemand database.
- ▶ Start the database manager.
- ▶ Stop the database manager.
- ▶ Create backup images of the OnDemand database. The ARSDB program can be used to create backup images of tablespaces and the full database.
- ▶ Reorganize and optimize the OnDemand system tables.

Note: The above is not relevant to iSeries. Although iSeries has an ARSDB program, it is never called or executed by the user directly. It would typically be executed through other user utilities and functions as required. In addition, on iSeries, the database manager is never started or stopped - It is integrated into i5/OS (OS/400) and it is automatically started or stopped when the entire system is started or stopped. The OnDemand system is only initialized once during the initial installation routines and no backup images of the OnDemand database are ever made. We use the normal i5/OS (OS/400) backup and restore facilities to save and restore the OnDemand libraries and related tables. We also do not reorganize and optimize database tables (only in very exceptional cases when we identify major database-related system problems. i5/OS (OS/400) has an automated database management feature built in that takes care of this.

2.5 Storage management

The OnDemand storage manager maintains files on cache storage volumes and works with TSM to manage files stored on archive media, such as optical or tape storage libraries.

Note: OnDemand for iSeries currently does not support TSM.

2.5.1 Overview

OnDemand provides a client/server system for distributed storage management. Storage management systems can contain a mixture of disk, RAID, optical, and tape. Normally, OnDemand will load documents to disk cache and the storage manager concurrently. The cache copy then expires after a specified period of time (defined by the administrator). The copy is maintained in longer-term

storage by the storage manager. Figure 2-4 shows an overview of the OnDemand storage management.

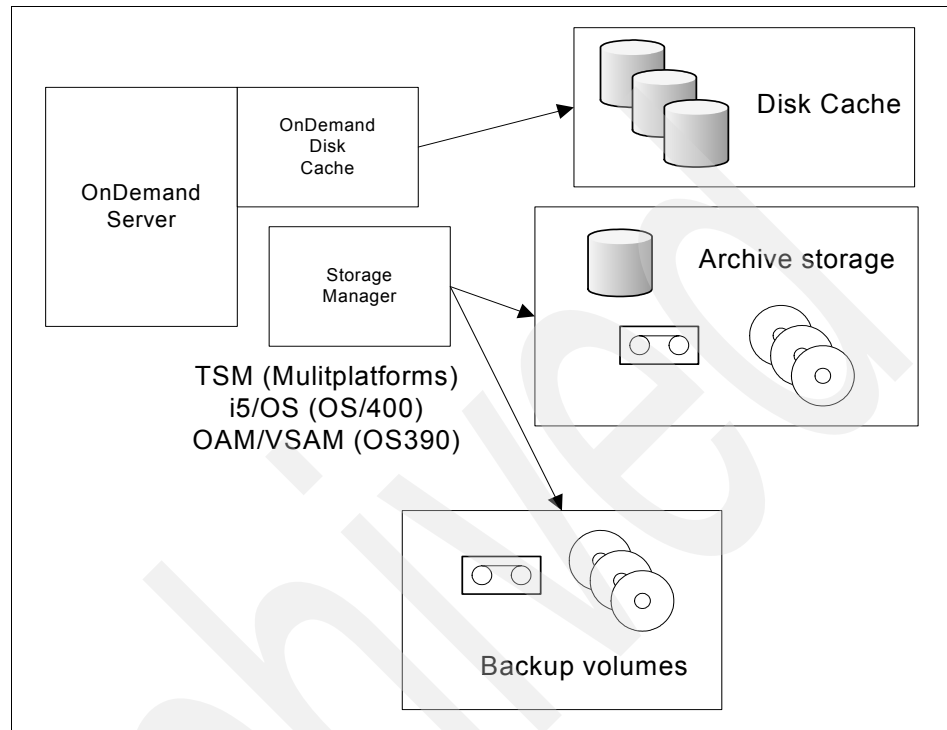


Figure 2-4 Storage management overview

2.5.2 OnDemand storage management policy and objects

There are two types of OnDemand storages, cache storage and archive storage. The OnDemand storage management policy determines:

- ▶ Where reports are stored on the system. For example, cache storage and archive storage.
- ▶ When reports are copied to archive storage. For example, when data is loaded, and the next migration.
- ▶ How long OnDemand maintains report data in the database and cache storage.

The OnDemand storage management objects are:

- ▶ Application
- ▶ Application group
- ▶ Storage node

► Storage set

The OnDemand storage management objects are added to the system by using the OnDemand administrative client.

2.5.3 Cache storage and archive storage

Cache storage is for short-term, high-speed retrieval of report data. Cache storage can be located on one or more Object Servers in a distributed OnDemand system, depending on how the storage nodes are configured. By default, reports are loaded into cache storage on the Object Server that is identified by the storage node that has the Load Data option enabled. However, an administrator can elect to disable cache storage for an application group. The *Cache Data for nnnn Days* setting on the Storage Management page determines whether OnDemand copies documents to cache storage.

Resources are always copied to cache storage on the Object Server. When a user retrieves a document that references a resource, the server sends the resource to the client. The client writes the resources to the RES directory.

The RES directory is located in <Path to client>\OnDemand32\res. For Windows, the default path is C:\Program Files\IBM\OnDemand32\RES. For iSeries, as an example, with a default instance of QUSROND, for an application group with the application group identifier VBA, the cached resources are stored in /QIBM/UserData/OnDemand/QUSROND/CACHE/0/VBA/RES.

The resource remains at the client as long as the folder is open. Every time you view a document that has a resource, the client reports whether it already has the resource for viewing that document or not. If so, it is not transferred to the client again. The client just uses the one it has cached in the RES directory. When the folder is closed, the client removes the resource from the directory.

Note: iSeries does not distinguish between Library Server and Object Server. Cache storage on iSeries is in an IFS directory.

Archive storage, maintained by TSM, is for long-term storage and for backup copies of reports. The system can be configured to copy the report data to archive storage when the report is initially loaded into the system or at a later time, depending on how the application groups are configured. You can configure a system to copy report data to cache storage and archive storage at the same time when the data is loaded into the system.

2.5.4 Storage set and storage nodes

A *storage set* is a collection of storage nodes that support application groups with similar storage management characteristics. A storage set can write to only one storage node at a time. At any point in time, all of the data that is being written to the storage set will be maintained using the same type of media and devices and the data will be maintained on the system for the same length of time.

You can define one or two storage sets for an OnDemand system based on the storage requirements of the reports that you plan to load into the system. For example, you may define two storage sets: one to maintain data for ninety days and another to maintain data for seven years.

A storage set can contain one or more primary storage nodes. A primary storage node determines where the system manages the reports and resources that are stored in an application group.

A *storage node* identifies the Object Server on which the application group data is stored. Application group data is automatically loaded into cache storage on the specified Object Server, unless an administrator disables this option in the application group.

A storage node can also identify a client node in TSM. If TSM is being used to maintain application group data in archive storage, then each storage node that writes data to TSM-managed storage must identify a client node in a TSM domain. The properties of the TSM domain determine the devices that are used to hold the data and how long TSM maintains the data.

If a storage node does not identify a client node in TSM, it is known as a *cache-only storage node*. Although this property can be changed after data has been loaded, only new data will be written to TSM. Any data that was written to the storage node before the change was made must be reloaded, unless the application groups that use the storage set were configured to migrate data some time after the data was loaded and before the data expires from cache storage.

Note: On iSeries, you cannot create or maintain storage set and storage nodes. These are automatically created and integrated into the migration policies.

In addition, OnDemand for iSeries uses ASM (archive storage manager) for long term storage of reports.

Again, TSM, Library Server, and Object Server are not relevant to OnDemand for iSeries.

2.6 Report migration and removal

OnDemand provides the ability to migrate reports from one storage node to another and the ability to remove the reports from storage.

2.6.1 Migrating reports

OnDemand provides automatic migration to copy reports from cache storage to archive storage (for reports that were not loaded to archive storage) and to make reports eligible for deletion to maintain free space in cache storage file systems. Migration helps to ensure that there is sufficient free space in the cache storage file systems, where faster devices can provide the most benefit to users.

The ARSMAINT program uses an application group's storage set (also known as migration policy in iSeries) to control when migration for an application group occurs:

- ▶ If the *Next Cache Migration* option is specified to control when migration for an application group occurs, the cache storage manager runs migration processing each time that the ARSMAINT program is started with the appropriate options.
- ▶ If the *After nnnn Days in Cache* option is specified to control when migration for an application group occurs, a report must be stored in cache storage for at least the specified number of days before it is eligible to be migrated.

The ARSMAINT program migrates reports from each cache storage file system listed in the cache storage file system file. The space in cache storage that is taken by migrated reports can be reclaimed by the cache storage manager by running expiration processing. In general, after migration processing completes, expiration processing should be run so that the cache storage manager can reclaim the cache storage space that was occupied by migrated reports.

2.6.2 Removing reports

Reports *expire* (are eligible for removal) because their cache expiration date or archive retention period has passed. Expired reports can then be removed by the storage managers. The cache storage manager identifies reports for removal by using the application group's expiration policy and high and low expiration thresholds. The archive storage manager marks reports for removal based on the criteria defined in the archive copy group.

Reports expire from cache storage when they reach their cache expiration date. If a report's cache expiration date is less than its Life of Data period, then the report is simply removed from cache storage. Subsequent requests for the report are satisfied by the archive storage manager. When the report reaches its Life of

Data period, information about it is removed from the OnDemand database (the report can no longer be retrieved).

When the report's archive retention period has passed, information about it is removed from the archive storage manager database. Because a report is eligible to be removed does not mean that it will be deleted from storage. The cache storage manager does not delete expired reports from storage until expiration processing runs. During expiration processing, the archive storage manager deletes information about expired reports from its database. However, the actual reports remain on archive media until such time that the space that they occupy is reclaimed.

Important: The system copies data to and retrieves data from storage by using storage management policies that are defined in both OnDemand and TSM. OnDemand and TSM maintain data *independently* of each other. For example, OnDemand and TSM independently determine when the data that they maintain expires and should be removed. OnDemand and TSM use their own utilities and schedules to remove data. In general, you should plan to keep data in TSM as long or longer than they are kept in OnDemand.

OnDemand for iSeries currently does not support TSM. The above description is not applicable to OnDemand for iSeries.

2.6.3 Removing reports from cache storage

The expiration policy determines when reports are eligible for deletion from cache storage. The expiration policy is defined when the application group is created.

As mentioned earlier, the cache storage manager does not delete expired reports from cache storage until expiration processing runs. The ARSMAINT program is the expiration utility. The ARSMAINT program can be scheduled to run automatically or it can be run manually. The ARSMAINT program should be run periodically so that the cache storage manager can reclaim the space that is occupied by expired reports.

2.7 Tivoli Storage Manager

As discussed earlier, an OnDemand system can maintain copies of reports in cache storage and in archive storage. The copies in archive storage are for long-term storage. TSM is the product that OnDemand works with to maintain reports in archive storage. TSM is used primarily to maintain a backup or long-term copy of OnDemand documents. It can also be used to maintain index

data that has been migrated to archive storage and to maintain DB2 archived log files and DB2 backup images files.

Note: TSM is not applicable to OnDemand for iSeries.

For OnDemand for iSeries, the STRDSMOND command should be run periodically to manage cache storage.

TSM includes the following components:

- ▶ A *server program* that maintains information about the devices and data that it manages. The server program also controls the storage media and devices that are defined to TSM.
- ▶ An *administrative client program* that can be used to control and monitor the server program activities and define storage management policies. The activities include *expiration* processing, which is the process of deleting data that is eligible to be removed from the system, and *reclamation* processing, which is the process of reclaiming the space take by expired data. Storage volumes that have been reclaimed can be reused. The storage management policies determine where data is stored and how long TSM maintains the data.
- ▶ An *API* that OnDemand uses to work with TSM.

Note: The TSM API is required on the Library Server and all Object Servers that use TSM.

- ▶ *Device support modules* that provide support for storage devices and storage libraries. TSM supports many optical and tape storage devices.

2.7.1 TSM storage management policy and objects

The TSM storage management policy determines:

- ▶ The media and devices on which reports are stored.
- ▶ The length of time that TSM maintains report data in the storage that it maintains.

The TSM storage management policy is specified by using the TSM administrative client. The TSM storage management objects are:

- ▶ *Client node*: Represents an Object Server on which the TSM backup-archive client program has been installed, and has been assigned to a policy domain.

- ▶ *Policy domain*: Contains the policy set, management class, and archive copy group that is used by the client nodes that are assigned to the policy domain.
- ▶ *Policy set*: Contains the rules that are currently in use by all client nodes that are assigned to the policy domain.
- ▶ *Management class*: Determines where data is stored and how it is managed.
- ▶ *Archive copy group*: Used to copy data to TSM for long-term storage.

A client node is registered in a policy domain. The other TSM policy objects are within the policy domain. When a report is copied to archive storage, it is bound to a management class. The management class and the archive copy group within it specify where the report is stored and how it is managed.

The TSM storage management objects are added to the system by using the TSM administrative client.

2.7.2 TSM storage devices and media

The TSM storage devices and media are:

- ▶ *Library*: A TSM library is one or more drives (and possibly robotic devices) with similar media mounting requirements.
- ▶ *Drive*: Each drive defined to TSM represents a drive mechanism in a tape or optical device.
- ▶ *Device class*: Each device is associated with a device class that specifies the device type and how the device manages its media.
- ▶ *Storage pools and volumes*: A storage pool is a named collection of storage volumes from the same media type. A storage pool is associated with a device class. For example, an OPTICAL storage pool contains only optical storage volumes. A storage pool volume is associated with a specific storage pool.

A storage pool is the destination for reports that are copied to archive storage. An archive copy group specifies the name of the storage pool. The storage pool is mapped to a device class, which represents a device. The storage pool contains volumes as indicated in the device type that is associated with the device class. All devices require a device class that specifies their associated device type. Optical and tape devices also require a library and drive for management of media, including the mounting of that media.

For more information about TSM, refer to the *Tivoli Storage Manager for Windows Administrator's Guide*, GC35-0410, and the *Tivoli Storage Manager for Windows Administrator's Reference*, GC35-0411.



Part 2

Multiplatforms

In Part 2, we focus on OnDemand for Multiplatforms. We describe the backup and recovery strategies and options, and high availability and business continuity strategies and options for OnDemand on Multiplatforms. Along with these options, we provide practical procedures and steps to accomplish the backup, recovery, and high availability with sample commands and scripts. In addition, two case studies are presented to show you how real-world businesses implement backup procedures, high availability configurations, and disaster recovery plans.

Archived

Backup and recovery for OnDemand Multiplatforms

In this chapter, we describe the backup and recovery strategies and options available for OnDemand on Multiplatforms. We also discuss the configuration files and data to be backed up to prepare for a disaster recovery, the ways to back up such information, and the recovery procedures to restore the backup data into a new system or existing system.

The following topics are covered:

- ▶ Backup strategies and options
- ▶ Practical procedures
- ▶ Recovery plans under different scenarios
- ▶ Recovery procedures
- ▶ Problem determination

The procedures provided in this chapter include:

- ▶ Backup and recovery of OnDemand database and system files
- ▶ Backup and recovery of TSM databases
- ▶ Best practices on restoration of TSM storage pools

3.1 Backup strategies and options

In times of recovery, everyone wants to restore as much data as possible back to where it was; however, this usually depends on how often and how much data are backed up at the time. In many scenarios, it might be more cost effective and practical to reload the data that were just loaded recently, rather than to backup all the data until the very last bit using incremental backup data.

A typical OnDemand installation involves several components:

- ▶ The operating system for the software to run on
- ▶ The storage area, disk space to keep the cache data
- ▶ A database to store the indexes to allow users to retrieve the data
- ▶ Set of media managed by storage management software (optional)
- ▶ OnDemand configurations

In this section, we explore different ways to backup the above components and the frequency of backing them up. The steps apply to both general backup purposes as well as for disaster recovery purposes. Note, for disaster recovery purposes, provision has to be made to transport the backup media off-site on a regular basis.

3.1.1 Operating system backup

A fresh OnDemand installation takes time to complete, not to mention applying all the changes made to the system. To speed up recovery, we should have a backup of the operating system.

The operating system should be backed up on:

- ▶ Regular basis, when there is no significant changes to the system. We recommend doing this on a quarterly basis.
- ▶ Ad hoc basis, when there are changes to the system such as an update of patches or fixes (PTFs). We recommend doing this before and after the changes.

Although it is convenient to keep the backup at the server side, we recommend to bring at least one system backup off-site regularly to protect against site disaster.

3.1.2 Cache backup

OnDemand cache storage is the primary, short-term storage location for reports. Typically, they are located in either journal file systems in UNIX® servers or drives in Windows servers. By default, if data resides on both the cache and the secondary storage, OnDemand will search for the data in cache first; if the data

is not in cache, then OnDemand will search for the data in the secondary storage. If your system is setup this way, the cache backup might not be necessary. If, however, you store data in the hard disk as cache-only storage node, then, OnDemand has only one place to locate the data and cache backup would be extremely important.

Since the cache storages usually contains small files that are scattered across all the cache file systems, we could use available operating system commands or software to back them up just like any other ordinary files.

Note: When backup a cache storage, make sure the system is at quiescence so that the cache is in a consistent state. This is because any loading, migration, or expiration activities might change the content of the cache directories. Moreover, the OnDemand maintenance program, ARSMAINT, always check if the cache is in consistent state before running migration or expiration activity.

In AIX, the `tar` or the `pax` command can be used to backup the entire cache directory. The backup can either be stored on a storage media or stored as a file in another directory, drive or hard disk. In case of multiple cache directories, you could backup each of them into different files. It is better not to use absolute path so that you can have to freedom of restoring the backup into another location.

Using TSM to manage cache backup

To bring the cache backup offsite, the backup files can be copied to tapes or stored in TSM; however, it might not be a good idea to backup or archive the cache directories *directly* into TSM or any other intelligent storage software. This is because the cache directories typically make up of millions of small files and links, and there is an overhead in terms of time and space for each file that stores into a storage manager; doing so might cause the backup and archive time to become too long. It might be more efficient and less time consuming to backup the cache directories into a few large backup files, and then backup and/or archive the files into TSM. Alternatively, instead of storing onto the TSM libraries residing in the OnDemand server, these large files could be stored into your established corporate central backup server regularly. In this way, they could be used to restore to any system on the same network, including the disaster recovery server.

3.1.3 Database backup

The OnDemand database not only stores the indexes for all the OnDemand data, it also contains the OnDemand application definitions. The following are the three important parts that need to be backup for database:

- ▶ Database configurations
- ▶ The database
- ▶ Database logs for roll forward

Database configurations

The database configurations describe the property and options used for the database, such as heap size and log path. These configurations can be backup via database commands.

The database

OnDemand provides the ARSDB program to create the backup images of the OnDemand database. The ARSDB program can take incremental tablespace backups and full database backups. You can run it online while other users are connected to the system or offline when the OnDemand server programs and other related processes are stopped. With ARSDB, we have choice of backup database online or offline, incremental or full, or even just a tablespace alone.

Offline backup

The offline backup is done while users are not working on the system and no other system activities are performed. The safest and cleanest way of a DB2 backup is the full offline database backup. This is because we do not need to use any logs to roll forward the database and the system can be functional after just one restoration without roll forward. To restore the database to point in time, we could also make use of the logs to roll forward.

The disadvantage of doing offline backup is that users cannot access the system during the backup period and loading has to be stopped.

Online backup

The online backup can be done during the normal business operation.

The advantage of doing online backup is that the application can continues to run during the backup; users can continue to login and loading of data is allowed during the backup.

The disadvantage of doing online backup is that the log files become very important. Roll forward of database will fail if there are missing logs that cannot be found.

Incremental backup

An incremental backup means that OnDemand backup only those tablespaces that have changed since the last full backup, whether it was offline or online backup. OnDemand creates one backup image for each tablespace that has

changed since the last backup. In this sense, it is very much similar to the incremental tablespace backup.

Incremental tablespace backup

We could use the same OnDemand program, ARSDB, to perform incremental backup on a specific tablespace. This can be used during the testing phase of applications.

Examples on DB2 backup is illustrated in 3.2.2, “DB2 backup procedures” on page 60.

Database logs backup

There are the primary log and the archive log (unless TSM is handling the archiving) that need to be backup.

On AIX, the primary log resides in the directory where the parameter `ARS_DB2_PRIMARY_LOGPATH` specified and the archive log will be copied to the directory specified by `ARS_DB2_ARCHIVE_LOGPATH` unless TSM is handling it.

If you are not sure whether the database logs are handled by TSM, check the link of the file `/usr/opt/db2_08_01/bin/db2uext2`, it should be either linked to `db2uext2.disk` or `db2uext2.adsm`:

- ▶ The program *db2uext2.disk* copies the online log files from the primary log file directory to the archive log file directory. A log file becomes offline when it is no longer stored in the primary log file directory. After creating a backup image of the database using the ARSDB program, ARSDB deletes the offline archived log files.
- ▶ If *db2uext2.adsm* is used instead, the archive log will be managed by TSM and will only be expired based on TSM policy. The ARSDB program will not delete the archive log files.

Backup strategy

With different types of backups, a combination of several types of backup methods is often employed.

The factors in deciding which type of the backups to use and how often to perform the backup include the following:

- ▶ Ease of restoration

Always keep restoration in mind while deciding the backup policy. A backup policy that is too complicated to restore might take long time to do so. It might also introduces more errors or failure point during restoration. Comparing all

the backup methods, a full offline backup is the easiest and most reliable for restoration, because it does not even bother if the logs are all missing.

- Backup window/system available time

Some systems need to run 24 x 7. In this case, there might not be time for offline full backup; online full backup might become the only option. For the scenario where users only need to access the system during normal business hours, it would be feasible to perform offline backup, even everyday.

- Amount of data to reload

This normally determines how often a full backup can be done without losing data. If the source of the data is still available within seven days of being loaded to OnDemand, then we can afford to backup the database offline every seven days. In case of failure, the offline database backup can be restored and the data for the past seven days can be reloaded. Be aware that if the OnDemand system has annotation, those annotations added after the offline backup will be lost using this method of backup and restore.

- Size of database

Depends on the size of database, it might be more viable to backup the database using incremental instead of full backup. If the frequency of online/offline full backup decreases, there might be few side effects. Firstly, the size of each of the incremental backup increases because incremental backup is based on the last full backup, and there will be more and more changes since the last full backup. Secondly, more tablespaces would have changes and therefore, more backup images will be generated as the time between the last full backup becomes further apart. Thirdly, the time taken to complete the incremental backup increases.

- Purpose of backup

If backup is performed because of a impending migration or upgrade, then a baseline of the data loaded needs to be identified. Because there is a higher chance to perform restoration than normal, ease of backup is important in this case. Due to the purpose of this type of backup, a full offline backup would be a better choice.

In general, if the backup windows allows, perform offline full backup everyday. If the system has to be available most of the time, then a combination of offline and online full backup will be fine. In the worst scenario, if you do not have the luxury to take the system down at all, then a full backup, whether offline or online, should be performed at least once a week, with incremental backup in between. This is to make sure that the incremental backups do not grow so much that they get out of hand. Again, with the last option, both the restoration time and complexity increases.

3.1.4 TSM backup

The TSM database contains information that is needed for TSM server operations and information about client data that has been backed up, archived, and space-managed. The TSM database does not store client data. Instead, the database points to the locations of the client files in the storage pools. The TSM storage pools contains the data which OnDemand stored into long term archival.

There are two major area to be concerned with TSM backup: prevention against hardware failure and protection against disaster.

Prevention against hardware failure

Disk corruption is normally caused by hardware failure. To protect data lost against such failure, operating system utility such as mirroring or RAID could be used to make it more reliable. This is also the way to protect the operating system and cache.

In TSM, there is additional ways for prevention against hardware failure. For TSM database volume and log volume, there is the TSM mirroring which can mirror up to three copies of data. With this feature, the mirror copies could be placed in other physical hard disk. In case there is a hard disk failure on one of the volume group, TSM can still function with the second or third copy residing on another physical disks.

For TSM managed media, we can make use of copy storage pool feature which allows data duplication onto the same or different device class; therefore, we could make use of slower media, such as tapes, to store a copy of the primary storage pool. If TSM detect problem on the primary, it will automatically retrieve data from the copy storage pool on the next access.

Protection against disaster

Assume that the primary site has been destroyed, offsite backups will be needed to restore the data. In TSM, the most important backup is the TSM database backup. Without the TSM database, the media is meaningless to us. The next important backup is the configuration files that starts the TSM server. Lastly, it is the data itself.

TSM database can be backup into the following:

- ▶ Media that belongs to TSM device class, such as tape or optical platter.
- ▶ TSM file device class. Note, you still need to copy it to the tape for offsite storage.

Details of how to perform TSM database backup is explained in 3.2.4, “TSM backup procedure” on page 66.

To protect data in TSM libraries, TSM media could be duplicated to copy storage pools. The device class for the copy storage pools can be different from the original storage. It is good practice to have media of bigger capacity so that more data can be stored into one media and we do not need to bring too many media offsite. Another advantage of having the copy storage pool on another library is that the number of drives, used during backup storage pool in the library of the primary storage pool, will be reduced compare to backing up the primary storage pool onto the same library.

Different primary storage pools can be combined to backup to the same copy storage pools. This helps to reduce the number of copy storage pools, which in turns reduce the number of media used and ease the media management.

Backing up TSM for disaster recovery

In order to recover all the data in TSM offsite, we need all the volumes from the copy storage pool. At the production site, in case a primary volume goes bad and a user retrieves data from it, the copy volume will be needed. In such situation, if all the copy volumes are sent to offsite, TSM will not be able to retrieve from the copy storage pool and the user will not be able to retrieve the data. Moreover, it will be an operational hazard to bring back the storage pool volume, run backup storage pools and send them back to offsite everyday. One way to overcome this is to have *two* copy storage pools for each primary storage pool. In this way, we can send one set offsite, while leaving one set behind as a backup copy.

The frequency of swapping the two sets of copy storage pool volumes can be based on how much data can be lost in case of disaster versus the optimal frequency to bring the copy storage pool volume offsite. Only the set of copy storage pool volumes which is at the production site will have the latest data. The set that was sent offsite will not contain the data loaded into TSM after they were being checkout from TSM libraries. One of the common practice is during the weekly full database offline backup, backup cache and TSM, and then send all of them offsite for disaster recovery purpose. In this case, we will have the full set of backup for up to the very last week.

3.1.5 OnDemand configuration and definition

Although the OnDemand configuration files could be backup by operating system, we recommend keeping a copy of these file at some handy place just in case. Similarly, even though the OnDemand definition is in DB2, it is a good practice to have the definition copied to another place for easy reference and restore. There are also customized programs and files that we would want to keep in case of recovery.

OnDemand configuration files

In AIX, OnDemand configuration files are flat files, located in the directory /usr/lpp/ars/config. On Windows, these settings are in the Windows' registry, under the HKEY_LOCAL_MACHINE. You can export the registry information out to a file.

OnDemand definitions

Other than the configuration files, the OnDemand software definitions such as user, group, application, application group, and folder information are all stored in the DB2 database.

While deleting the application group will delete the actual data, deleting user, group, application or folder does not delete any data. If any of these data (other than the application group) is deleted, it might not be worth the effort to restore the entire DB2 database just to get back these definitions. On the other hand, it could be very painful to re-define users, groups, and applications should they be accidentally deleted.

It is, therefore, handy to keep the copies of the OnDemand definitions in a local server via Export function or keep the definitions on a file via Report or Summarize function. The local server is self-contained and is defined using files in a directory located on a workstation rather than in a database. The files represent the system tables that define the various objects such as users, groups and applications. Details on how to do this is discussed in 3.2.5, "OnDemand configuration and definition backup" on page 70.

OnDemand customized files

If user exits are used, they should be backup because they are customized configuration. Moreover, if AFP resources are used, then under the AFP resources directories, there should be formdef, pagedef, overlays or page segments. These files should be backup as well.

3.2 Practical procedures

This section discuss the steps and the commands used to perform different kinds of backup as described in 3.1, "Backup strategies and options" on page 52. Examples of the actual commands used are provided as well.

3.2.1 Operating system

Although most of the time it is the data that needs to be restored, there might be occasion when restoration of the operating system is necessary.

Backup steps

On AIX, perform the backup using **mksysb** command as root users, one of the important flag to use is to set the option “Disable software packing of backup?” to yes. Do remember that mksysb only backup mounted journal file systems in the rootvg volume group. Note that in AIX, OnDemand software is installed in the directory /usr/lpp/ars, and the configuration files are under the /usr/lpp/ars/config directory. These directories are normally inside the rootvg and therefore, they will be backup via mksysb.

We could use smit to manually perform mksysb, or alternatively, we could use command line as follows, assuming backing up to the drive in device path /dev/rmt0:

```
/usr/bin/mksysb -i -X -p /dev/rmt0
```

For Windows, there are many options available to backup the operating system, such as TSM, Norton Ghost and Veritas. It is advisable to use existing backup solution in your organization and make use of the steps well known in your company to restore the system.

Practical considerations

If the backup is to restore into another server, the backup media used must be compatible on the tape drives for both servers. It might look obvious now but just make sure that both servers use the same type of media.

On AIX, if restoring the mksysb backup to another server, you should boot the machine in maintenance mode using OS CD of the same version as the OS level of the machine where the mksysb is performed. If rootvg is mirrored, then by default, you need double the space to restore.

For Windows server, TSM requires the server to have *identical hardware*. Be sure to take note of the configuration of the existing hardware.

In both cases, the TSM database or DB2 database should not be in the operating system's drive or volume group; otherwise, database restoration is needed after the operating system restoration. This is because operating system restoration includes all the directories in the operating system volume unless explicitly excluded.

3.2.2 DB2 backup procedures

OnDemand provides the ARSDB program to perform backup which will remove the archive log upon successful backup depending on which db2uext2 is used. This helps to house keep the archive log directory and avoid problems resulting from manual premature deletion of archive logs.

ARSDB is used in many ways in OnDemand. The syntax for performing database backup is shown in Table 3-1. Note that the -v option is for verbose mode and the <device_name> is the file name of media device.

Table 3-1 OnDemand database backup commands

Backup command	Purpose
arsdb -vy <device_name>	Create offline full backup of the database
arsdb -vY <device_name>	Create incremental, offline full backup of the database
arsdb -vz <device_name>	Create online full backup of the database
arsdb -vZ <device_name>	Create incremental, online backup of the database

Online backup

Online backup can be performed anytime when OnDemand system is running. Example 3-1 shows the execution and the output of an online full backup. In this example, the backup is stored in a local directory, /ondemand/backup_dir.

Example 3-1 Online backup command and its output

```
#arsdb -vz /ondemand/backup_dir
Backing up the DB2 ARCHIVE database online
Backing up the DB2 ARCHIVE database at 08/17/04 10:45:15
Timestamp for backup image is: 20040817104516
#
```

The archival of the archive logs are done via the OnDemand db2uext2 program, which is linked to either db2uext2.adsm or db2uext2.disk, depending on whether they are managed by TSM or placed in disk.

If it is managed by disk, the old logs in the archive log directory will be removed after backup. In rare occasions, if a primary log was filled up during a backup, and the primary log is moved to the archive log directory, then the log may be removed by the backup process without being backed up and we have missing log situation. To prevent this type of missing log situation, always backup the primary and archive logs *prior* to the online backup.

If TSM is used to manage the archive logs, the ARSDB program does not automatically remove the archive logs.

Offline backup

During offline backup, the OnDemand socket daemon, ARSSOCKD, needs to be shutdown to ensure that there is no connection to the database. Once it is shutdown, users will not be able to access OnDemand during the backup.

Beginning with version 7.1.0.14, OnDemand provides the following command to stop the server process on Unix servers:

arssockd stop <instance_name>

Where <instance_name> is the name of the OnDemand instance to stop. The default instance is the archive instance. Note the following:

- ▶ When you stop the server process, all users who are connected over the network to the OnDemand system will be disconnected. It is, therefore, a good practice to warn connected users before stopping the server process.
- ▶ The **arssockd stop** command is for Unix servers only. For Windows servers, the server process runs as a service. The service can be stopped by using the Services administrative tool in Windows or the OnDemand configurator client.

If you are running OnDemand prior to version 7.1.0.14 on Unix, use the **ps** command to find the process identification number (PID) and stop the process using the **kill** command as shown in Example 3-2.

Example 3-2 Steps to stop OnDemand socket daemon in Unix in the old version

```
PID=`ps -Aef | grep "arssockd: (accepting)"|grep -v "grep"|head -1|awk '{ print $2 }'`  
kill $PID > /dev/null 2>&1
```

Example 3-3 shows the error message when DB2 offline backup was performed while ARSSOCKD is active.

Example 3-3 Offline backup failed when OnDemand is still active

```
#arsdb -vy /ondemand/backup_dir  
Backing up the DB2 ARCHIVE database offline  
Deactivating database ARCHIVE  
The ARCHIVE database was deactivated, however there is still a connection to  
one or more nodes  
Backing up the DB2 ARCHIVE database at 08/17/04 11:16:27  
Unable to backup database ARCHIVE. err=-1035  
arsdb: Unable to backup DB2 ARCHIVE database to /ondemand/backup_dir.  
rc=-1035  
arsdb: The DB2 ARCHIVE database was deactivated. Please use -k to activate  
the database  
#
```

As shown in the previous example, *always* stop ARSSOCKD before starting offline backup. The next example, Example 3-4, shows a successful offline

backup when ARSSOCKD is stopped before the execution. Once again, the old archive logs will be removed.

Example 3-4 Successful offline backup

```
#arssockd stop
#arsdb -vy /ondemand/backup_dir
Backing up the DB2 ARCHIVE database offline
Deactivating database ARCHIVE
Backing up the DB2 ARCHIVE database at 08/17/04 11:19:37
Timestamp for backup image is: 20040817111938
#arssockd
```

In Windows server, if you want to run DB2 offline backup via batch job, you could do similar steps as shown in Example 3-5.

Example 3-5 Offline database backup in Windows

```
SET ODPATH=C:\Program Files\IBM\OnDemand for WinNT\bin
SET DEVICE="C:\arsbackup"
SET LOG="C:\arslog\backup.log"
Net stop "OnDemand LibSrvr (ARCHIVE)" > %LOG% 2>&1
"%ODPATH%\arsdb" -y %DEVICE%>> %LOG% 2>&1
Net start "OnDemand LibSrvr (ARCHIVE)" >> %LOG% 2>&1
```

Using TSM to maintain backup images

TSM can be used to maintain DB2 backup image files. TSM can maintain the incremental tablespace backups and full database backups that are created with the ARSDB program.

Do not be confused between the TSM managed DB2 archive log and the DB2 database backup on TSM. You can still backup the DB2 database into TSM even if the logs are managed on disk. If the archive log is managed by TSM, then using ARSDB program to backup the database will not remove the archive log files in TSM.

In order to use TSM for DB2 backup, configure the policy in TSM and define a node for DB2 backup use. You may name the TSM option file dsm.opt.db2, as this is the default name in ars.cfg.

On the OnDemand server, update the ars.cfg file with the correct path and file name for the dsm.opt file:

```
ARS_DB2_TSM_CONFIG=/usr/tivoli/tsm/client/api/bin/dsm.opt.db2
```

The default path on Windows server is:

```
\Program Files\Tivoli\TSM\Baclient
```

Example 3-6 shows online backup of DB2 database into TSM. Note the keyword used for the ARSDB program to backup to TSM is **ADSM** and in *capital letters*.

Example 3-6 Backup DB2 online to TSM

```
#arsdb -vz ADSM
Backing up the DB2 ARCHIVE database online
Backing up the DB2 ARCHIVE database at 08/17/04 12:07:30
Timestamp for backup image is: 20040817120731
```

3.2.3 Cache backup procedure

The OnDemand cache is made up of directories, compressed data files and links. The content or permission changes as data are being loaded into OnDemand, during cache migration or when cache expiration is run. It is therefore important to backup the cache directories only when there is no loading, expiration, or migration. One of the ideal time to backup cache is during offline backup of DB2 because at that time the OnDemand socket daemon will be stopped.

On Windows, there is native backup software which could be used to backup the cache drives into files or other media. For disaster recovery purpose, the cache data can be backup to removable media for offsite storage.

On Unix, there are commands such as **tar** or **pax**, which could be used to backup the cache directories directly into a removable media or a file. This is one of the more efficient way to perform a backup; however, it requires manual tracking of the individual tapes or file. Alternatively, backup software such as TSM or Veritas can be used.

Practical considerations

The best time to backup OnDemand cache is when there is no change to the file systems. Note that the permission on the cache directories has special meaning to OnDemand: It changes as data are migrated from cache. In addition, the disaster recovery site might not have the same device as the production site. We therefore recommend cache backup after cache migration.

As cache becomes bigger, the time it takes to backup via backup software might become too long and the database of the backup software will become very big. This is because if backup is performed on individual files, each file will need an entry in the database; as cache grow, the sum of the overhead associates to each of them become very significant. One way to overcome this is to use operating system command such as **tar** or **pax** to backup into a few large files, then use TSM or Veritas to archive or backup these files into the library. This will greatly reduces the overhead and improves efficiency, as TSM will only need to

handle few large files instead of examining and recording each of the small files. It will also save disk space on backup as individual cache directories can be backup into one large file and send to TSM or Veritas one after the other.

When using command such as **tar** or **pax**, be careful on the use of path. It is easier to restore when the backup is done using relative path rather than absolute path. Otherwise, the restore will overwrite existing data on cache directories.

A script can be written to backup cache directory and archive to TSM one by one. In Example 3-7, we show how to **tar** cache directory to a disk and send to TSM one by one. Note that for this to work, the directory to contain the tar backup, as defined in BKCDIR, must be at least the size of the largest cache. The TSM option file used is dsm.opt.cache, which has to be registered in TSM for the archive purpose.

Example 3-7 Sample script to backup cache in Unix using TSM archive function

```
export DSM_CONFIG=/usr/tivoli/tsm/client/ba/bin/dsm.opt.cache
export DSM_DIR=/usr/tivoli/tsm/client/ba/bin
MAIL_ID=xxx@abc.com
BKCDIR=/bkcache
LOG=/arslog/cachebkTSMarchive.log
DATE=`date +%Y%m%d`
TARC1=${BKCDIR}/archive.c1.${DATE}.tar
TARC2=${BKCDIR}/archive.c2.${DATE}.tar
TARCN=${BKCDIR}/archive.cN.${DATE}.tar

mount ${BKCDIR}
cd ${BKCDIR}
if [ $? -eq 0 ]
then
    echo "===== Starting Cache Backup on $DATE ====="
    cd /
    tar cvvf ${TARC1} ./cache1 >/dev/null 2>&1
    if [ $? -eq 0 ]
    then
        dsmc arch -deletefiles ${TARC1}
        echo "Send ${TARC1} to TSM completed at `date`"
    else
        echo "Tar Cache1 to disk Fails on" ${DATE}
    fi
fi

#Test if archival of the cache1 has finished
ls ${TARC1}
RESULT=$?
while [ $RESULT -eq 0 ]
do
```

```

        echo "archival of cache1 is still running"
        sleep 300
        ls ${TARC1}
        RESULT=$?
    done
    echo "archival of cache1 is has been completed"
    echo

#Add in more cache directories as needed
#The last one looks the same

    cd /
    tar cvvf ${TARCN} ./cacheN > /dev/null 2>&1
    if [ $? -eq 0 ]
    then
        dsmc arch -deletefiles ${TARCN}
        echo "Send ${TARCN} to TSM completed at `date`"
    else
        echo "Tar CacheN to disk Fails on ${DATE}"
    fi
    echo "End of Cache Backup to TSM"
else
    echo "Backup directory does not exist, aborting backup"
fi

```

3.2.4 TSM backup procedure

When using TSM backup for long term archival storage of OnDemand data, we need to backup the following:

- ▶ The TSM configuration files
- ▶ The TSM database
- ▶ The media managed by TSM

We can also use the TSM mirroring utility to safe guard the TSM database.

TSM configuration files

The main TSM server configuration file is called dsmserv.opt. It is in the following directory:

- ▶ For AIX, /usr/tivoli/tsm/server/bin
- ▶ For Windows, \Program Files\Tivoli\tsm\server1

In the same directory, the file dsmserv.dsk provides information about the name of database and log volume, as shown in Example 3-8. It does not provide the size of the files, for recovery to another server. Take note of the current size of the

database. If the size of database is underestimated, restoration of TSM database will fail.

Example 3-8 A sample dsmserve.dsk file on AIX server

```
#dsk_comment#page_shadow_token:1040818153401
/ondemand/tsmlog/log1.dsm
/ondemand/tsmdb/db1.dsm
/ondemand/tsmdb_mir/dbmir1.dsm
/ondemand/tsmlog_mir/logmir1.dsm
```

In following Example 3-9, the output of the commands shows the total size of database volumes is defined to be 16 MB and the total size of log volume to be 8 MB.

Example 3-9 Output of query database and query log on TSM

tsm: TSM>q db

Available Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Usable Pages	Used Pages	Pct Util	Max. Pct Util
16	16	0	12	4,096	4,096	293	7.2	7.7

tsm: TSM>q log

Available Space (MB)	Assigned Capacity (MB)	Maximum Extension (MB)	Maximum Reduction (MB)	Page Size (bytes)	Total Usable Pages	Used Pages	Pct Util	Max. Pct Util
8	8	0	4	4,096	1,536	126	8.2	17.2

Backup TSM database

The TSM database can be backup with the command **backup db**. This command is used to backup TSM database to sequential access volumes. Note that if it is backup to a media device other than file, the entire media will be dedicated to *one* backup and it cannot be used to backup again until the backup is deleted using the **delete volhist** command.

Command to backup TSM database to a file device is as follows:

tsm: TSM>backup db type=full devclass=filedev

Example 3-10 shows how to check for TSM database backup. It displays all existing backups information in TSM. If the device class is a removable media, it

could be check out to offsite storage. If it is on a file device, we could also copy it to a removable media for offsite storage.

Example 3-10 Successful backup of TSM database

```
tsm: TSM>q volhist type=dbback

      Date/Time: 08/10/04   16:38:32
      Volume Type: BACKUPFULL
      Backup Series: 1
      Backup Operation: 0
      Volume Seq: 1
      Device Class: FILEDEV
      Volume Name: /ondemand/tsmfile/92177512.DBB
      Volume Location:
      Command:
```

We always recommend to backup and save the volume history file and the device configuration file every time after the TSM database backup. This is because the history file will then contains records of the backup. Use the following command to back them up:

```
tsm: TSM>backup volhist
tsm: TSM>backup devconf
```

These commands, together with the **backup db** command, can be scheduled using the TSM scheduler, in crontab of AIX or in Windows' scheduler.

The volume history and device configuration are flat files that can be copied to another directory, a tape or diskette. To find out the location of these two files, look into the file dsmserv.opt or issue the following command:

```
tsm: TSM>query option
```

The parameters VOLUMEHistory and DEVCONFig shows the location and name of the files. Because the content of these files change as new configuration are added, it is a good practice to keep a separate copy of each file somewhere else.

To delete a TSM database backup, use the **delete volhist** command. Volumes for the most recent database backup series are not deleted. This is because TSM always keep the very last backup of the database, regardless of when it was performed.

Example 3-11 shows the output of deleting database backup which is older than 5 days. This command deletes the actual backup file if it is on a file device class.

Example 3-11 Deleting database backup output

```
tsm: TSM>delete volhist todate=-5 type=dbbackup
```


Do you wish to proceed? (Yes (Y)/No (N)) y
ANR2467I DELETE VOLHISTORY: 1 sequential volume history entries were
successfully deleted.

When a TSM backup is offsite and is deleted, the volume becomes scratch depending on the reuse delay parameter, Delay Period for Volume Reuse. After the delay period, the scratch volume can be reused again.

Copy storage pools

The copy storage pools can be useful in two ways:

- ▶ Protect the data by having a duplicate copy.
- ▶ Bring the copy volumes offsite for disaster recovery purposes.

Copy storage pool always uses sequential devices. It is not necessary to use the same device class as the primary storage pool. A multiple primary storage pools can backup to the same copy storage pool.

If there are more than two libraries managed by TSM, the copy storage pool can be defined to use the slower but higher capacity devices, such as the LTO, which provides up to 200 GB of compressed storage in just one tape. Compared to a size of 5.2 GB of a CDROM, more than 38 CDROM data can be put into one LTO tape.

The following is a sample command to define a copy storage pool with name OD01YRCOPY using devclass ltodev:

```
tsm: TSM>define stgpool OD01YRCOPY ltodev pooltype=copy maxscratch=10  
OVFLocation=SiteA
```

To backup the primary storage pool to copy storage pool, use the following syntax:

```
tsm: TSM>backup stgpool primary_stgpool copy_stgpool
```

This should be scheduled to run at least once a day.

Mirroring TSM database and log volumes

To safeguard against disk failure, we can make use of either the operating system mirroring or TSM mirroring of database and log volumes.

To mirror TSM database or log volumes, follow the steps below:

1. Format TSM database or log volume using the **dsmfmt** command.
2. Issue **define dbcopy** command in TSM for database volume.

3. Issue **define logcopy** command in TSM for log volume.

A sample procedure is as shown in Example 3-12.

Example 3-12 Creating mirror copy of database and log volumes in TSM

At the operating system

```
#dsmfmt -m -db /tsmdb_mir/dbmir1.dsm 16  
#dsmfmt -m -log /tsmlog_mir/logmir1.dsm 8
```

In TSM

```
define dbcopy /tsmdb/db1.dsm /tsmdb_mir/dbmir1.dsm  
define logcopy /tsmlog/log1.dsm /tsmlog_mir/logmir1.dsm
```

Practical considerations

In order to have the latest entry in the database, we should backup the TSM database after backup of the primary storage pools is completed. Then we can send the copy storage pool volume together with the database backup for offsite storage.

Setting Delay Period for Volume Reuse in the storage pool sometimes can help in recovering data. Imagine DB2 database backup has been deleted and the volume containing those data is reused the next day, the data would be totally gone. If a reuse delay is set, such as 7 days, then TSM will not reuse the media until 7 days later. In the mean time, if it is found that we need the database backup again, then TSM database can be restored. The deleted DB2 database backup in TSM will be recognized as valid data as though no deletion has happened.

Most of the OnDemand installation make use of optical jukebox library as the primary archival storage, because it is faster to retrieve data from a CDROM than a tape. In this scenario, you need to consider whether the disaster recovery site has an optical jukebox or not. If LTO media is used, you need to consider if there is a LTO library offsite as well. This should be factor into your backup and recovery planning.

3.2.5 OnDemand configuration and definition backup

The OnDemand configuration should be backup. They include:

- ▶ OnDemand configuration files
- ▶ OnDemand application definitions
- ▶ OnDemand customized programs and resources

OnDemand configuration backup

On AIX, backup the following files in the /usr/lpp/ars/config directory:

- ▶ ars.ini
- ▶ ars.cfg
- ▶ ars.cache
- ▶ ars.dbfs

On Windows, the configuration is in the registry. The following steps show how to export the OnDemand registry setting into a file:

1. Launch regedit in Windows command window.
2. Expand **HKEY_LOCAL_MACHINE**.
3. Select **SOFTWARE**.
4. Select **IBM**.
5. Select **OnDemand for WinNT**.
6. Click **Registry** and select **Export Registry File**.

OnDemand definition backup

OnDemand definitions can be backup in many ways. It can be copied as a second copy in the same instance, export to a local drive or as a report as long as you have the permission to do so. The OnDemand definitions that can be copied or exported include all the categories that you could find in the administrative clients. You could export as a group of the same categories or individually. The categories are listed as follows:

- ▶ Users
- ▶ Groups
- ▶ Applications
- ▶ Application groups
- ▶ Storage Sets
- ▶ Folders
- ▶ Printers

Create duplicate copy of an OnDemand definition

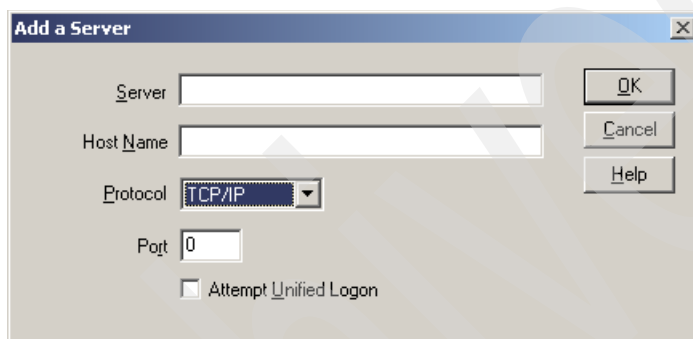
This is the simplest way to make a backup, although only make the duplicate on the same instance. To start, simply right-click the OnDemand definition that you want to duplicate and click **Copy**. A dialog will appear. Modify items such as Name or Description and click **OK**. A new OnDemand definition of the category having the same property will be created. This method is sufficient if you are about to make changes to some definition and want to keep the original for a reference.

Important: Do not delete an existing application group. Deleting application group will cause OnDemand to delete all the data which were previously loaded!

Create local server on OnDemand client

In order to export OnDemand definition to a local server, the local server must be created first. To create a local server, use the following steps:

1. Open the OnDemand administrative client.
2. Click **File**.
3. Select **New Server**. A dialog box similar to Figure 3-1 appears.

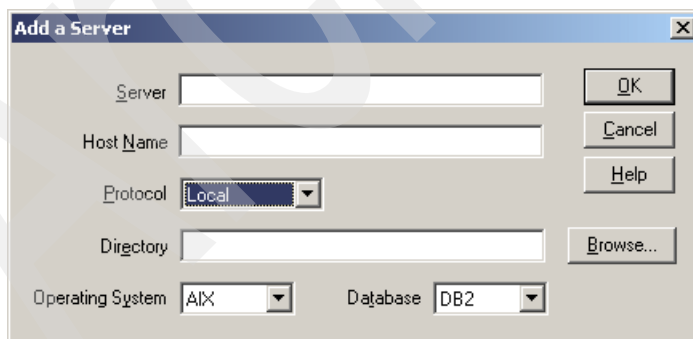


The 'Add a Server' dialog box contains the following elements:

- Server:** A text input field.
- Host Name:** A text input field.
- Protocol:** A pull-down menu currently showing 'TCP/IP'.
- Port:** A text input field with '0' entered.
- Attempt Unified Logon:** An unchecked checkbox.
- Buttons:** 'OK', 'Cancel', and 'Help' buttons are located on the right side.

Figure 3-1 Add a server

4. Choose Local under Protocol pull-down menu. A dialog box similar to Figure 3-2 appears.



The 'Add a Server' dialog box, configured for a local server, contains the following elements:

- Server:** A text input field.
- Host Name:** A text input field.
- Protocol:** A pull-down menu currently showing 'Local'.
- Directory:** A text input field with a 'Browse...' button to its right.
- Operating System:** A pull-down menu currently showing 'AIX'.
- Database:** A pull-down menu currently showing 'DB2'.
- Buttons:** 'OK', 'Cancel', 'Help', and 'Browse...' buttons are located on the right side.

Figure 3-2 Add a server with local protocol

5. Enter information for the name of the local server in the Server and Host Name boxes.

6. Enter the directory where you want the local server to be created at the Directory box.
7. Select the correct Operating System and Database where you want to export from.
8. Click **OK**.

A local server will be created in the directory specified. We can now export OnDemand definition into this local server.

Export OnDemand definition to other server

The easiest way to duplicate OnDemand definition to another OnDemand server is by exporting. OnDemand definition can be exported to another OnDemand instance or to local server residing on a local PC directory. In order to perform export, we need to have password for both the exporting server and the server to be exported to.

The steps to export definition is as follows:

1. Select a definition or a group of the same type of definition you want to export.
2. Right-click and choose **Export**.
3. A dialog box appears as in Figure 3-3 on page 73. An application group LoanData is chosen to be exported.
4. Choose a server/instance to be exported to under Server box.
5. Select Ignore Warnings if you want OnDemand to add an item regardless of any warnings encountered. Otherwise, OnDemand stops transferring the item when the first warning is encountered. If the item exists on the destination server, the export will fail.
6. If you do not want OnDemand to assign a storage set to the exported application group, select No Storage Set.
7. Click **Export**. OnDemand will export the selected definitions.

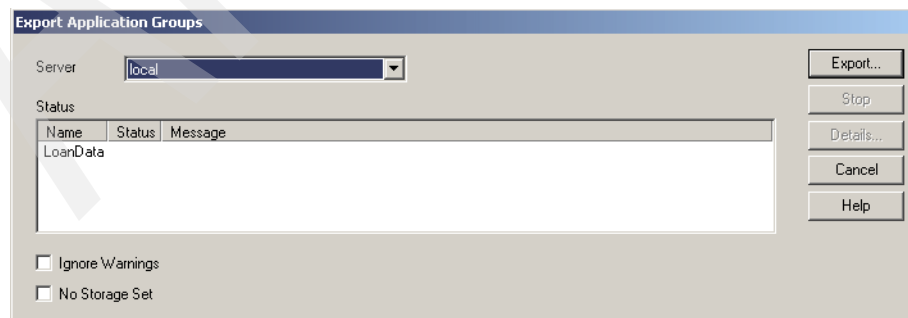


Figure 3-3 Export application groups

Generate summary/report on OnDemand definition

Alternatively, you can generate a report or summary of OnDemand definitions as follows:

1. On an OnDemand administrative client, right-click the definition that you want. This can be users, groups, application, application group, folder, storage sets or printer. Or, highlight a few definitions of the same type and right-click.
2. Choose **Report** for version 7.1.0 or **Summarize** for version 7.1.1.
3. A dialog box similar to Figure 3-4 on page 74 appears, displaying the options that you can choose.
4. Named a file to be created or take the default report file name under Name in the File Information section.
5. Click the information you want to include under the Summary Information section.
6. Click **Create**. This will create the report as named.

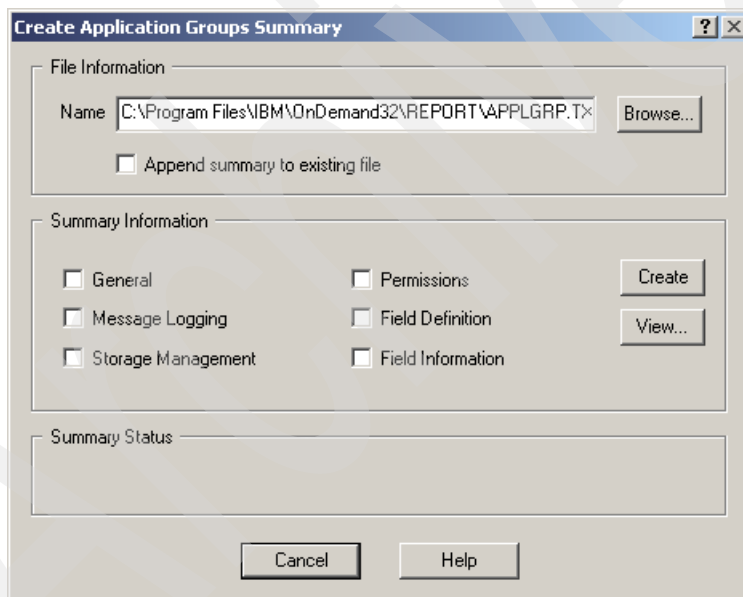


Figure 3-4 Create summary report for an application group

OnDemand resources and customized programs

If you use AFP heavily, there could be many resources such as formdef, pagedef or overlays. Depending on the set up of these files, they might not be stored in the rootvg or the main installation drive of Windows. It is important to keep one copy somewhere else.

If customized user exits are used, these files can be backed up as well. Typically, they are in the directory /usr/lpp/ars/exits in AIX.

3.3 Recovery plans under different scenarios

It depends on what has failed. Most of the time, we do not need to rebuild the entire OnDemand server or the database. In this section, we discuss what to recover under different scenarios based on the nature of failure which we have discussed Chapter 1, “Basic concepts” on page 3.

3.3.1 Human factor

A machine does not normally fail by itself. Usually, something or somebody causes it to fail. Human intervention, either planned or unplanned is one of the most common causes that might require a system to be recovered. A planned event could be a system or software upgrade; unplanned event could be plain carelessness.

System upgrade

In order to keep pace with technology, enjoy new features, and most important of all, stay supported, upgrade is necessary. During a system upgrade, the system may fail and restoration/recovery is necessary.

We highly recommend that before any system upgrade, always prepare at least one backup copy of the operating system.

If software upgrade is performed on Windows, and it fails, it is always easy to roll back by the Add/Remove program or the installation program itself will provide a way to uninstall. These programs can be used to roll back or remove the installed software.

If it fails when the Windows operating system is to be upgraded, then the only way to roll back will be restoration.

In AIX, during update of software, we can select **yes** for the option **COMMIT software updates?** This saves a copy of the currently installed software. In case of roll back, we need to select **Reject Applied Software Updates (Use Previous Version)** to remove the newly installed version. This is an easy and clean way to rollback to previous version.

In the case of system migration from one level to another, restoration of system backup is needed.

Human error

The unplanned intervention normally is due to system operator or user mistakes. Human error is hard to prevent because anyone who have access to the system in one way or the other can be careless one day. We discuss what to do under different types of human error, such as system files or applications were accidentally deleted.

System files

If someone accidentally remove system files, we can use the system backup to restore. If the extend of deletion is too big, it would be cleaner and faster to restore the entire operating system.

Note that restoring operating system does not mean losing OnDemand data. Data typically is not stored in Windows system files drives or the root volume group in Unix server.

Database files

The DB2 database can be tempered by any user with the right privilege. If such case happens, the entire database will need to be restored to a consistent state.

OnDemand definitions

OnDemand users with the appropriate rights can delete or update configuration or data. Table 3-2 shows the default permissions of four user types in OnDemand. These are the default rights. User can be permitted to do more or less operations by selecting options under Authority during creation or in the Permissions tab of groups, application groups and folders.

Table 3-2 Different user types and their default rights

user type	user	group	application/ application group	Folder
User	no	no	no	no
User Administrator	yes	no	no	no
Application Group/Folder Administrator	no	no	yes	yes
System Administrator	yes	yes	yes	yes

As seen from Table 3-2, System Administrator is the most powerful user in OnDemand. Both System Administrator and Application Group/Folder Administrators has the default right to delete application group. In OnDemand, if application group is deleted, the data loaded into the application group will all be deleted.

User and User Administrator have less rights unless they are permitted to do so. User Administrator has the right to delete users. There is no actual data lost even if a user is deleted.

One way to minimize human error is to assign as little System Administrators and Application Group/Folder Administrators as possible. For those people who only need to create or delete users, they can be assigned to User Administrators instead.

If application group is deleted, you must restore the DB2 database, the cache, TSM databases and the media. If data is stored on disk or file device class, TSM will delete the data *immediately*!

The following are what need to be done; it is practically everything concerning the data:

- ▶ Restore the entire DB2 database to the point in time before the human error (updates might be lost).
- ▶ Restore the cache file systems if this is a cache-only application group. Data loaded later than the date of cache backup will be lost.
- ▶ If data was loaded into TSM together when data is loaded, the TSM database can be restored as well; however, expire inventory should not be run before the restore. If disk or file device type is used instead, restoration of the storage pool could be necessary.

If it is just the application, users, group, or folders that are deleted, we could import back the definition that was exported to the local server. If no backup copy has been exported to a local server, and there are reports containing the information, then they need to be keyed in manually.

Cache directories

The cache directories store the actual data that has not been expired from OnDemand. If the data is stored in Cache Only - Library Server storage set, then the data will be gone once the cache is deleted.

If TSM or other storage is used, the cache can be set to be a duplication of TSM or other storage after OnDemand migration is run. This does not mean that cache can be deleted. Manually deleting data from the cache directory will cause inconsistency which might cause cache validation to fail. Unless you know exactly what you have deleted, you can try to restore just those files and links that were removed; otherwise, it is best to restore the entire cache backup.

During restoration, always restore all the cache directories that were backup on the same day at the time when there was absolutely no activity in the cache to avoid inconsistency. In some cases, there is really no choice but to restore back

different versions of different cache directories, a lot of manual work need to be done on the cache directories. You must know what you are doing and it is not supported by OnDemand support.

3.3.2 Hardware failure

Normally, hardware failure can be detected early by the system. The system administrators need to check the error report regularly in order to prevent the error from manifesting itself and become a bigger problem. If there is hardware failure on the system, and the part that fails is critical to the operation of the system, the hardware must be repaired or replaced.

In most cases, the operating system sits on mirrored disk and there is a rare chance that both disks fail together. If this does indeed happens, certainly we need to restore the entire operating system. The same goes to cache, DB2 database or TSM database.

If something happens to the TSM primary storage pool's volume, we could use the command **restore volume** or **restore stgpool** to restore data from copy storage pool to primary storage pool.

High Availability helps to minimize downtime in this case because it duplicates resources and minimizes the single point of failure.

3.3.3 Transaction failure

Transaction failure refers to the abnormal situation which happens during database update. Since DB2 are equipped with logs, it could be roll back easily. Most of the time, the transaction does not require users to do anything unless it is really bad and needs to restore.

3.3.4 Disaster

Disaster refers to a complete site failure: the production system is no longer able to run. In such cases, unless another hot backup site is available, we need to restore the operating system, the database, cache, TSM and all the data residing in the TSM libraries from backup that are held offsite.

3.4 Recovery procedures

This section concentrates on how to perform recovery procedures for each different types of backup. Remember that restore typically overwrites what is already there. Always think *thrice* before performing a restoration!

3.4.1 Recovery of operating system

On AIX, to restore the system backup, follow the steps below. Alternatively, the mksysb tape is bootable and you should be able to restore from there easily.

Restore the system backup as follows:

1. Press **F1** or **1** for PCI machine or Turn key to Service for MCA machine.
2. Put the OS CD into CDROM drive, and reboot the System.
3. Press **F1** to Define the System Console.
4. Press **1** for English and Press **Enter**.
5. Press **3** for Start Maintenance Mode for System Recovery.
6. Put in the mksysb tape, and press **4** for Install from a system backup.
7. Follow the screen to choose your device. If you only have one tape drive, only one choice (choice **1**) is available. Choose **1**.
8. Press **1** for Start Install now with Default Settings.
9. Press **1** for Continue with Install. Wait for the system restoration to finish. Machine will be rebooted by itself.

On Windows, different software have different ways to restore the systems. Follow your company's tried and used method to restore the operating system.

Alternatively, install the operating system and start installing all the required software one by one. Install DB2 first, and put in the patches. Then, install OnDemand and TSM. After that, follow the steps in this section to restore each of the database or files into respective places.

3.4.2 TSM recovery procedure

The TSM recovery procedure will be discussed first before we discuss the DB2 recovery procedure. If DB2 backup is done via TSM, then the TSM server need to be ready before DB2 can restore database from TSM.

In this section, we look into the steps needed to restore TSM from available media until the server is up and running.

Restoring TSM configurations and database

Place the dsmserv.opt, dsmserv.dsk, volume history files, and the device configuration file into their respective directories. For example, dsmserv.opt and dsmserv.dsk reside in the /usr/tivoli/tsm/server/bin directory in AIX and \Program Files\Tivoli\tsm\server1 directory in Windows, as mentioned in 3.2.4, "TSM

backup procedure” on page 66. The location of the device configuration file and volume history file can be found inside the dsmserv.opt file.

From the dsmserv.dsk file, and information about the database size, determine how many database and log volume need to create.

In this example, assuming the configuration in Table 3-3 on page 80, we can create the database and log volume with **dsmfmt** command and run the **dsmserv format** command on the operating system to initialize the database and log volume.

Table 3-3 Sample configuration of TSM database and log volumes

Type of volumes	Number of volumes	Location	Size of each volumes	Names of volumes
TSM database volumes	2	/tsmdb	16MB	db1.dsm db2.dsm
TSM log volumes	2	/tsmdb	8MB	log1.dsm log2.dsm

Example 3-13 below shows the command to create and format TSM database and log volumes according to the table above.

Example 3-13 Command to create and format TSM database and log volumes

```
#dsmfmt -m -db /tsmdb/db1.dsm 16 /tsmdb/db2.dsm 16
#dsmfmt -m -log /tsmlog/log1.dsm 8 /tsmlog/log2.dsm 8
#dsmserv format 2 /tsmlog/log1.dsm /tsmlog/log2.dsm 2 /tsmdb/db1.dsm
/tsmdb/db2.dsm
```

Next, use the dsmserv restore command to restore the database to the latest or a specific date. The following command restore TSM database:

```
#dsmserv restore db todate=MM/DD/YYYY
```

Where, MM/DD/YYYY is the date a backup was created and you want to restore.

Recreating the devices in TSM

For newer version of TSM, other than the drive and library, there is a path parameter that we need to define. In addition, the serial number of the device is being recorded in the TSM device configuration file as well. In a disaster recovery situation, all these will have to be changed because different physical device is used. The optical library will need to be defined as TSM devices first before performing the following steps to configured them as libraries and drives.

The Example 3-14 show commands used to delete library and drives in TSM. Before deleting them, save the device configuration file in a safe place first because part of the define command is found inside of the file.

Example 3-14 Command to delete TSM libraries and drives

```
tsm: TSM>delete path TSM optdrv0 srctype=server desttype=drive library=archlib0
tsm: TSM>delete path TSM optdrv1 srctype=server desttype=drive library=archlib0
tsm: TSM>delete path TSM optdrv2 srctype=server desttype=drive library=archlib0
tsm: TSM>delete path TSM optdrv3 srctype=server desttype=drive library=archlib0
tsm: TSM>delete path TSM optdrv4 srctype=server desttype=drive library=archlib0
tsm: TSM>delete path TSM optdrv5 srctype=server desttype=drive library=archlib0
tsm: TSM>delete drive archlib0 optdrv0
tsm: TSM>delete drive archlib0 optdrv1
tsm: TSM>delete drive archlib0 optdrv2
tsm: TSM>delete drive archlib0 optdrv3
tsm: TSM>delete drive archlib0 optdrv4
tsm: TSM>delete drive archlib0 optdrv5
tsm: TSM>delete path TSM archlib0 srctype=server desttype=library
tsm: TSM>delete library archlib0
tsm: TSM>delete path TSM ltodrv0 srctype=server desttype=drive library=ltolib
tsm: TSM>delete path TSM ltodrv1 srctype=server desttype=drive library=ltolib
tsm: TSM>delete drive ltolib ltodrv0
tsm: TSM>delete drive ltolib ltodrv1
tsm: TSM>delete path TSM ltolib srctype=server desttype=library
tsm: TSM>delete library ltolib
```

After deleting all the drive definition, we can then re-define them in TSM as shown in Example 3-15. Alternatively, follow the device configuration files, but leave the serial number out. TSM will detect the new serial number.

Example 3-15 Command to define TSM libraries and drives

```
tsm: TSM>DEFINE LIBRARY ARCHLIB0 LIBTYPE=SCSI SHARED=NO
tsm: TSM>DEFINE PATH ONDEMAND ARCHLIB0 SRCTYPE=SERVER DESTTYPE=LIBRARY
tsm: TSM>DEFINE DRIVE ARCHLIB0 OPTDRV0 ELEMENT=6 ONLINE=Yes
tsm: TSM>DEFINE DRIVE ARCHLIB0 OPTDRV1 ELEMENT=1 ONLINE=Yes
tsm: TSM>DEFINE DRIVE ARCHLIB0 OPTDRV2 ELEMENT=2 ONLINE=Yes
tsm: TSM>DEFINE DRIVE ARCHLIB0 OPTDRV3 ELEMENT=3 ONLINE=Yes
tsm: TSM>DEFINE DRIVE ARCHLIB0 OPTDRV4 ELEMENT=4 ONLINE=Yes
tsm: TSM>DEFINE DRIVE ARCHLIB0 OPTDRV5 ELEMENT=5 ONLINE=Yes
tsm: TSM>DEFINE PATH ONDEMAND OPTDRV0 SRCTYPE=SERVER DESTTYPE=DRIVE
LIBRARY=ARCHLIB0 DEVICE=/dev/rop0 ONLINE=YES
tsm: TSM>DEFINE PATH ONDEMAND OPTDRV1 SRCTYPE=SERVER DESTTYPE=DRIVE
LIBRARY=ARCHLIB0 DEVICE=/dev/rop1 ONLINE=YES
tsm: TSM>DEFINE PATH ONDEMAND OPTDRV2 SRCTYPE=SERVER DESTTYPE=DRIVE
LIBRARY=ARCHLIB0 DEVICE=/dev/rop2 ONLINE=YES
tsm: TSM>DEFINE PATH ONDEMAND OPTDRV3 SRCTYPE=SERVER DESTTYPE=DRIVE
LIBRARY=ARCHLIB0 DEVICE=/dev/rop3 ONLINE=YES
```

```
tsm: TSM>DEFINE PATH ONDEMAND OPTDRV4 SRCTYPE=SERVER DESTTYPE=DRIVE  
LIBRARY=ARCHLIB0 DEVICE=/dev/rop4 ONLINE=YES  
tsm: TSM>DEFINE PATH ONDEMAND OPTDRV5 SRCTYPE=SERVER DESTTYPE=DRIVE  
LIBRARY=ARCHLIB0 DEVICE=/dev/rop5 ONLINE=YES  
tsm: TSM>DEFINE LIBRARY LTOLIB LIBTYPE=SCSI SHARED=NO  
tsm: TSM>DEFINE PATH TSM LTOLIB SRCTYPE=SERVER DESTTYPE=LIBRARY  
DEVICE=/dev/smc0 ONLINE=YES  
tsm: TSM>DEFINE DRIVE LTOLIB LTODRV0 ELEMENT=256 ONLINE=Yes  
tsm: TSM>DEFINE DRIVE LTOLIB LTODRV1 ELEMENT=257 ONLINE=Yes  
tsm: TSM>DEFINE PATH TSM LTODRV0 SRCTYPE=SERVER DESTTYPE=DRIVE LIBRARY=LTOLIB  
DEVICE=/dev/rmt1 ONLINE=YES  
tsm: TSM>DEFINE PATH TSM LTODRV1 SRCTYPE=SERVER DESTTYPE=DRIVE LIBRARY=LTOLIB  
DEVICE=/dev/rmt2 ONLINE=YES
```

After defining the library and drives, perform a **checkin libvol** command to let TSM search for all the volumes in the library. The following command can be used if all the volumes are placed inside the library. We assume volumes inside the library have a status of private; if there are scratch volumes, **checkin** the scratch first.

```
tsm: TSM>checkin libvol ltolib search=yes status=private checklabel=yes
```

If it is LTO library and it has a barcode reader, use the following instead:

```
tsm: TSM>checkin libvol ltolib search=yes status=private checklabel=barcode
```

After that, we recommend to run **audit library** to verify and ensure that the library volume inventory is consistent with the volumes that are physically in the libraries that were just redefined. Note that **audit library** command has to be run after successful completion of **checkin libvol** to add the volume into the server inventory.

Restore storage pool volumes into TSM

After check in all the copy storage pools volumes, we could use them to restore into primary storage pools. If you are rushing for time, this can be done later when the OnDemand server is up and running. If the system cannot find the data from the primary volumes, TSM will retrieve it from the copy storage pools volume when needed.

Before anything, use the following command first to set the state of all the primary storage pool volumes to destroy:

```
tsm: TSM>update vol * wherestg-XXX access=destroy
```

To restore storage pool volume, do a preview of the process to check that all the copy storage pool volumes needed are present. If multiple storage pools were backup into the same storage pool, we could separate them again using the **restore stgpool** command into different primary storage pool:

```
tsm: TSM>RESTORE STGP00L primary_pool copy=copy1pool preview=yes
```

We can change the name of the new primary storage pool as. In this case, you need to add a new load data node in the storage node definition.

Example 3-16 on page 83 shows the TSM activity log for the command. In the example, TSM complains some of the volume have the access mode of offsite or unavailable. In actual cases, the physical volumes could be inside the library but are in offsite or unavailable state. Before the actual restore, make sure to check those volumes mentioned. If they are indeed in the library, set the access status to readonly or readwrite; otherwise, these volumes need to be checked into the library and their access status need to be changed.

Example 3-16 TSM activity log output of restore storage pool command

```
ANR2017I Administrator ADMIN issued command: RESTORE STGP00L primary_pool
copy=copy1pool preview=yes
ANR0984I Process 19 for RESTORE STORAGE POOL (PREVIEW) started in the
BACKGROUND at 15:02:49.
ANR1231I Restore preview of primary storage pool primary_pool started as
process 19.
ANR2110I RESTORE STGP00L started as process 19.
ANR1255W Files on volume A00007 cannot be restored - access mode is
"unavailable" or "offsite".
ANR1255W Files on volume A00005 cannot be restored - access mode is
"unavailable" or "offsite".
ANR1255W Files on volume A00004 cannot be restored - access mode is
"unavailable" or "offsite".
ANR1255W Files on volume A00006 cannot be restored - access mode is
"unavailable" or "offsite".
ANR1255W Files on volume A00008 cannot be restored - access mode is
"unavailable" or "offsite".
ANR1234I Restore process 19 ended for storage pool ODAIX1.
ANR0986I Process 19 for RESTORE STORAGE POOL (PREVIEW) running in the
BACKGROUND processed 342810 items for a total of 797,617,413,793 bytes with a
completion state of SUCCESS at 15:09:58.
ANR1239I Restore preview of primary storage pool primary_pool has ended. Files
Restored: 342810, Bytes Restored: 797617413793.
```

When there is no error returning from the preview of restore storage pool command, it is time to run the actual command without preview.

Make sure there are enough volumes available for the restore to be done.

The restore storage pool command needs to run for each of the primary storage pool. Depending on the number of drives available in the libraries and the number of users accessing OnDemand server, there are some limits as to how many sessions can be run concurrently.

3.4.3 Recovery of DB2 database

In OnDemand, there is the ARSDB program which take care of DB2 database backup. There is no customized command in OnDemand to perform restoration of DB2 database. To restore OnDemand database, you need to use native DB2 commands.

If DB2 is backed up via TSM, we need to export the correct environmental variables first before running the restore command. Before doing that, make sure the valid dsm.opt.db2 and dsm.sys files are present. In AIX, the dsm.sys file is in the /usr/tivoli/tsm/client/ba/bin directory and the option file is in the /usr/tivoli/tsm/client/api/bin. For Windows, there is no dsm.sys file; all the information resides in the respective option file which normally resides in directory \Program Files\Tivoli\tsm\api.

After setting the environment variable and the DB2 profile, the **db2adutl** command can be used to query the database backup to TSM. The following Example 3-17 shows the output from the query.

Example 3-17 Checking DB2 backup on TSM

```
#export DSMI_CONFIG=/usr/tivoli/tsm/client/api/bin/dsm.opt.db2
#db2adutl query full
```

Query for database ARCHIVE

Retrieving FULL DATABASE BACKUP information.

```
1 Time: 20040819171104 Oldest log: S0000014.LOG DB Partition Number: 0
Sessions: 1
2 Time: 20040819170841 Oldest log: S0000012.LOG DB Partition Number: 0
Sessions: 1
3 Time: 20040819164129 Oldest log: S0000010.LOG DB Partition Number: 0
Sessions: 1
4 Time: 20040817120731 Oldest log: S0000008.LOG DB Partition Number: 0
Sessions: 1
```

Retrieving INCREMENTAL DATABASE BACKUP information.

No INCREMENTAL DATABASE BACKUP images found for ARCHIVE

Retrieving DELTA DATABASE BACKUP information.

No DELTA DATABASE BACKUP images found for ARCHIVE

Alternatively, if TSM is not used to backup DB2 database, the native DB2 command can be used. The following command shows the backup history of an archive instance:

```
#db2 list history backup ALL for archive
```


Preparation for restoration

Before restoration, it is important to save the current database configurations into files. This can be done with the following command, issued as the instance owner:

```
db2 get db cfg for archive
db2 get dbm cfg
db2 list db directory
```

You should run the above commands again after restoration is completed to make sure that the configuration settings are still the same.

Restoration of DB2 offline backup

Restoration of offline backup is the easiest and fastest way to recover a database. The database can be brought online without roll forward. In Example 3-18, we show the command to restore database. Note that if the **db2 connect** command return errors stating that the database is in roll forward pending state, then you need to issue the following command to roll forward without logs:

```
db2 rollforward db archive to end of logs and stop
```

Example 3-18 DB2 restoration of a offline full backup

```
#db2 restore db archive from /ondemand/backup_dir taken at 20040824131338 to
/ondemand/arsdb without rolling forward
SQL2539W Warning! Restoring to an existing database that is the same as the
backup image database. The database files will be deleted.
Do you want to continue ? (y/n) y
DB20000I The RESTORE DATABASE command completed successfully.
#db2 connect to archive
Database Connection Information

Database server      = DB2/6000 8.1.0
SQL authorization ID = ROOT
Local database alias = ARCHIVE
```

Restoration of DB2 online backup

Unlike full offline backup, restoration of full online backup requires logs to be roll forward. During the roll forward process, the log files have to be inside the primary log file directory. Example 3-19 shows the restoration of a full online backup.

Example 3-19 DB2 restoration of a full online backup

```
#db2 restore db archive from /ondemand/backup_dir taken at 20040824132739 to
/ondemand/arsdb
```

```

SQL2539W Warning! Restoring to an existing database that is the same as the
backup image database. The database files will be deleted.
Do you want to continue ? (y/n) y
DB20000I The RESTORE DATABASE command completed successfully.
/ondemand/data/USD>db2 connect to archive
SQL1117N A connection to or activation of database "ARCHIVE" cannot be made
because of ROLL-FORWARD PENDING. SQLSTATE=57019
/ondemand/data/USD>db2 rollforward db archive to end of logs and stop

```

Rollforward Status

```

Input database alias           = archive
Number of nodes have returned status = 1

Node number                    = 0
Rollforward status             = not pending
Next log file to be read       =
Log files processed            = S0000017.LOG - S0000019.LOG
Last committed transaction     = 2004-08-24-19.31.13.000000

```

```

DB20000I The ROLLFORWARD command completed successfully.

```

Restoration of DB2 database to another server

The steps to restore DB2 database to another server is similar as mentioned above. The only difference is that you need to create the instance first before restoration. Take note that on AIX, the server name is hard coded inside the db2nodes.cfg file. You must update the server name within the file to match your new server name. The db2nodes.cfg file is located under sqllib in the instance home directory. On Windows server, the same file is located under \Program Files\IBM\SQLLIB\DB2 by default.

Practical considerations

It is always important to make sure that the backup you are doing now works well during and after restoration. While you might have tested your DB2 restoration long ago during the setup time, it is a good practice to test it every time an upgrade of DB2 is performed on the system to make sure that it still works the same way as you have expected.

3.4.4 Recovery of cache directory

The cache directories can be recovered by simply copy back all the files that was backup previously:

- If the files are backup directly into tape, restore the tape content into the correct directories.

- If TSM is used to backup the large tar file, then first restore the tar file into a directory, then use **tar** command to extract all the files into the cache directories.

Example 3-20 on page 87 shows an example of restoring cache from TSM. It is better to perform the restoration manually to avoid writing into other location. In this example, the cache file is extracted from the root directory, because it was **tar** from root directory with the cache path. If another destination is preferred, then change directory to the desired location and extract from there.

Example 3-20 Restoring cache from TSM

```
export DSM_CONFIG=/usr/tivoli/tsm/client/ba/bin/dsm.opt.cache
export DSM_DIR=/usr/tivoli/tsm/client/ba/bin
dsmc retrieve /bkcache/archive.c1.20040818.tar
```

```
#note: when TSM retrieve has finished, tar the cache out to the destination
cd /
tar -xvf /bkcache/archive.c1.20040818.tar
```

Practical considerations

Once again, remember to restore the cache backup from the same day and preferably at the same time. If not, there might be discrepancies between the cache directories; you might run into error while running ARSMaint program.

3.4.5 After the restoration

After the database is restored successfully, we should perform database maintenance. The following are some programs that should be run after upgrade or restoration. In fact, they should be run everyday before database backup for the benefit of performance:

- **arsmaint -r**

This command runs database statistics, which causes the database manager to optimize application group index data and make access to information as efficient as possible.

- **arsdb -m**

This command runs maintenance on the OnDemand database and reorganizing the OnDemand system tables. The option refreshes the tables and optimizes access to information in the database.

- **arsdb -s**

This command runs database statistics. It is used to optimize indexes and tables to make access to information as efficient as possible. The default is all OnDemand system tables.

3.5 Problem determination

To determine the source of problems, you can ask users questions and check the appropriate logs and directories.

The questions to ask include:

- ▶ What have the users or administrators done recently? This helps to diagnose the problem and is a good starting point.
- ▶ What error messages are encountered and under what circumstances do they occur?

Check the following logs and directories during the time when the error was encountered:

- ▶ OnDemand system log. This usually provides a very good indication of what and where the problem is.
- ▶ TSM activity log. The library could be out of database space, log space or media. All these errors will be recorded in the TSM activity log. The command to check is **query actlog**.
- ▶ Check the operating system error log and event log for errors or failures. These logs normally report hardware failure or file system problems.
- ▶ Check that all the directories are in place and are not missing. These include database directories, the cache directories, and the log directories.
- ▶ Check any other log files generated by customized scripts. They may provide further clues.
- ▶ Check the space allocation of all the directories. They should not be full. In Unix, it is done via the **df** command.

High availability and business continuity for OnDemand Multiplatforms

In this chapter, we describe strategies and options to achieve high availability for an IBM DB2 Content Manager OnDemand (OnDemand) implementation for Multiplatforms. We also introduce the basic concepts and assumptions commonly used to define disaster recovery and business continuity.

The following topics are covered in this chapter:

- ▶ OnDemand high availability strategies and options
- ▶ Practical procedures for high availability
- ▶ Business continuity strategies and options

4.1 OnDemand high availability strategies and options

High availability is the requirement that the system and data be available during production hours, minimizing any downtime due to an unplanned outage. The focus is on *system uptime*. High availability is of vital importance as content-enabled applications evolve from the back office environment to client-facing, mission-critical applications.

4.1.1 High availability: Clustering

Cluster multi-processing is a group of loosely coupled machines networked together, sharing disk resources. In a *cluster*, multiple server machines cooperate to provide a set of services or resources to clients. There are several clustering software packages available primarily based on the operating system.

A node (a processor that runs the clustering software and the operating system) must have access to one or more shared external disk devices. A *shared external disk device* is a disk physically connected to multiple nodes. A node must also have internal disks that store the operating system and application binaries, but these disks are not shared.

In this section, we discuss various external disk devices. We introduce the two basic configuration options and provided an example for each configuration option.

External disk devices

In cluster configurations, the disk subsystems that are usually shared as the external disk storage are:

- ▶ 7122 Serial Storage Architecture (SSA) serial disk subsystems
- ▶ IBM 2105 Enterprise Storage Server (ESS)
- ▶ IBM FASTT family storage servers
- ▶ SCSI disks

Many shared external disk devices use Redundant Arrays of Independent Disks (RAID) technology to eliminate the device as a single point of failure.

IBM Serial Storage Architecture (SSA) disk subsystem

You can use SSA disk subsystems as shared external disk storage devices in cluster configurations. If you include SSA disks in a volume group that uses LVM mirroring (RAID 1), you can replace a failed drive without powering off the entire subsystem.

IBM 2105 Enterprise Storage Server (ESS)

ESS provides multiple concurrent attachments and sharing of disk storage for a variety of open systems servers. IBM @server pSeries processors can be attached, as well as other UNIX and non-UNIX platforms.

The ESS uses IBM SSA disk technology. ESS provides many availability features, such as RAID-5 to distribute parity across all disks in the array, sparing to allow you to assign a disk driver as a spare for availability, and others.

IBM FAST family storage servers

This is the storage server of choice for medium range storage consolidation and data sharing on multiple or heterogeneous server platforms. The FAST500 supports rapid universal access to vast quantities of data.

For a good reference about the different components of a cluster, review “Physical Components of an HACMP Cluster”, in *Concepts and Facilities Guide for HACMP V5.1*, SC23-4864. There, you can find the different components of an HACMP cluster and their definitions.

Configuration options

There are two basic types of cluster configurations:

- ▶ *Standby configurations (active/standby mode)* — These are the traditional redundant hardware configurations where one or more standby nodes stand idle, waiting for a server node to fail and leave the cluster. The advantage of the standby configuration is steady performance. The disadvantage is that redundant hardware is needed.
- ▶ *Takeover configurations (active/active mode)* — In this configuration, all cluster nodes do useful work, processing part of the cluster's workload. There are no standby nodes. Takeover configurations use hardware resources more efficiently than standby configurations since there is no idle processor. Performance can degrade after node detachment, however, since the load on the remaining node would increase.

In both of these configurations, an application runs on only one primary node at a time. If a primary node fails, the application that is running on the failed node fails over to another node and continues running.

During an OnDemand fail-over scenario, the application or application processes that move from the primary system to the backup includes several steps:

1. Stop and exit the failed process.
2. Release the resources.
3. Detach the disk array from the primary system.
4. Reattach the disk array to the backup system.

5. Check the disk and the file system.
6. Repair data integrity.
7. Gain all resources for running the process in the backup system.
8. Start the process in the backup system.

This fail-over process can take several seconds to several minutes after the fault is detected.

High availability example 1: Standby configuration

This configuration uses an idle second system (a standby) configured to take over the operation of the primary system in case of a hardware or software failure. Figure 4-1 on page 93 shows this configuration setup.

Both the active and standby nodes would have equivalent setups for the operating system and application programs. The active system would own the resource group (a grouping of resources, such as a file system or IP address, combined into a single logical entity for easier management). The shared disk resources (ESS disks) in the resource group would contain the configuration files, database, cache storage objects, and a file system for temporary storage. The HACMP monitoring link would track the health of the active system. Should a failure occur on the active system, HACMP would execute the scripts to have the standby system assume the role of the active system. The standby system would come online, take ownership of the resource group, and start the applications necessary to continue operations.

Note: For clarification, in our test scenario, the primary (active) server we used is called *euclid* and the secondary (standby) node is called *aristotle*.

The failover is transparent to the clients and requires no manual steps on their part. However, the administrator must provide scripts to start and stop the different components of the OnDemand system and any other applications that are within the control scope of HACMP.

While the failover or fallback may be transparent to end-user client's accessing the system, there are certain issues that you may need to address. For example, if you are loading data to your database and a crash or an outage occurs on the primary node, some additional "clean-up" steps will be necessary to complete prior to restarting the load process on the backup node. Refer to "Data loads" on page 147 for more details.

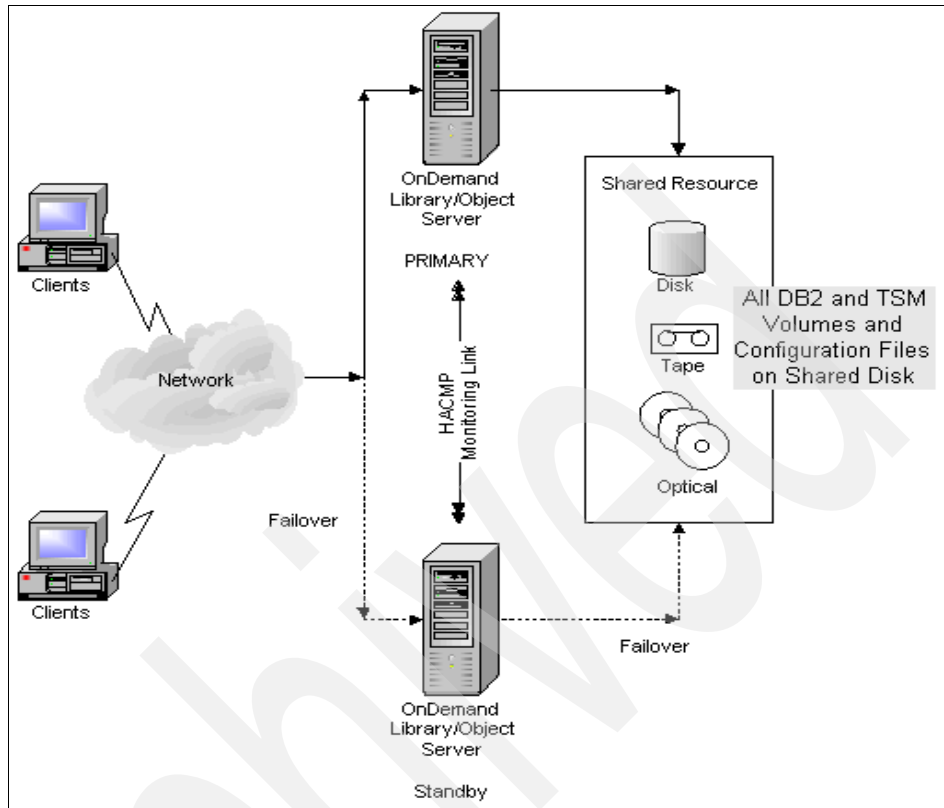


Figure 4-1 High availability standby site configuration

Tivoli Storage Manager (TSM) has an additional requirement that if the optical or tape libraries are involved, a SCSI, SAN switch, or twin-tail connection will be needed in order for both the primary and backup machines to have physical connectivity in the event of a host failure.

High availability example 2: Mutual takeover configuration

A mutual takeover configuration can make use of the resources on two systems that share the workload. The two systems can be configured as separate Library Server and Object Server. When both systems are healthy, one system acts as a dedicated Library Server, and the other acts as a dedicated Object Server. Each system monitors the health of each other. Either system can take over the operation of the other system in case of a hardware or software failure on the other system. Figure 4-2 on page 95 shows this configuration setup.

Both the primary and standby sites would have equivalent setups for the operating system and the application programs. Using this setup, two sets of

OnDemand configuration files would need to be available on each of the systems. One set with all of the information necessary to run a distributed configuration (for either the Library Server *or* the Object Server role), and one set with all of the information necessary to run a consolidated configuration (for both the Library Server *and* the Object Server roles).

For example, when both systems are healthy, the system running the OnDemand Library Server would own a resource group containing the database related file systems. The system running the OnDemand Object Server would own a resource group containing the storage related file systems. The HACMP monitoring link would track the health of both systems. Should a failure occur on either system, HACMP would execute the scripts to have the remaining healthy system assume the role of the failed system in addition to continuously running its existing services. The healthy system would take ownership of the failed system's shared resource. A second set of configuration files would be necessary to specify that the healthy node should be acting in both roles, as the Library Server and the Object Server. The scripts would also start the applications necessary to continue the operations.

Again, all OnDemand configuration files, the database, the data objects (defined cache file systems), and the TSM volumes should reside on the shared resource. At startup, each system owns their respective resource group on the shared resource: the Library Server owns the database related file systems and the Object Server owns the OnDemand cache and TSM related file systems.

Tivoli Storage Manager (TSM) has an additional requirement that if the optical or tape libraries are involved, a SCSI, SAN switch, or twin-tail connection will be needed in order for both the primary and backup machines to have physical connectivity in the event of a host failure.

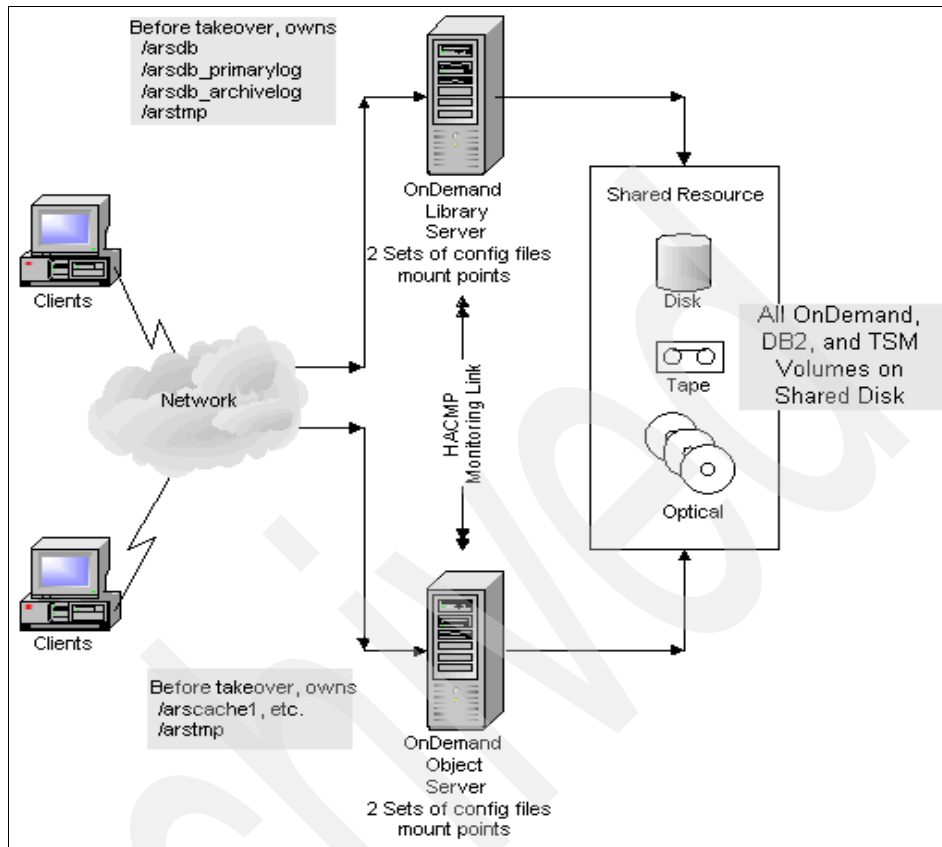


Figure 4-2 High availability mutual takeover site configuration

Note: An important consideration is that the hardware for each system needs to be sized to handle the operations of both the Library Server and the Object Server, despite the fact that ideally it will only be acting as one or the other.

Level of availability

High availability focuses on system uptime or the availability of the system. Table 4-1 on page 96 shows the level of availability for the two examples we presented earlier in this section. Please refer to Figure 1-2 on page 14 for a description of the levels of availability.

Table 4-1 High availability summary chart

HA example	Description	Level of availability
Example 1	Clustering with a standby configuration	Level 3: failover
Example 2	Clustering with a mutual takeover configuration	Level 3: failover

4.2 Practical procedures for high availability

This section includes steps for implementing the two previously mentioned configurations of a highly available OnDemand system. The first, addressing “High availability example 1: Standby configuration” on page 92, details the steps for creating a setup and configuration that includes all the necessary resources to run a combined Library Server and Object Server environment on two separate nodes. The second, addressing “High availability example 2: Mutual takeover configuration” on page 93, details the steps for creating a setup and configuration that includes all the necessary resources to run a distributed Library Server and Object Server environment (when the system is healthy), and also the necessary resources to run a combined Library Server and Object Server environment (when a failover has occurred).

The scenarios focus only on the high availability of the OnDemand server components. The configuration and setup of TSM server and the client components are beyond the scope of this redbook.

4.2.1 Test case scenario

The configuration that we use to setup the examples of this chapter consists of two logical partitions in different IBM @server pSeries 690 systems with AIX 5.2 ML-04. High availability is achieved using IBM High Availability Cluster Multi-Processing (HACMP) software Version 5.1 with APAR IY45695. We implemented IBM DB2 Content Manager OnDemand for AIX version 7.1.1.3 in conjunction with IBM DB2 Universal Database for AIX version 8.1 with fixpack 3.

Both servers share storage from an ESS Shark disk subsystem and are connected to it using two fibre channel adapters. To take advantage of this and also to meet a high level of availability, SDD is configured to use both adapters connected to the shared disks. Also, two different Gigabit ethernet adapters are connected to different switches to avoid a single point of failure.

To avoid using an RS/232 cable for TCP/IP or network point of failure, we use a disk to do the heart beat over the cluster. We are going to explain how to set up that in following sections.

An important concept of HACMP is the resource group. From the HACMP point of view (applied just to our example), a resource group is the logical container of:

- ▶ An IP address associated with a service such as the Library Server or the Object Server on which the clients connect to them
- ▶ The shared disk resources where the application programs (binaries) and data are stored
- ▶ The scripts that start or stop the applications

The resource groups that are configured in the practical examples that follow will be set as *Cascading* with the option *without fallback*. With this configuration, if a takeover occurs, the healthy (takeover) node, after acquired the resources, will run the load of the failing one. Once the failed server comes up again, the resources will remain on the takeover node until a manual intervention occurred to move the resource group back to the original node that it was issued. This configuration gives the administrator more control over the fallback of the resource group. Keep in mind that this configuration might not suit your system availability need or it might not cover some specific situations.

To learn more about resource groups and takeover relationships, refer to Chapter 1, “Cluster Resources and Resource Groups,” in *Concepts and Facilities Guide for HACMP V5.1*, SC23-4864.

4.2.2 Steps to configure example 1

The goal on this configuration is to setup a resource group that contains the necessary resources for the Library Server and the Object Server to run on the same node. The standby node will wait for an outage on the active server and then it will takeover the resources and continue providing services to clients or processing data.

Software installation

Before doing any configuration, all the required software must be installed on both nodes (euclid and aristotle) of the cluster.

Note: In our scenario, *euclid* is used as the primary (active) server, and *aristotle* is used as the secondary (standby) server.

The following is a list of the software which is installed on both euclid and aristotle for our test environment:

- ▶ AIX 5L™ Version 5.2 with maintenance level 2
- ▶ HACMP version 5.1 with APAR IY45695
- ▶ DB2 UDB for AIX version 8.1 with fixpack 3
- ▶ Content Manager OnDemand for AIX version 7.1.1.3

These products should be installed and their fix packs applied, but no further configuration should be done. *Do not create* a DB2 instance and do not create the OnDemand database. These configuration steps are explained in the following sections.

File system setup and user creation

We are going to define the user and group ID that OnDemand will use. Also we are going to define the shared volume groups, the logical volumes, and the file systems.

We recommend that all the configuration steps are executed initially on only one node. Once we verify that the first node is properly configured, we can replicate this information or send it to the remaining node. This helps us to ensure that no differences exist between the configuration of both nodes.

In order to create the file systems as an HACMP requirement, we need to make sure that the disks are shared (are available to) among the nodes that are part of the configuration. Ask your storage administrator and the operating system administrator about fulfilling this requirement and the data that you may need to make this configuration.

The file system layout of our scenario is shown in Table 4-2. For more information about the OnDemand file system layout, refer to Chapter 5, “Disk Storage” in *DB2 Content Manager OnDemand for Multiplatforms V8.3 Introduction and Planning Guide*, GC18-9236.

Table 4-2 OnDemand file system layout

File system name	Volume group name	Logical volume name	Physical volume name
/arsdb	odlsvg	arsdblv	vpath1
/arsdb_primarylog	odlsvg	arsbpllv	vpath1
/arsdb_archive	odlsvg	arsdballv	vpath1
/arsdb/SMS1	odlsvg	sms1lv	vpath1
/arsdb/SMS2	odlsvg	sms2lv	vpath1
/arstmp	odlsvg	arstmplv	vpath1

File system name	Volume group name	Logical volume name	Physical volume name
/home/archive	odlsvg	archivelv	vpath1
/arscache	odosvg	arscachelv	vpath2
/arsload	odosvg	arsloadlv	vpath2
/arsacif	odosvg	arsaciflv	vpath2

The disks that we use are Data Path Devices (SDDs) which give more bandwidth and redundancy. SDDs cover single point SCSI adapter failure in our configuration. For more information about SDD, go to:

<http://www.ibm.com/servers/storage/support/software/sdd/index.html>

Proceed with the following steps to setup file system setup and create users:

1. Create volume groups (VGs).

According to our scenarios, we execute the following commands:

```
/usr/sbin/mkvg -f -y'odlsvg' -s'64' '-n' -L'256' vpath1
/usr/sbin/mkvg -f -y'odosvg' -s'64' '-n' -L'256' vpath2
```

Remember that these parameters are suitable for our configuration. You might have to change them to suit your environment. Contact your system administrator if you have any questions.

Note: Make sure you select *not to activate* the volume group at system restart flag (flag -n); HACMP will take care of the activation.

2. Create logical volume, JFS log and file system.

Logical volumes and file system definitions must be the same on both nodes. We must avoid having different names of each device across nodes because of the following reason. You might have a previously system-defined name for logical volumes. When you import a volume group that contains a logical volume with the same name of another defined logical volume, the name will be changed automatically and the definition that is on the other node will not be consistent. As a result, you need to re-import the volume group again on the other node. Be careful with this.

According to our configuration and with the definitions that we have on Table 4-2 on page 98, we do the following:

- a. Create the logical volumes.

You can accomplish this with smit:

- i. Enter:

```
# smit mklv
```
- ii. Enter the name of the volume group in which you want to create the logical volume and complete the parameters for the definition. Figure 4-3 shows the values for our configuration.

Add a Logical Volume

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

	[Entry Fields]
Logical volume NAME	[arsdblv]
* VOLUME GROUP name	odlsvg
* Number of LOGICAL PARTITIONS	[16]
PHYSICAL VOLUME names	[vpath1]
Logical volume TYPE	[jfs2]
POSITION on physical volume	middle
RANGE of physical volumes	minimum
MAXIMUM NUMBER of PHYSICAL VOLUMES to use for allocation	[]
Number of COPIES of each logical partition	1
Mirror Write Consistency?	active
Allocate each logical partition copy on a SEPARATE physical volume?	yes
RELOCATE the logical volume during reorganization?	yes
Logical volume LABEL	[]
MAXIMUM NUMBER of LOGICAL PARTITIONS	[512]
Enable BAD BLOCK relocation?	yes
SCHEDULING POLICY for reading/writing logical partition copies	parallel
Enable WRITE VERIFY?	no
File containing ALLOCATION MAP	[]
Stripe Size?	[Not Striped]
Serialize IO?	no

Figure 4-3 Create a logical volume

Alternatively, you can use the command line to accomplish this. Example 4-1 lists the commands that you can execute to create all the logical volumes in our test scenario (the first is the one on Figure 4-3).

Example 4-1 Commands to create all the logical volumes for our test scenario

```
# mklv -y'arsdblv' -t'jfs2' odlsvg 16 vpath1
```



```
# mklv -y'arsdbpll' -t'jfs2' odlsvg 32 vpath1
# mklv -y'arsdball' -t'jfs2' odlsvg 32 vpath1
# mklv -y'sms1lv' -t'jfs2' odlsvg 32 vpath1
# mklv -y'sms2lv' -t'jfs2' odlsvg 32 vpath1
# mklv -y'arstmplv' -t'jfs2' odlsvg 8 vpath1
# mklv -y'archivelv' -t'jfs2' odlsvg 8 vpath1
# mklv -y'arscachelv' -t'jfs2' odlsvg 47 vpath2
# mklv -y'arsloadlv' -t'jfs2' odlsvg 235 vpath2
# mklv -y'arsaciflv' -t'jfs2' odlsvg 157 vpath2
```

The number of logical partitions depend on the size of the file system. Please look at Table 4-3 on page 103 for reference.

b. Create JFS2Logs.

You need to create at least one JFS2Log for each volume group. In addition, to mount a file system (JFS or JFS2), you need a JFS Log. For our example, we are going to define one log per volume group but this definition could differ from one environment to another because the log placement and the number of file systems that point to it impact the performance. For more information about JFS Log and performance impact, refer to Chapter 6.6.8 in *IBM Certification Study Guide - pSeries AIX System Administration*, SG24-6191.

You can accomplish this with smit:

i. Execute:

```
# smit mklv
```

ii. Type the name of the volume group in which you want to create the logical volume and complete the parameters for the JFS2 Log definition. Figure 4-4 on page 102 shows the values for our configuration.

Add a Logical Volume

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

	[Entry Fields]
Logical volume NAME	[loglvls]
* VOLUME GROUP name	odlsvg
* Number of LOGICAL PARTITIONS	[2]
PHYSICAL VOLUME names	[vpath1]
Logical volume TYPE	[jfs2log]
POSITION on physical volume	middle
RANGE of physical volumes	minimum
MAXIMUM NUMBER of PHYSICAL VOLUMES to use for allocation	[]
Number of COPIES of each logical partition	1
Mirror Write Consistency?	active
Allocate each logical partition copy on a SEPARATE physical volume?	yes
RELOCATE the logical volume during reorganization?	yes
Logical volume LABEL	[]
MAXIMUM NUMBER of LOGICAL PARTITIONS	[512]
Enable BAD BLOCK relocation?	yes
SCHEDULING POLICY for reading/writing logical partition copies	parallel
Enable WRITE VERIFY?	no
File containing ALLOCATION MAP	[]
Stripe Size?	[Not Striped]
Serialize IO?	no

Figure 4-4 Create a JFS2 log

Alternatively, you can use the command line to accomplish this. The following is a list of commands you can execute to create all the logical volumes on our test scenario (the first is the one on Figure 4-4):

```
# mklv -y'loglvls' -t'jfs2log' odlsvg 2 vpath1
# mklv -y'loglvos' -t'jfs2log' odosvg 2 vpath2
```

- iii. Format the JFS2Logs by issuing the **logform** command with each JFS2Log created and answer y when asked to destroy the logical volume. SeeExample 4-2.

Example 4-2 Formatting a JFS2Log volume

```
# logform /dev/loglvos
```

logform: destroy /dev/loglvos (y)? y

c. Create file systems.

File systems size had been defined when we choose the number of logical partitions on the creation of the logical volumes earlier. To summarize this, Table 4-3 shows the file system size that is necessary for our testing.

Note: The file system sizes specified here are based on our test environment; they fall far short of the requirements for a production environment.

Table 4-3 Shared file system sizes

File system name	Description	size
/arsdb	Library Server database	1 GB
/arsdb_primarylog	Database primary logs	2 GB
/arsdb_archivelog	Database archive logs	2 GB
/arsdb/SMS1	Application group tablespace container	2 GB
/arsdb/SMS2	Application group tablespace container	2 GB
/arstmp	Temporary space used by OnDemand	500 MB
/home/archive	DB2 instance directory and owner's home	500 MB
/arscache	OnDemand managed cache file system storage	3 GB
/arsload	OnDemand load daemon monitored directory	15 GB
/arsacif	Temporary index file location directory	10 GB

The OnDemand configuration files (in /usr/lpp/ars/config) will not be shared between the nodes. Instead, the configuration files that reside there will be kept synchronized by manually copying changes each time they have been updated.

With logical volumes created, the only thing that remains to be done is to configure the file systems over the predefined devices.

You can accomplish this with smit:

- i. Enter:
- ```
smit crjfs2lvstd
```

- ii. Press **F4** over “LOGICAL VOLUME name” entry and select the logical volume that you want to define the file system. Figure 4-5 shows an example.

| Add an Enhanced Journaled File System                                                   |                          |
|-----------------------------------------------------------------------------------------|--------------------------|
| Type or select values in entry fields.<br>Press Enter AFTER making all desired changes. |                          |
|                                                                                         | [Entry Fields]           |
| * LOGICAL VOLUME name                                                                   | arsdb +                  |
| * MOUNT POINT                                                                           | [/arsdb]                 |
| Mount AUTOMATICALLY at system restart?                                                  | no                       |
| PERMISSIONS                                                                             | read/write               |
| Mount OPTIONS                                                                           | <input type="checkbox"/> |
| Block Size (bytes)                                                                      | 4096                     |
| Inline Log?                                                                             | no                       |
| Inline Log size (MBytes)                                                                | <input type="checkbox"/> |

Figure 4-5 Add an enhanced JFS over a predefined logical volume

Alternatively, you can use the command line to accomplish this. Example 4-3 lists the commands that you can use to create all the file systems for our test scenario.

Example 4-3 Commands to create all the file system in our test scenario

```
crfs -v jfs2 -d'arsdblv' -m'/arsdb' -A 'no' -p'rw' -a agblksiz='4096'
crfs -v jfs2 -d'arsdbpllv' -m'/arsdb_primarylog' -A 'no' -p'rw' -a
agblksiz='4096'
crfs -v jfs2 -d'arsdballv' -m'/arsdb_archivelog' -A 'no' -p'rw' -a
agblksiz='4096'
crfs -v jfs2 -d'sms1lv' -m'/arsdb/SMS1' -A 'no' -p'rw' -a agblksiz='4096'
crfs -v jfs2 -d'sms2lv' -m'/arsdb/SMS2' -A 'no' -p'rw' -a agblksiz='4096'
crfs -v jfs2 -d'arstmplv' -m'/arstmp' -A 'no' -p'rw' -a agblksiz='4096'
crfs -v jfs2 -d'archivelv' -m'/home/archive' -A 'no' -p'rw' -a
agblksiz='4096'
crfs -v jfs2 -d'arscache' -m'/arscachelv' -A 'no' -p'rw' -a agblksiz='4096'
crfs -v jfs2 -d'arsload' -m'/arsload' -A 'no' -p'rw' -a agblksiz='4096'
crfs -v jfs2 -d'arsaciflv' -m'/arsacif' -A 'no' -p'rw' -a agblksiz='4096'
```

**Note:** Use “no” on the “Mount AUTOMATICALLY at system restart?” menu or add an option “-A no” if using the command line; HACMP will take care of the file system mounting when bringing the resource groups online.

In addition, be careful with the logical volume and file system relationship. Be sure that the logical volume you selected is the one that matches the selected mount point. In our example, this information is presented in Table 4-2 on page 98.

- iii. Mount all the created file system using the **mount** command and using the mount point as a parameter. In our example, we use:

```
mount /arsdb
```

3. Create users and permissions.

Once the file systems are created and mounted, we can create the users and user groups.

**Important:** User and user group IDs *must* be the same on all nodes that are part of the cluster. This is because on the standby (backup) node, the ownerships and the permissions should be the same as on the primary (active) one.

Any changes to user IDs, group IDs, passwords, or limits on the users that are involved in a HACMP configuration must be reflected on *all* the nodes that on the cluster.

For our setup, we use *archive* as the name of the Library Server database. Other parameters that we have to define for this environment are the AIX user IDs, groups, and the file systems that have to be created. These must be precisely defined and document for you to perform an appropriate HACMP definition. Table 4-4 lists the group and user ID defined in our system.

Table 4-4 Library Server IDs

| Description                  | Name    |
|------------------------------|---------|
| Library server database name | ARCHIVE |
| DB2 instance owner's group   | sysadm1 |
| DB2 instance owner           | archive |

You need to issue all of the following commands on *both* nodes (or to any nodes that are part of the cluster) because the definition of the users and

groups *need to be the same* on every participating node. We choose user and group ID number 340 for our example because it was available on both nodes.

- a. Create the group with the following command:

```
mkgroup id=340 sysadm1
```

- b. Create the user with the following command:

```
mkuser id='340' pgrp='sysadm1' groups='sysadm1' home='/home/archive' archive
```

- c. Set password for user *archive*:

```
passwd archive
```

For more information about the **mkgroup**, **passwd**, or **mkuser** commands, refer to the AIX documentation or man pages.

- d. Set ownership and permissions.

In order to enable OnDemand and DB2 to read, write, and execute on the defined file systems, we must change the ownership and permissions of some of the file systems. These commands must be run on the node where the file systems are mounted. In the following example, we show you what we did on *euclid*. Later in this chapter, when the file systems are mounted on the other node, *aristotle*, you need to issue the same commands.

**Important:** The following steps should be done on *only one* node now. These commands should be executed on the other node later.

Because you might have a different layout for file systems than what we have in our example, pay special attention to which files and directories that need permission changes. You can ask the system or database administrator about which files or directories need to be changed.

In our environment, we use the following commands:

```
chown archive:sysadm1 /arsdb /arsdb_archive\log \ /arsdb_primary\log
/arsdb/SMS1 /arsdb/SMS2
chmod 770 /arsdb /arsdb_archive\log /arsdb_primary\log \ /arsdb/SMS1
/arsdb/SMS2
```

## DB2 and OnDemand instance creation on the primary node

**Important:** Do not put any DB2 or Library Server's start or stop script on /etc/inittab; because when the server boots, the file systems will not be mounted and some services are not started. After the HACMP daemons start, these resources will be available and ready to use. The start and stop scripts should be managed by an *application server*. It is a cluster resource that manages the applications.

After all the users and groups have been created and the file systems mounted with the proper permissions in place, it is time to perform all the required configuration steps to have a ready to use Library Server and Object Server on this node. Before continuing with this part of the redbook, we highly recommend that you read *DB2 Content Manager OnDemand for Multiplatforms V8.3 Installation and Configuration Guide*, SC18-9232.

In our environment, user IDs and file systems are already created, but there is still no DB2 instance. So, the work to be done consists of the creation and customizing of the DB2 instance, the editing of OnDemand's configuration files, and the creation of a new Library Server database and system logging facility as follows:

1. Create a database instance.

With all file systems mounted on the primary node, log in as root and create a DB2 instance using the following command:

```
[root@euclid] # /usr/opt/db2_08_01/instance/db2icrt -u archive archive
```

This creates the DB2 instance on the /home/archive directory with default settings.

2. Configure the OnDemand instance.

Edit the OnDemand configuration files to reflect values consistent with a consolidated Library Server and Object Server configuration.

- a. Edit the ars.ini file. Example 4-4 shows the file we used in our example configuration.

*Example 4-4 Sample ars.ini file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.ini
[@SRV@_ARCHIVE]
HOST=
PROTOCOL=2
PORT=0
SRVR_INSTANCE=ARCHIVE
SRVR_INSTANCE_OWNER=root
SRVR_OD_CFG=/usr/lpp/ars/config/ars.cfg
```

```
SRVR_DB_CFG=/usr/lpp/ars/config/ars.dbfs
SRVR_SM_CFG=/usr/lpp/ars/config/ars.cache
```

---

- b. Edit the ars.cfg file. Example 4-5 shows the file we used in our example configuration.

*Example 4-5 Sample ars.cfg file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.cfg
ARS_NUM_LICENSE=10
ARS_LANGUAGE=ENU
ARS_SRVR=
ARS_LOCAL_SRVR=
ARS_NUM_DBSRVR=10
ARS_TMP=/arstmp
ARS_PRINT_PATH=/arstmp
ARS_DB_ENGINE=DB2
ARS_DB_IMPORT=0
ARS_DB_PARTITION=
DB2INSTANCE=archive
ARS_DB2_DATABASE_PATH=/arsdb
ARS_DB2_PRIMARY_LOGPATH=/arsdb_primarylog
ARS_DB2_ARCHIVE_LOGPATH=/arsdb_archive_log
ARS_DB2_LOGFILE_SIZE=1000
ARS_DB2_LOG_NUMBER=40
ARS_STORAGE_MANAGER=CACHE_ONLY
```

---

- c. Edit the ars.dbfs file. Example 4-6 shows the file we used in our example configuration.

If you are using SMS tablespaces to store your data tables, edit the ars.dbfs file to specify which file system(s) the tables will be created on.

*Example 4-6 Sample ars.dbfs file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.dbfs
ars.dbfs - OnDemand Database Filesystems Configuration File
#
DEFINITIONS:
Filesystem Tablespace Type (SMS)

/arsdb/SMS1 SMS
/arsdb/SMS2 SMS
```

---

- d. Edit the ars.cache file. Example 4-7 shows the file we used in our example configuration.

*Example 4-7 Sample ars.cache file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.cache
```



```

ars.cache - OnDemand Cache Configuration File

/arscache
```

---

3. Create the OnDemand database and test the instance.

a. Create the database by running the ARSDB program:

```
[root@euclid] /root # /usr/lpp/ars/bin/arsdb -I archive -gcv
```

In our installation, DB2 UDB maintains the archive log files on disk; so we enter 1 when prompted.

b. Create the system logging facility by running the ARSSYSCR program:

```
[root@euclid] /root # /usr/lpp/ars/bin/arssyscr -I archive -l
```

c. Start and activate the database:

```
[root@euclid] /root # /usr/lpp/ars/bin/arsdb -I archive -gkv
```

d. Start OnDemand:

```
[root@euclid] /root # /usr/lpp/ars/bin/arssockd start archive
```

e. Start the OnDemand Client for Windows. Logon to the Library Server and perform a default search of the system log folder. Ensure that you get a document hit list with recent activity.

## DB2 and OnDemand instance setup on the standby node

In order for the Library Server to work properly on the standby (backup) node, the node must be prepared with the same data and configuration as the active (primary node). Perform the configuration procedures as described in “DB2 and OnDemand instance creation on the primary node” on page 107 for the standby node. In addition, perform the following tasks to the standby node so that we have the same environment available on both machines:

1. Create a database instance.

A DB2 instance consists of the instance owner user ID with its home directory and profile scripts, the directory structure under the sqllib directory, the DB2 binaries, and some additional information that tells the DB2 installation which instances should be present in a particular node.

In our scenario, the user ID is defined in both nodes with the same ID number, the file system for the home directory that holds the profile scripts and the sqllib directory is configured to be shared, and the DB2 binaries are at the same level on both machines. Also, the permissions for the local directories on which this file system is mounted were configured to be the same.

The only missing part is the information that tells DB2 that an instance is present in this second node also. This is the information under the /var

directory. To create this information in the second node, use the standard DB2 utilities to create a new instance and then physically remove the created sqllib directory, leaving the information in /var intact. Of course, the information for the creation of this instance has to be the same as the information used for the creation of the original instance on the primary node.

To do this:

- a. Verify that the instance owner's home directory is not mounted on the standby node:

```
[root@aristotle] # umount /home/archive
```

- b. Verify that there is enough space on the file system that holds the /home/archive directory at this point. It should be enough with around 40 MB. You can verify this by looking at home; much space is used on the sqllib directory of the instance that has been created on the primary node.
- c. Log in as root and create a DB2 instance on the standby node using the following command:

```
[root@aristotle] # /usr/opt/db2_08_01/instance/db2icrt -u archive
archive
```

This creates the DB2 instance on the /home/archive directory with default settings.

- d. Remove the files created in the home directory of the instance owner:

```
[root@aristotle] # rm -Rf /home/archive/*
```

This procedure supplies the DB2 database with the information that there is an instance on the standby node. All other information (such as the sqllib directory that have just been deleted) will be accessed from the shared file system /home/archive once it is mounted.

## 2. Configure the OnDemand instance.

To ensure that the configuration files are identical, ftp the ars.ini, ars.cfg, ars.cache, and ars.dbfs from the primary node to the standby node.

## Setting up shared disks and LVM

In this section, we are going to go through the steps to configure the volume groups and file systems that were defined on the primary node (euclid) so they will be available on the standby node, aristotle, in case a takeover or failure happens on the primary node or if you decide to move the resource group to the standby node for maintenance or a programmed outage.

Note, we do not intend to cover the requirements, configuration steps, or best practices to set up shared disks in this redbook. The requirement is that the disks involved in the configuration can be accessed (not concurrent) from all the participating nodes. You should ask your system administrator about how to

accomplish this. We also do not cover any performance-related issues or recommendations. This is because performance tuning is generally specific to a particular installation. In this section, we only follow the steps that we need to accomplish the tasks for the high availability configuration in our sample environment.

For our scenario, assuming that the same disks can be accessed through the nodes participating on the cluster, we do the steps to configure and set up the disks, volume groups, and file systems to continue with the HACMP configuration on the standby node.

In order to make the data available to the standby node *aristotle*, we need to import the volume groups defined on the primary node *euclid* in the previous steps:

1. Stop all applications that might be using or accessing any file or directory on the involved file systems. Unmount the file systems and deactivate the volume groups on primary node as follows:
  - a. Unmount the file systems using the following commands:

```
umount /arsdb
umount /arsdb_primarylog
umount /arsdb_archivelog
umount /arsdb/SMS1
umount /arsdb/SMS2
umount /arstmp
umount /home/archive
umount /arscache
umount /arsload
umount /arsacif
```

**Tip:** If the following message appears:

```
[root@aristotle] /root # umount /arsdb
umount: 0506-349 Cannot unmount /dev/arsdb1v: The requested resource
is busy
```

This means that at least one process is using a file or a file structure. Make sure that the Library Server has been shut down and try again.

You can run the **fuser -u /dev/<lv name>** command to see which process is still using the system.

- b. Deactivate volume groups.

You must run the **varyoffvg** command with the volume group name as the parameter in order to deactivate it. You can get the volume group name by issuing the **lsvg** command. See an example on Figure 4-6 on page 112.

```
[root@aristotle] /root # lsvg
rootvg
odosvg
odlsvg
hbvg
[root@aristotle] /root # varyoffvg odosvg
[root@aristotle] /root # varyoffvg odlsvg
```

Figure 4-6 Deactivate a volume group

2. Import the volume groups definition on the backup (standby) node.

In order to get the file systems available on the backup node, the definitions must be imported from the volume groups (of the primary node) that contain the data and the permissions should be modified on the new (backup) node.

You will need to logon on the backup node. If you can, leave a session open to the primary node. Perform the following tasks:

a. Identify the disks.

We need to identify the disks that contain the different volume groups. To identify the volumes (if they are more than one) we need the physical volume ID (PVID) of the disks that are part of each volume group (VG). Run the **lspv** command on the primary node that were defined (euclid in our example), the second column contains a 16 digit (in hexadecimal) number which is the one we are looking for. Example 4-8 shows the output of the **lspv** command. You only need one PVID to import the volume group on the backup node.

Example 4-8 Output of **lspv** on the primary node where VGs are defined

|                                                                 |                  |        |        |
|-----------------------------------------------------------------|------------------|--------|--------|
| <pre>[root@euclid] /root # lspv   grep -E "odlsvg odosvg"</pre> |                  |        |        |
| vpath1                                                          | 0020390a9b4c600f | odlsvg | active |
| vpath2                                                          | 0020390a9b4ee0fe | odosvg | active |

Issue the same command on the other node, but using the PVIDs as the parameters of **grep**. In our example, the command that we issued to get the volume names of the physical volumes on the backup node is:

```
lspv | grep -E "0020390a9b4c600f|0020390a9b4ee0fe"
```

Example 4-9 shows the output of the command for our example.

Example 4-9 Output of **lspv** on the node where VGs are not yet defined

|                                                                                       |                  |      |  |
|---------------------------------------------------------------------------------------|------------------|------|--|
| <pre>[root@aristotle] /tmp # lspv   grep -E "0020390a9b4c600f 0020390a9b4ee0fe"</pre> |                  |      |  |
| vpath1                                                                                | 0020390a9b4c600f | None |  |
| vpath2                                                                                | 0020390a9b4ee0fe | None |  |

If the volumes have not been imported yet, you should see, on the third column the value as “None”. This is because no volume group has been associated into this node yet.

If you get no volumes as output or you get less volumes than expected, then you may need to read the PVID information on the disk and store it onto the Object Data Manager (ODM) of the AIX system. To do this run this command:

```
chdev -l hdiskX -a pv='yes'
```

Where hdiskX is the physical volume that you want to read the PVID.

**Important:** If you are using Subsystem Device Driver (SDD), ask your system administrator how to accomplish this because the SDD needs the PVIDs stored on the vpath's pseudo devices instead of the physical volumes. It is beyond the scope of this redbook for us to explain how SDD works or which steps you should need to accomplish to get the PVID on the vpath's instead of the hdisk's. We recommend contacting your operating system or storage administrator for this information. You can also find more information about SDD from the following URL:

<http://www.ibm.com/servers/storage/support/software/sdd.html>

b. Importing the volume groups into the secondary node.

Once we have identified the physical volumes that corresponds to the ones that we use to define the volume groups on the primary node, we are ready to import the definitions from the volumes to the AIX Object Database Manager (ODM). To do this, invoke the **importvg** command using this syntax:

```
importvg -y VolumeGroupName PhysicalVolume
```

Pay attention on the combination of VG name and physical disk selected. No data loss would happen but it may lead to some confusion.

Run this command to all the VGs involved in the configuration.

**Note:** In our scenario, we do not set up the major number because we are not using NFS with HACMP. If you are using NFS within the scope of HACMP, you need to import the volume group matching this number with the one on the other node. It is better to set up at the moment of the volume group creation or before the import is done on other nodes

For further explanation of this and reference information, see Chapter 5, “Using NFS with HACMP”, over “Shared Volume Groups” section in *Installation Guide for HACMP V5.1*, SC23-4861.

c. Mount the file systems.

Now, you should be able to mount the file systems on the backup node. To do that in our example we run:

```
mount /arsdb
mount /arsdb_primarylog
mount /arsdb_archivelog
mount /arsdb/SMS1
mount /arsdb/SMS2
mount /arstmp
mount /home/archive
mount /arscache
mount /arsload
mount /arsacif
```

If the file systems were mounted correctly, it is a good opportunity to verify and correct the owner, group, and permissions, because now there is a new mount point created. Refer to “Create users and permissions.” on page 105 for details about how to do this,. Note, you need to setup identical permissions and ownerships on the backup node.

d. Change volume groups to not activate at system restart.

Change the auto varyon attribute to no. The reason for doing this is because HACMP takes care of the volume activation in order to avoid a node lock on any volume group when it starts. Suppose that you do not change this attribute, and both nodes (primary and backup) are powered off. If you turn on the backup node first, it would activate the volume groups. If you turned on the primary server second, it would not be possible to mount the volume groups, because the backup node has already done it first.

With the volume groups active on the backup node, the **chvg** command must be run with the flag “-a n” (AutoOn set to no) in order to get what we are looking for.

In our example the commands that we run for this purpose are:

```
chvg -a 'n' od1svg
chvg -a 'n' odosvg
```

3. Configure host names and order resolution.

We need to configure the hosts and IP address on both nodes. We must ensure that the name resolution is the same across the cluster; any mismatch between the host name and the IP address must be avoided. To do so, perform the following tasks:

a. Setup name resolution order.

Update or create the /etc/netsvc.conf file to include the following syntax:

hosts=local,bind

This will force TCP/IP name resolution first to check the local /etc/hosts file for the name and then go to the name server if the name was not found. This is used to avoid the name resolution issues between the nodes on the cluster if a network problem occurs, for example.

- b. Update the /etc/hosts file with the IP addresses/host names that we use on this configuration. For our example, Table 4-5 shows the high availability IP addresses to be added to the hosts file. Also, there is some additional information that will be useful later in this chapter. It is a good practice to avoid getting confused.

Table 4-5 High availability IP addresses

| IP Address  | Host Name      |
|-------------|----------------|
| 9.30.130.66 | aristotle      |
| 9.30.130.71 | aristotle_boot |
| 10.30.10.66 | aristotle_stby |
| 9.30.130.67 | euclid         |
| 9.30.130.72 | euclid_boot    |
| 10.30.10.67 | euclid_stby    |

**Note:** For a reference of service, boot, standby adapters, and IP Address Take Over via IP replacement, see Chapter 2, “Cluster Networks”, in *Concepts and Facilities Guide for HACMP Version 5.1*, SC23-4864. Because we are going to use this concept to configure the cluster, we recommend reading this chapter.

For our examples, IP Address Take Over (IPAT) will be implemented without using aliases. The IPAT via IP alias is a feature introduced in HACMP 5.1. In order to do IPAT via IP replacement, we need at least two adapters per node; one adapter for the service and boot address, and the other one will be used as standby. This adapter works as a heartbeat transport; when a failover occurs, the IP address of the failing node is swapped to the standby adapter on the resource group acquiring the node.

4. Configure the IP addresses.

According to Table 4-5, we need to configure the IP addresses first on AIX. Both adapters must be configured with the boot and standby adapters. Service IP addresses are controlled by HACMP when the resource group with the file systems and applications is brought up. The primary service adapter

will be defined with the boot IP address, and the standby adapter to the standby IP address. Use the **smit chinnet** fast path to select and define the adapters according to your configuration. After that, set the host name using **smit hostname**.

**Important:** You must assign different subnets between the service (boot) and standby adapters; the netmask for all adapters must be the same to avoid communication problems between the standby adapters after an adapter swap. The communication problem occurs when the standby adapter assumes its original address but retains the netmask of the takeover address.

Example 4-10 is our network configuration on the secondary (standby) node.

*Example 4-10 Network configuration on secondary node*

---

| [root@aristotle] /root # netstat -in |       |          |                 |         |       |         |       |      |  |
|--------------------------------------|-------|----------|-----------------|---------|-------|---------|-------|------|--|
| Name                                 | Mtu   | Network  | Address         | Ipkts   | Ierrs | Opkts   | Oerrs | Coll |  |
| en0                                  | 9000  | link#2   | 0.2.55.33.1a.d8 | 296116  | 0     | 322689  | 1     | 0    |  |
| en0                                  | 9000  | 9.30.130 | 9.30.130.66     | 296116  | 0     | 322689  | 1     | 0    |  |
| en2                                  | 9000  | link#3   | 0.2.55.33.1e.ec | 346683  | 0     | 174249  | 1     | 0    |  |
| en2                                  | 9000  | 9.30.130 | 10.30.10.66     | 346683  | 0     | 174249  | 1     | 0    |  |
| lo0                                  | 16896 | link#1   |                 | 1076861 | 0     | 1082652 | 0     | 0    |  |
| lo0                                  | 16896 | 127      | 127.0.0.1       | 1076861 | 0     | 1082652 | 0     | 0    |  |
| lo0                                  | 16896 | ::1      |                 | 1076861 | 0     | 1082652 | 0     | 0    |  |
| [root@aristotle] /root # hostname    |       |          |                 |         |       |         |       |      |  |
| aristotle                            |       |          |                 |         |       |         |       |      |  |

---

5. Start and activate the database:

```
[root@euclid] /root # /usr/lpp/ars/bin/arsdb -gkv
```

6. Start OnDemand:

```
[root@euclid] /root # /usr/lpp/ars/bin/arssockd
```

Once you start the OnDemand Client for Windows, log on to the Library Server and perform a default search of the system log folder. Ensure that you get a document hit list with recent activity.

### HACMP configuration steps

In this section we go through the different steps to configure HACMP 5.1 for “High availability example 1: Standby configuration” on page 92. We assume that you have configured and tested the OnDemand product on all nodes that are part of the cluster. All steps must be performed using the root user. The examples and configuration steps are based on the configuration that we used in our scenario.



According to the different example scenarios that we want to show in this chapter, the first one will have one resource group defined because we want to show the active/standby configuration with the entire OnDemand application running on just one node of the cluster at a time. In HACMP terms, we only need one service address. We select the *euclid* server to be the primary. In our scenario, the only service address for the cluster is 9.30.130.67.

**Tip:** For a good HACMP reference relating to the questions about the configuration steps in this chapter, please read Chapter 2 and especially Chapter 3 from *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862. There is a very detailed explanation about the steps that we are presenting in this chapter. Most of the configuration steps are going to be done via the Extended Configuration Menus since they are more flexible and may fit best for different needs.

Perform the following steps for HACMP configuration:

1. Set cluster name and configure nodes.
  - a. Type **smitty hacmp**.
  - b. Select **Initialization and Standard Configuration** —> **Add Nodes to a HACMP Cluster**.
  - c. Enter the cluster name field with the name of the cluster that you want. On the new nodes field, enter the node names with the host name of the nodes that you are planning to work with. It is always a good practice (you will notice later) to have the host name and node names to be the same. Figure 4-7 shows the smit panel completed with our configuration example.

Configure Nodes to an HACMP Cluster (standard)

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

|                                              |                    |
|----------------------------------------------|--------------------|
|                                              | [Entry Fields]     |
| * Cluster Name                               | [odcluster]        |
| New Nodes (via selected communication paths) | [euclid aristotle] |
| Currently Configured Node(s)                 |                    |

Figure 4-7 Configure nodes to an HACMP cluster (standard) smit panel

Identify the cluster nodes and establish communication paths between them using the Configure Nodes to an HACMP Cluster menu options. Enter the cluster name and select the nodes (listed in */etc/hosts*) either by their names or by their IP addresses. HACMP uses this information to communicate with

the nodes that are involved in the cluster. Once this information is set and the communications paths exist, HACMP automatically runs a discovery operation to identify the basic components within the cluster.

2. Discover HACMP-related information.

After completing all the previous steps of importing the volume groups on the secondary node, setting up the host name for the nodes and configuring communication paths to the other nodes (Step 1), HACMP can collect information on the AIX files and ODM Configuration and make it available for the cluster configuration. Perform the following tasks:

- a. Type `smit hacmp`.
- b. Select **Extended Configuration** —> **Discover HACMP-related Information from Configured Nodes**.
- c. Wait until the process finishes.

3. Configure cluster topology and networks.

You can accomplish this by performing the following tasks:

- a. Configure Ethernet Network as follows:
  - i. Type `smit hacmp`.
  - ii. Select **Extended Configuration** —> **Extended Topology Configuration** —> **Configure HACMP Networks** —> **Add a Network to the HACMP Cluster**.
  - iii. Over the Predefined IP-based network types label, there are the supported network protocols. Select the one that corresponds to your configuration. In our example, we have ethernet adapters, so we choose **ether**.
  - iv. You can change the network name. We prefer to leave the default. On the netmask attribute from the panel, if the discovery process was successful, the netmask value should be completed with the right one. If you want to change it to another one, select one with F4 or type it. The netmask must be the *same* among all the adapters on a defined HA network; otherwise the synchronization process will fail.

In our example, because we have two adapters per node, we prefer not to use the IPAT via IP aliases, so we change the value to **"No"**.

Figure 4-8 on page 119 is the smit panel example for our network definition.

|                                                                                         |                          |
|-----------------------------------------------------------------------------------------|--------------------------|
| Add an IP-Based Network to the HACMP Cluster                                            |                          |
| Type or select values in entry fields.<br>Press Enter AFTER making all desired changes. |                          |
|                                                                                         | [Entry Fields]           |
| * Network Name                                                                          | [net_ether_01]           |
| * Network Type                                                                          | ether                    |
| * Netmask                                                                               | [255.255.255.128]        |
| * Enable IP Address Takeover via IP Aliases                                             | [No]                     |
| IP Address Offset for Heartbeating over IP Aliases                                      | <input type="checkbox"/> |

Figure 4-8 Add an IP-based network to the HACMP cluster smit panel

b. Configure serial networks.

It is very important to have a serial or at least two different networks for the heartbeat. We do not recommend to rely on one network for the heartbeat because if there is a problem or an outage on it, the cluster manager will assume that the nodes are down and it will start the failover process. In our example, we setup a serial line (rs232) over the nodes of the cluster to function as a heartbeat network. This configuration is very reliable and stable.

In Chapter 13, “Configuring an RS232 Serial Line” in *Planning and Installation Guide for HACMP Version 5.1*, SC23-4861, there is a complete reference on how to set up the hardware and AIX in order to get the serial lines available to be used as heartbeat devices. Also you can test them following the instructions on section “Testing the Serial Connection” in the same chapter of the book.

To configure the devices, follow these steps:

i. Configure serial network:

Type **smit hacmp**.

Select **Extended Configuration** —> **Extended Topology Configuration** —> **Configure HACMP Networks** —> **Add a Network to the HACMP Cluster**.

Select rs232 in the Predefined serial device types section and choose the network name that you want.

ii. Configure communication devices.

In this step, we must define two communication devices, one from euclid to aristotle and one for the reverse one:

Enter **smit hacmp**.

Select **Extended Configuration** —> **Extended Topology Configuration** —> **Configure HACMP Communication Interfaces/Devices** —> **Add Communication Interfaces/Devices** —> **Add Predefined Communication Interfaces and Devices** —> **Communication Devices**.

Select network name according to the one created on the previous step. Complete the entries, according to Table 4-6.

Table 4-6 Serial device communications

| Device name      | Device path | Node name | Comment                                           |
|------------------|-------------|-----------|---------------------------------------------------|
| euclid_serial    | /dev/ttyN * | euclid    | Serial device definition from euclid to aristotle |
| aristotle_serial | /dev/ttyN * | aristotle | Serial device definition from aristotle to euclid |

From Chapter 4, “Configuring Predefined Communication Devices to HACMP” in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862, we present the definitions for each field of this panel:

- Node name

The name of the node on which this network interface physically exists.
- Network name

A unique name for this logical network.
- Network interface

Enter the network interface associated with the communication interface (for example, en0).
- IP label/address

The IP label/address associated with this communication interface which will be configured in the network interface when the node boots. The pick list filters out IP labels/addresses already configured to HACMP.
- Network type

The type of network media/protocol (for example, ethernet, token ring, and fddi). Select the type from the predefined list of network types.

The previous steps from the HACMP configuration are the same for the two examples scenarios.

**HACMP setup for standby configuration (example 1)**

To setup HACMP for standby configuration for example 1, perform the following steps:

1. Configure HACMP topology, non-service and persistent adapters:
  - a. Configure non-service adapters:

- i. Type `smit hacmp`.
- ii. Select **Extended Configuration** —> **Extended Topology Configuration** —> **Configure HACMP Communication Interfaces/Devices** —> **Add Communication Interfaces/Devices** —> **Add Predefined Communication Interfaces and Devices** —> **Communication Interfaces**.
- iii. Select the appropriate network that you want to configure. In our example, the selection shows “net\_ether\_01 (9.30.130.0/25 10.30.10.0/25)”.

On this menu, select the IP labels or address, the node name on which will be available or configured, and optionally the preferred interface.

Figure 4-9 shows the information of the smit panel with the info of the boot address for node aristotle.

| Add a Communication Interface                                                           |                  |
|-----------------------------------------------------------------------------------------|------------------|
| Type or select values in entry fields.<br>Press Enter AFTER making all desired changes. |                  |
|                                                                                         | [Entry Fields]   |
| * IP Label/Address                                                                      | [aristotle_boot] |
| * Network Type                                                                          | ether            |
| * Network Name                                                                          | net_ether_01     |
| * Node Name                                                                             | [aristotle]      |
| Network Interface                                                                       | [en0]            |

Figure 4-9 Add a communication interface smit panel

You need to do the same steps for all the boot and standby adapters.

Alternatively, you can use the command line to define this. Example 4-11 lists the commands that we use to define the different interfaces (the first one is the same as Figure 4-9).

Example 4-11 Commands to define different interfaces

---

```

/usr/es/sbin/cluster/utilities/claddnode -a'aristotle_boot' \
:'ether':'net_ether_01' : : : -n'aristotle' -I'en0'
/usr/es/sbin/cluster/utilities/claddnode -a'euclid_boot' : 'ether':\
'net_ether_01' : : : -n'euclid' -I'en0'
/usr/es/sbin/cluster/utilities/claddnode -a'aristotle_stby' : 'ether':\
'net_ether_01' : : : -n'aristotle' -I'en2'
/usr/es/sbin/cluster/utilities/claddnode -a'euclid_stby' : 'ether':\
'net_ether_01' : : : -n'euclid' -I'en2'

```

---

- b. Configure HACMP topology, persistent adapters.

Because, for this first example, we are going to define only one resource group, so only one service adapter should be defined. For the other node (backup/aristotle), we decided to configure a persistent IP address. This IP along with the boot address using an alias are bound to the same adapter. The other adapter remains with the standby IP address in case a takeover occurs and the adapter must swap the IP to the service of the primary resource group. Perform the following steps:

- i. Type `smit hacmp`.
  - ii. Select **Extended Configuration** —> **Extended Topology Configuration** —> **Configure HACMP Communication Interfaces/Devices** —> **Configure HACMP Persistent Node IP Label/Addresses** —> **Add a Persistent Node IP Label/Address**.
  - iii. Select the node that was designated as the backup node. In our example, we choose aristotle. On the network name field, select the correct one for this type of network in the node IP label/address type or select the primary node as the input. In our example, we use aristotle.
2. Configure service adapter.

To do this, follow these steps:

- a. Enter `smit hacmp`.
- b. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resources Configuration** —> **Configure HACMP Service IP Labels/Addresses** —> **Add a Service IP Label/Address** —> **Configurable on Multiple Nodes** —> **Select Network Name**. See Figure 4-10.

Add a Service IP Label/Address configurable on Multiple Nodes (extended)

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

\* IP Label/Address

[euclid]

\* Network Name

net\_ether\_01

Alternate Hardware Address to accompany IP Label/A

ddress

[Entry Fields]

Figure 4-10 Add a service IP label/address configurable on multiple nodes

You may want to use an alternate hardware address to go with the IP address. This prevents some of the clients to get disconnected but it depends on individual application.

3. Configure application server.  
Perform the following tasks:
  - a. Enter `smit hacmp`.
  - b. Select **Extended Resource Configuration** —> **Configure HACMP Application Servers** —> **Add an Application Server**.
  - c. Enter the application server name, and the start/stop scripts. Figure 4-11 shows what we entered for our example.

Add Application Server

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

\* Server Name

\* Start Script

\* Stop Script

[Entry Fields]

[OnDemandAS]

[start\_od.ksh]

[stop\_od.ksh]

Figure 4-11 Add an application server smit panel

Our start and stop scripts along with their explanation are detailed in Appendix A, “Sample scripts and programs for the high availability scenarios” on page 317. In our test environment, these scripts are stored in the files `start_od.ksh` and `stop_od.ksh`.

4. Configure resource groups.
  - a. Configure a resource group by performing the following tasks:
    - i. Enter `smit hacmp`.
    - ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resource Group Configuration** —> **Add a Resource Group**.
    - iii. Select **Cascading** and enter the resource group name. In the participating node attribute, enter the node names in the *order* that you want the resource to be available on the cluster. This will determine the node where the resource group will be activated in a normal situation. For example, if you start the aristotle or backup node first, the resource group will not be activated there. It will be activated on the primary one when the cluster manager is activated. For a complete reference, see Chapter 4, “Configuring HACMP Resource Groups Using the Extended Path” in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862.

Figure 4-12 on page 124 shows what we entered for our example.

|                                                                                         |                    |
|-----------------------------------------------------------------------------------------|--------------------|
| Add a Cascading Resource Group (extended)                                               |                    |
| Type or select values in entry fields.<br>Press Enter AFTER making all desired changes. |                    |
|                                                                                         | [Entry Fields]     |
| * Resource Group Name                                                                   | [OnDemandRG]       |
| * Inter-Site Management Policy                                                          | [ignore]           |
| * Participating Node Names (Default Node Priority)                                      | [euclid aristotle] |

Figure 4-12 Add a cascading resource group (extended) smit panel

- b. Change resource group attributes as follows:
  - i. Type `smit hacmp`.
  - ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resource Group Configuration** —> **Change/Show Resources and Attributes for a Resource Group**.
  - iii. Select the resource group OnDemandRG. Enter the application server, service IP labels, volume groups in the corresponding entry fields. We changed the attribute “Cascading Without Fallback Enabled” to *true*. This means if a failover occurs, the resource group will be acquired by the backup node; but when the primary node rejoins the cluster, the resource group will remain on the backup node. As soon as a resource group movement via C-SPOC happens or the cluster manager on both nodes are stopped and restarted again, the primary node will regain acquisition of the resource group.

Figure 4-13 on page 125 shows the configuration of the resource group in our example.



| Change/Show All Resources and Attributes for a Cascading Resource Group                 |                          |
|-----------------------------------------------------------------------------------------|--------------------------|
| Type or select values in entry fields.<br>Press Enter AFTER making all desired changes. |                          |
| [TOP]                                                                                   | [Entry Fields]           |
| Resource Group Name                                                                     | OnDemandRG               |
| Resource Group Management Policy                                                        | cascading                |
| Inter-site Management Policy                                                            | ignore                   |
| Participating Node Names (Default Node Priority)                                        | euclid aristotle         |
| Dynamic Node Priority (Overrides default)                                               | <input type="checkbox"/> |
| Inactive Takeover Applied                                                               | false                    |
| Cascading Without Fallback Enabled                                                      | true                     |
| Application Servers                                                                     | [OndemandAS]             |
| Service IP Labels/Addresses                                                             | [euclid]                 |
| Volume Groups                                                                           | [odlsvg odosvg]          |

Figure 4-13 Change/show resources/attributes for a cascading resource group

## 5. Check configuration.

It is advisable to check the configuration made before continuing in order to avoid problems when the cluster is ready to start. You need to check the topology and resource group configurations. You can compare the obtained output with the desired output and check if everything is correct. In the previous sections, for example, we have shown tables of the file system layout and IP addresses of the cluster. This is a good practice because after you finish with the HACMP configuration, you can compare the planning data with the configured data.

Perform the following tasks:

### a. Check cluster definitions.

Run `/usr/es/sbin/cluster/utilities/cltopinfo`.

or

#### i. Type `smit hacmp`.

#### ii. Select **Initialization and Standard Configuration** —> **Display HACMP Configuration**.

The output is a brief description of what is defined for the entire cluster including how many nodes are defined, network, and nodes.

### b. Check topology configuration.

Run `/usr/es/sbin/cluster/utilities/cltopinfo -i`.

or

- i. Type `smit hacmp`.
- ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **Extended Topology Configuration** —> **Show HACMP Topology** —> **Show Topology Information by Communication Interface** —> **Show All Adapters**.
- iii. Check if all the adapters are properly configured with the matching IP addresses and host name from each node.

Figure 4-14 shows an example output of the our example 2 configuration.

```
COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

Cluster Description of Cluster: odcluster
Cluster Security Level: Standard
There are 2 node(s) and 2 network(s) defined
NODE aristotle:
 Network net_ether_01
 aristotle 9.30.130.66
 aristotle_boot 9.30.130.71
 aristotle_stby 10.30.10.66
 Network net_rs232_01
 aristotle_serial /dev/tty1
NODE euclid:
 Network net_ether_01
 euclid 9.30.130.67
 euclid_boot 9.30.130.72
 euclid_stby 10.30.10.67
 Network net_rs232_01
 euclid_serial /dev/tty1

Resource Group LibraryServerRG
 Behavior cascading
 Participating Nodes euclid aristotle
 Service IP Label euclid

Resource Group ObjectServerRG
 Behavior cascading
 Participating Nodes aristotle euclid
 Service IP Label aristotle
```

Figure 4-14 Display HACMP configuration smit output

- c. Check resource group definition:
  - i. Type `smit hacmp`.
  - ii. Select **Extended Configuration** —> **HACMP Extended Resource Group Configuration** —> **Show All Resources by Node or Resource Group** —> **Show Resource Information by Resource Group**.
  - iii. Select the resource group that you want to get the information about. Pay special attention to the participating node name tag because the node priority is defined here (if no custom method was configured). This tells HACMP what to do when a node joins (start cluster services) or rejoins the cluster about the resource group. For example, if you have a configuration similar to our example “*euclid aristotle*” for the resource group, OnDemandRG, if no cluster services are started on node *euclid* and we start *aristotle*, the resource group remains offline until we start *euclid* or the resource group was brought online manually (with C-SPOC). For more information, see Chapter 13, “Managing Resource Groups in a Cluster” in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862.
  - d. Another thing to check is the service IP label, volume group, file systems and node relationship. Unpredictable results might happen if the right configuration information was not added.
- 6. HACMP verification and synchronization of the configuration files.

We are now ready to verify our configuration and replicate it to the other node. This process will confirm that there are no errors in the configuration, and that the proper set up of the different operating system tasks were done (for example the IP addresses match those in the `/etc/hosts` file). If everything is correct, the configuration files will be sent to the remaining node.

To do this, follow these steps:

  - a. Enter `smit hacmp`.
  - b. Select **Extended Configuration** —> **Extended Verification and Synchronization**.

The default options as shown in Figure 4-15 on page 128 are the ones that you might choose in a normal situation as this one.

| HACMP Verification and Synchronization (Active Cluster on a Local Node)                 |                |
|-----------------------------------------------------------------------------------------|----------------|
| Type or select values in entry fields.<br>Press Enter AFTER making all desired changes. |                |
|                                                                                         | [Entry Fields] |
| * Emulate or Actual                                                                     | [Actual]       |
| Force synchronization if verification fails?                                            | [No]           |
| * Verify changes only?                                                                  | [No]           |
| * Logging                                                                               | [Standard]     |

Figure 4-15 HACMP verification and synchronization (active cluster on local node)

Pay special attention to the result of this operation. If the synchronization detects any errors, the configuration will not be passed to the other nodes in the cluster, and you will need to correct the errors and try this process again. After the synchronization process finishes, it is a good time to test if the different components that you set up are working as expected. For more information, please read Chapter 6 in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862.

### 4.2.3 Steps to configure example 2

In this section we present the steps to configure our example 2 test scenario (see “High availability example 2: Mutual takeover configuration” on page 93). We set up an active/active configuration where two nodes run different components of an OnDemand distributed Library Server and Object Server topology. On one The Library Server will be running on one node and the Object Server will be running on another node. Unlike the example presented in the active/standby configuration earlier, there is no primary or backup for the entire cluster definition. Each node is the backup for the other node, and in case any node fails, the other will acquire the failing node’s resource group and continue with the service.

Many steps taken in configuring this example are the same as in the previous section because the concepts are the same. When a difference or additional comments are required, they will be outlined in this section. Where setup is identical, a referral from 4.2.2, “Steps to configure example 1” on page 97 is made with the corresponding page number.

#### File System setup and user creation

The steps involving file system and user creation are the same for this configuration as they are for our example 1. Please refer to “File system setup and user creation” on page 98 for our example 1.

## DB2 and OnDemand instance creation on the Library Server

**Note:** For instructions on installing IBM DB2 Universal Database software, please refer to *IBM DB2 Universal Database Installation and Configuration Supplement for Version 8*, GC09-4837.

After all the file systems and users are created, it is time to perform all the required configuration steps to run just a Library Server on this node but with the added capability of running a consolidated Library Server and Object Server. Before continuing with the remaining section, we highly recommend that you read *DB2 Content Manager OnDemand for Multiplatforms V8.3 Installation and Configuration Guide*, SC18-9232 and *DB2 Content Manager OnDemand for Multiplatforms V8.3 Introduction and Planning Guide*, GC18-9236.

In our environment, user IDs and file systems are already created, but there is still no DB2 instance. We need to create and customize the DB2 instance, edit the OnDemand's configuration files, and create a new Library Server database and system logging facility as follows:

1. Create a database instance.

With all file systems mounted on the primary node, log in as root and create a DB2 instance using the following command:

```
[root@euclid] # /usr/opt/db2_08_01/instance/db2icrt -u archive archive
```

This creates the DB2 instance on the /home/archive directory with default settings.

2. Configure the OnDemand instance.

In this configuration, each node in the cluster has the potential to operate in three different roles. These roles are

- Library Server only
- Object Server only
- Library Server and Object Server

Each role requires a different set of configuration files.

The solution we implemented in our example specifies different *ars.ini* files depending on the role the node would be required to play in the cluster. We need *three different ars.ini files* and stored them in the /usr/lpp/ars/config directory. The HACMP startup script copies the appropriate *ars.ini* file at startup time based on the role the server will play in the cluster which depends on the state of the other node(s) in the cluster. See Example 4-12 on page 131, Example 4-13 on page 131, and Example 4-14 on page 131 for the three *ars.ini* files.

The different `ars.ini` files specify the appropriate and corresponding `ars.cfg` files to fulfill configuration requirements for each of the roles. Only *two different `ars.cfg` files* are required because the configuration file for the server running as a Library Server only is the same as the configuration file for the server running as a consolidated Library Server and Object Server. The second `ars.cfg` file is required for the case where the node's role is that of Object Server only. See Example 4-15 on page 132 and Example 4-16 on page 132 below.

Different `ars.dbfs` files are needed to fulfill configuration requirements for each of the roles. Only *two different `ars.dbfs` files* are required because the database file system configuration file for the server running as a Library Server only is the same as the database file system configuration file for the server running as a consolidated Library Server and Object Server. The `ars.ini` file will specify which to use. See Example 4-17 on page 133 and Example 4-18 on page 133 below.

Different `ars.cache` files are needed to fulfill configuration requirements for each of the roles. Only *two different `ars.cache` files* are required because the cache configuration file for the server running as an Object Server only is the same as the cache configuration file for the server running as a consolidated Library Server and Object Server. The `ars.ini` file will specify which is used. See Example 4-19 on page 134 and Example 4-20 on page 134 below.

Table 4-7 summarizes the type of configuration file, the specific files to be created and their purposes.

Table 4-7 Configuration files summary

| File type | File name     | Purpose                                                                                     |
|-----------|---------------|---------------------------------------------------------------------------------------------|
| ars.ini   | ars.ls.ini    | For Library Server role only                                                                |
|           | ars.os.ini    | For Object Server role only                                                                 |
|           | ars.ls_os.ini | For Library Server and Object Server role only                                              |
| ars.cfg   | ars.ls.cfg    | For Library Server role only and for the consolidated Library Server and Object Server role |
|           | ars.os.cfg    | For Object Server role only                                                                 |
| ars.dbfs  | ars.ls.dbfs   | For Library Server role only and for the consolidated Library Server and Object Server role |
|           | ars.os.dbfs   | For Object Server role only                                                                 |
| ars.cache | ars.ls.cache  | For Library Server role only                                                                |
|           | ars.os.cache  | For Object Server role only and for consolidated Library Server and Object Server role      |

Create the OnDemand configuration files by perform the following tasks:

- a. Create ars.ls.ini file by copy the default ars.ini file and edit it using Example 4-12 as example.

*Example 4-12 Sample ars.ls.ini file for Library Server only role*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.ls.ini
[@SRV@_ARCHIVE]
HOST=
PROTOCOL=2
PORT=0
SRVR_INSTANCE=ARCHIVE
SRVR_INSTANCE_OWNER=root
SRVR_OD_CFG=/usr/lpp/ars/config/ars.ls.cfg
SRVR_DB_CFG=/usr/lpp/ars/config/ars.ls.dbfs
SRVR_SM_CFG=/usr/lpp/ars/config/ars.ls.cache
```

---

- b. Create the ars.os.ini file by copy the default ars.ini file and edit it using Example 4-13 as an example.

*Example 4-13 Sample ars.os.ini file for Object Server only role*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.os.ini
[@SRV@_ARCHIVE]
HOST=
PROTOCOL=2
PORT=0
SRVR_INSTANCE=ARCHIVE
SRVR_INSTANCE_OWNER=root
SRVR_OD_CFG=/usr/lpp/ars/config/ars.os.cfg
SRVR_DB_CFG=/usr/lpp/ars/config/ars.os.dbfs
SRVR_SM_CFG=/usr/lpp/ars/config/ars.os.cache
```

---

- c. Create the ars.ls\_and\_os.ini file by copy the default ars.ini file and edit it using Example 4-14 as an example.

*Example 4-14 Sample ars.ls\_and\_os.ini file for consolidated system role*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.ls_and_os.ini
[@SRV@_ARCHIVE]
HOST=
PROTOCOL=2
PORT=0
SRVR_INSTANCE=ARCHIVE
SRVR_INSTANCE_OWNER=root
SRVR_OD_CFG=/usr/lpp/ars/config/ars.ls.cfg
SRVR_DB_CFG=/usr/lpp/ars/config/ars.ls.dbfs
SRVR_SM_CFG=/usr/lpp/ars/config/ars.os.cache
```

---

- d. Create the `ars.ls.cfg` file by copy the default `ars.cfg` file and edit it using Example 4-15 as an example.

*Example 4-15 Sample `ars.ls.cfg` file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.ls.cfg
ARS_NUM_LICENSE=10
ARS_LANGUAGE=ENU
ARS_SRVR=
ARS_LOCAL_SRVR=
ARS_NUM_DBSRVR=10
ARS_TMP=/arstmp
ARS_PRINT_PATH=/arstmp
ARS_DB_ENGINE=DB2
ARS_DB_IMPORT=0
ARS_DB_PARTITION=
DB2INSTANCE=archive
ARS_DB2_DATABASE_PATH=/arsdb
ARS_DB2_PRIMARY_LOGPATH=/arsdb_primarylog
ARS_DB2_ARCHIVE_LOGPATH=/arsdb_archivelog
ARS_DB2_LOGFILE_SIZE=1000
ARS_DB2_LOG_NUMBER=40
ARS_STORAGE_MANAGER=CACHE_ONLY
```

---

- e. Create the `ars.os.cfg` file by copy the default `ars.cfg` file and edit it using example in Example 4-16 as an example.

**Note:** The `ARS_SRVR` and `ARS_LOCAL_SRVR` parameter values should reflect the respective Library Server and Object Server values. In our example (from euclid), where we are setting up the configuration for the dedicated Object Server node, `ARS_SRVR` should be `aristotle` (the Library Server), and `ARS_LOCAL_SRVR` should be `euclid` (the Object Server). Although it is unlikely that this configuration should be necessary, it is a possibility for the nodes to switch the roles.

*Example 4-16 Sample `ars.os.cfg` file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.os.cfg
ARS_NUM_LICENSE=10
ARS_LANGUAGE=ENU
ARS_SRVR=aristotle
ARS_LOCAL_SRVR=euclid
ARS_NUM_DBSRVR=10
ARS_TMP=/arstmp
ARS_PRINT_PATH=/arstmp
ARS_DB_ENGINE=DB2
ARS_DB_IMPORT=0
ARS_DB_PARTITION=
```



```
DB2INSTANCE=archive
ARS_DB2_DATABASE_PATH=/arsdb
ARS_DB2_PRIMARY_LOGPATH=/arsdb_primarylog
ARS_DB2_ARCHIVE_LOGPATH=/arsdb_archive_log
ARS_DB2_LOGFILE_SIZE=1000
ARS_DB2_LOG_NUMBER=40
ARS_STORAGE_MANAGER=CACHE_ONLY
```

---

- f. Create the ars.ls.dbfs file by copy the default ars.dbfs file and edit it using Example 4-17 as an example.

If you are using SMS tablespaces to store your data tables, edit the ars.ls.dbfs file to specify which file system(s) the tables will be created on.

Example 4-17 Sample ars.ls.dbfs file

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.ls.dbfs
ars.dbfs - OnDemand Database Filesystems Configuration File
#
DEFINITIONS:
Filesystem Tablespace Type (SMS)

/arsdb/SMS1 SMS
/arsdb/SMS2 SMS
```

---

- g. Create the ars.os.dbfs file by copy the default ars.dbfs file and edit it using Example 4-18 as an example.

Because the Object Server does not need to have database related file systems to store tables, all lines in this file should be commented out.

Example 4-18 Sample ars.os.dbfs file

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.os.dbfs
ars.dbfs - OnDemand Database Filesystems Configuration File
#
DEFINITIONS:
Filesystem Tablespace Type (SMS)

/arsdb/SMS1 SMS
/arsdb/SMS2 SMS
```

---

- h. Create the ars.ls.cache file by copy the default ars.cache file and edit it using Example 4-19 on page 134 as an example.

Because the Library Server does not need to have cache storage related file systems to store data objects, all lines in this file should be commented out.

*Example 4-19 Sample ars.ls.cache file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.ls.cache
#
ars.cache - OnDemand Cache Configuration File
#
/arscache
```

---

- i. Create the ars.os.cache file by copy the default ars.cache file and edit it using Example 4-20 as an example.

*Example 4-20 Sample ars.os.cache file*

---

```
[root@euclid] /root # cat /usr/lpp/ars/config/ars.os.cache
#
ars.cache - OnDemand Cache Configuration File
#
/arscache
```

---

3. Create the OnDemand database and test the instance as follows:

- a. Create the database by running the ARSDB program:

```
[root@euclid] /root # /usr/lpp/ars/bin/arsdb -I archive -gcv
```

In our installation, DB2 UDB will maintain the archive log files on disk, so enter 1 when prompted.

- b. Create the system logging facility by running the ARSSYSCR program:

```
[root@euclid] /root # /usr/lpp/ars/bin/arssyscr -I archive -l
```

- c. Start and activate the database:

```
[root@euclid] /root # /usr/lpp/ars/bin/arsdb -I archive -gkv
```

- d. Start the OnDemand Library Server:

```
[root@euclid] /root # /usr/lpp/ars/bin/arssockd start archive
```

## **DB2 and OnDemand instance setup on the Object Server**

In order for the Library Server to work properly on the Object Server node (aristotle), it has to be prepared with the same data and configuration as the primary node. Perform the creation and the configuration procedures as described in “DB2 and OnDemand instance creation on the Library Server” on page 129 on the Object Server. In addition, perform the following tasks to the Object Server so that we have the same environment available on both machines and that this Object Server machine can be capable of running both the Library Server and the Object Server:

1. Create a database instance.

A DB2 instance consists of the instance owner user ID with its home directory and profile scripts, the directory structure under the sqllib directory, the DB2 binaries, and some additional information that tells the DB2 installation which instances should be present in a particular node.

In our scenario, the user ID is defined in both nodes with the same ID number, the file system for the home directory that holds the profile scripts and the sqllib directory is configured to be shared, and the DB2 binaries are at the same level on both machines. Also, the permissions for the local directories on which this file system is mounted were configured to be the same.

The only missing part is the information that tells DB2 that an instance is present in this second node also. This is the information under the /var directory. To create this information in the second node, use the standard DB2 utilities to create a new instance and then physically remove the created sqllib directory, leaving the information in /var intact. Of course, the information for the creation of this instance has to be the same as the information used for the creation of the original instance on the primary node.

To do this:

- a. Verify that the instance owner's home directory is not mounted on the standby node

```
[root@aristotle] # umount /home/archive
```

- b. Verify that there is enough space on the file system that holds the /home/archive directory at this point. It should be enough with around 40 MB. You can verify this by looking at home; much space is used on the sqllib directory of the instance that has been created on the primary node.
- c. Log in as root and create a DB2 instance on the standby node using the following command:

```
[root@aristotle] # /usr/opt/db2_08_01/instance/db2icrt -u archive
archive
```

This creates the DB2 instance on the /home/archive directory with default settings.

- d. Remove the files created in the home directory of the instance owner:

```
[root@aristotle] # rm -Rf /home/archive/*
```

This procedure supplies the information which tells DB2 on the Object Server node that there is a database instance. All other information (such as the sqllib directory just deleted) will be accessed from the shared file system /home/archive once it is mounted.

2. Configure the OnDemand instance.

To ensure that the configuration files are identical, ftp the ars.ls.ini, ars.os.ini, ars.ls\_and\_os.ini, ars.ls.cfg, ars.os.cfg, ars.ls.dbfs, ars.os.dbfs, ars.ls.cache,

and the `ars.os.cache` from the Library Server's `/usr/lpp/ars/config` directory, created in “DB2 and OnDemand instance creation on the Library Server” on page 129 to the Object Server node.

Replace the existing `ars.ini` file with the `ars.os.ini` file since *aristotle* will operate as a dedicated Object Server by default.

```
[root@aristotle] # mv /usr/lpp/ars/config/ars.os.ini \
/usr/lpp/ars/config/ars.ini
```

Edit the `ars.os.cfg` file. This is the only file which will be out of sync with the files on *euclid*. Change the `ARS_SRVR` to specify the Library Server (*euclid*) and the `ARS_LOCAL_SRVR` to specify the Object Server, which is the local machine (*aristotle*). Example 4-21 shows an example of the `ars.os.cfg` file in our scenario.

*Example 4-21 Sample ars.os.cfg file*

---

```
[root@aristotle] /root # cat /usr/lpp/ars/config/ars.os.cfg
ARS_NUM_LICENSE=10
ARS_LANGUAGE=ENU
ARS_SRVR=euclid
ARS_LOCAL_SRVR=aristotle
ARS_NUM_DBSRVR=10
ARS_TMP=/arstmp
ARS_PRINT_PATH=/arstmp
ARS_DB_ENGINE=DB2
ARS_DB_IMPORT=0
ARS_DB_PARTITION=
DB2INSTANCE=archive
ARS_DB2_DATABASE_PATH=/arsdb
ARS_DB2_PRIMARY_LOGPATH=/arsdb_primarylog
ARS_DB2_ARCHIVE_LOGPATH=/arsdb_archive_log
ARS_DB2_LOGFILE_SIZE=1000
ARS_DB2_LOG_NUMBER=40
ARS_STORAGE_MANAGER=CACHE_ONLY
```

---

3. Start the OnDemand object server:

```
[root@aristotle]# /usr/lpp/ars/bin/arsobjd archive
```

4. Start the OnDemand Client for Windows. Log on to the Library Server, and perform a default search of the system log folder. Ensure that you get a document hit list with recent activity. Retrieve a viewable document from the system log folder to verify that data objects can be accessed from the storage manager on the Object Server. If you need to create a viewable document, log on to the Library Server using the OnDemand Administrative Client for Windows and create a new user.

## Setting up shared disks and LVM

The steps involving file system and user creation are the same for this configuration as they are for example 1. Please refer to “Setting up shared disks and LVM” on page 110 for High availability example 1: Standby configuration.

## HACMP configuration steps

In this section, we go through the different steps to configure HACMP 5.1 for example 2. We assume that you have configured and tested the OnDemand Library Server running on *euclid* and the OnDemand Object Server running on *aristotle*. All steps must be performed using the root user. The examples and configuration steps are based on the configuration that we used in our scenario.

In the second scenario, we are going to show an active/active configuration. In this case we are going to set up distributed OnDemand components (Library Server and Object Server) on the two nodes. Under the normal circumstances, one node will be running the Library Server and the other node the Object Server. In case of failure or a controlled failover, one of the servers will acquire the resources of the other and continue working. In this case we need to define *two* service address, one for each component, because each one will have its own resource group associated.

**Tip:** For a good HACMP reference relating to questions about configuration steps in this book, please read Chapter 2 and especially Chapter 3 from *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862. There is a very detailed explanation about the steps that we present here in this chapter. Most of the configuration steps are going to be done via the Extended Configuration Menus because they are more flexible and may fit best for different needs.

The first steps involving HACMP configuration are the same for this configuration as they are for example 1. Please refer to “Set cluster name and configure nodes.” on page 117, “Discover HACMP-related information.” on page 118, “Configure cluster topology and networks.” on page 118 and complete these steps before continuing to the following section.

## HACMP setup for mutual takeover configuration (example 2)

Perform the following steps for HACMP setup for mutual takeover configuration:

1. Configure the HACMP topology, non-service and persistent adapters:
  - a. Configure non-service adapters.

To do this, perform the following steps:

    - i. Enter `smit hacmp`.

- ii. Select **Extended Configuration** —> **Extended Topology Configuration** —> **Configure HACMP Communication Interfaces/Devices** —> **Add Communication Interfaces/Devices** —> **Add Predefined Communication Interfaces and Devices** —> **Communication Interfaces**.
- iii. Select the appropriate network that you want to configure. In our example, the selection shows “net\_ether\_01 (9.30.130.0/25 10.30.10.0/25)”.
- iv. From this menu, you need to select the IP labels or address, the node name on which it will be available or configured, and optionally the preferred interface.

You need to perform the same steps for all the boot and standby adapters.

Alternatively, you can use the command line to define this. Example 4-22 lists the commands that we use to define the different interfaces.

*Example 4-22 Commands to define different interfaces*

---

```
/usr/es/sbin/cluster/utilities/claddnode -a'aristotle_boot' \
:'ether':'net_ether_01' : : : -n'aristotle' -I'en0'
/usr/es/sbin/cluster/utilities/claddnode -a'euclid_boot' : 'ether':\
'net_ether_01' : : : -n'euclid' -I'en0'
/usr/es/sbin/cluster/utilities/claddnode -a'aristotle_stby' : 'ether':\
'net_ether_01' : : : -n'aristotle' -I'en2'
/usr/es/sbin/cluster/utilities/claddnode -a'euclid_stby' : 'ether':\
'net_ether_01' : : : -n'euclid' -I'en2'
```

---

b. Configure service adapters.

To do this, follow these steps:

- i. Enter `smit hacmp`.
- ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resources Configuration** —> **Configure HACMP Service IP Labels/Addresses** —> **Add a Service IP Label/Address** —> **Configurable on Multiple Nodes** —> **Select Network Name**. See Figure 4-16 on page 139 as an example.

Add a Service IP Label/Address configurable on Multiple Nodes (extended)

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

[Entry Fields]

\* IP Label/Address

[aristotle]

\* Network Name

net\_ether\_01

Alternate Hardware Address to accompany IP Label/Address

Figure 4-16 Add a service IP label/address configurable on multiple nodes

You will need to go over this step twice for example 2 because now we are going to define *two* resource groups, *euclid* and *aristotle* respectively. One for the Library Server and the other for the Object Server. Each one has its own different IP address. We first add a service IP label/address for *euclid* (See Figure 4-10 on page 122) and then for *aristotle* (See Figure 4-16).

2. Configure application servers.

Remember that because we need two resource groups for this configuration, two applications servers must be defined. Table 4-8 summarizes the relationship between the different components of the HACMP configuration.

Table 4-8 Resource groups detailed

| Resource group name | Application server name | High priority node for RG | Service address name | Start/stop scripts          |
|---------------------|-------------------------|---------------------------|----------------------|-----------------------------|
| LibraryServerRG     | LibraryServerAS         | euclid                    | euclid               | start_ls.ksh<br>stop_ls.ksh |
| ObjectServerRG      | ObjectServerAS          | aristotle                 | aristotle            | start_os.ksh<br>stop_os.ksh |

- a. Add an application server using the following steps:
- i. Enter `smit hacmp`.

ii. Select **Extended Resource Configuration —> Configure HACMP Application Servers —> Add an Application Server**.

iii. Enter the application server name and the start/stop scripts. Start (startup) scripts are normally run when a node acquires a resource group. Stop scripts are normally run when the cluster service is stopped.

The purpose of the start scripts is to start the applications on a node necessary to fulfill the appropriate roles now required by the node after a change has occurred to the cluster. For example, if the node (euclid) that is running the dedicated Library Server (euclid) fails, the node (aristotle) that is running the dedicated Object Server will take over the role of the Library Server. The new role of aristotle will be to act as a consolidated Library Server and Object Server. When aristotle acquires the LibraryServerRG, the start script runs. It is responsible for stopping the Object Server process (ARSOBJD) and starting the Library Server and Object Server process (ARSSOCKD).

The purpose of the stop (shutdown) scripts is to stop and perform cleanup for the applications on a node anytime the cluster service is stopped, such as during a fallback. The stop scripts also start applications on the node necessary to fulfill the appropriate new roles required by that node. To continue using the example in the previous paragraph, *aristotle* is running as a consolidated Library Server and Object Server and the failure on *euclid* has been addressed. Now, *euclid* is ready to rejoin the cluster. Because we are using the *without fallback* option, a manual fallback would be initiated. The cluster service would be stopped and consequently the “stop Library Server” script would be run on *aristotle*. In our example, the “stop Library Server” script stops the Library Server and Object Server process (ARSSOCKD), determines that the node now needs to operate as a dedicated Object Server and runs the command to start the Object Server on *aristotle*. The LibraryServerRG would be returned to *euclid* and the “start Library Server” script would be run there.

In the case of a node failure, the stop scripts would likely not be run because a serious event has probably occurred which would prevent the system from continuing normal operations.

Figure 4-17 shows the names of the start and stop scripts we used in our scenario. Appendix A.2, “HACMP mutual takeover configuration scripts” on page 321 provide the source for and explanation of these scripts.

Add Application Server

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

\* Server Name

\* Start Script

\* Stop Script

[Entry Fields]

[LibraryServerAS]

[start\_ls.ksh]

[stop\_ls.ksh]

Figure 4-17 Add an application server smit panel



The same steps should be used to add the Object Server application server.

3. Configure resource groups.
  - a. Configure a resource group as follows:
    - i. Enter **smit hacmp**.
    - ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resource Group Configuration** —> **Add a Resource Group**.
    - iii. Add the two resource groups according to Table 4-8 on page 139.

**Attention:** Remember that the entry “Participating Node Names (Default Node Priority)” determines the activation resource group priority. Put the node that supposed to acquire it first and then put the backup node. For example, aristotle is the backup node of RG LibraryServerRG, so the entry for this RG should be “euclid aristotle”.

- iii. Add the two resource groups according to Table 4-8 on page 139.
    - b. Change resource group attributes as follows:
      - i. Enter **smit hacmp**.
      - ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resource Group Configuration** —> **Change/Show Resources and Attributes for a Resource Group**.
      - iii. Set the entry for “Cascading Without Fallback Enabled” to *true*.
      - iv. Define two resource groups according to Table 4-8 on page 139.
  4. Check configuration.

It is advisable to check the configuration made before continuing in order to avoid problems when the cluster is ready to start. Basically, you need to check the topology and resource group configurations. You can compare the obtained output with the desired output and check if everything is setup correctly. In the previous sections, we have shown tables of the file system layout and IP addresses of the cluster. This is a good practice because after you finish with the HACMP configuration, you can compare the planning data with the configured data.

Perform the following steps:

- a. Check the cluster definitions as follows:

Run **/usr/es/sbin/cluster/utilities/cltopinfo**.

Alternatively, perform the following steps:

  - i. Enter **smit hacmp**.

- ii. Select **Initialization and Standard Configuration** —> **Display HACMP Configuration**.

The output is a brief description of what is defined for the entire cluster, including how many nodes are defined, the network and nodes information.

- b. Check topology configuration as follows:

Run `/usr/es/sbin/cluster/utilities/cltopinfo -i`.

Alternatively, perform the following steps:

- i. Enter `smit hacmp`.
- ii. Select **Extended Configuration** —> **Extended Topology Configuration** —> **Show HACMP Topology** —> **Show Topology Information by Communication Interface** —> **Show All Adapters**.

Check if all the adapters are properly configured with the matching IP addresses and host name from each node.

Figure 4-18 on page 143 shows the output of our Example 2 configuration.

```
COMMAND STATUS

Command: OK stdout: yes stderr: no

Before command completion, additional instructions may appear below.

Cluster Description of Cluster: odcluster
Cluster Security Level: Standard
There are 2 node(s) and 2 network(s) defined
NODE aristotle:
 Network net_ether_01
 aristotle 9.30.130.66
 aristotle_boot 9.30.130.71
 aristotle_stby 10.30.10.66
 Network net_rs232_01
 aristotle_serial /dev/tty1
NODE euclid:
 Network net_ether_01
 euclid 9.30.130.67
 euclid_boot 9.30.130.72
 euclid_stby 10.30.10.67
 Network net_rs232_01
 euclid_serial /dev/tty1

Resource Group LibraryServerRG
 Behavior cascading
 Participating Nodes euclid aristotle
 Service IP Label euclid

Resource Group ObjectServerRG
 Behavior cascading
 Participating Nodes aristotle euclid
 Service IP Label aristotle
```

Figure 4-18 Display HACMP configuration smit output

- c. Check resource group definition as follows:
  - i. Enter **smit hacmp**.
  - ii. Select **Extended Configuration** —> **Extended Resource Configuration** —> **HACMP Extended Resource Group Configuration** —> **Show All Resources by Node or Resource Group** —> **Show Resource Information by Resource Group**.
  - iii. Select the resource group which you want to get the information about. Pay special attention to the participating node name tag because the node priority is defined here (if no custom method is configured). This

tells HACMP what to do when a node joins (start cluster services) or rejoins the cluster about the resource group. For example, if you have a configuration similar to our example “*euclid aristotle*” for the resource group OnDemandRG, if no cluster services are started on node *euclid* and we start *aristotle*, the resource group will remain offline until we start *euclid* or the RG was brought online manually (with C-SPOC). For more information, refer to Chapter 13 “Managing Resource Groups in a Cluster” in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862.

Another thing to check is the service IP label, volume group, file systems and node relationship. Unpredictable results might happen if the right configuration information is not added.

5. Check HACMP verification and synchronization of configuration files.

To do this, perform the following steps:

- a. Enter **smit hacmp**.
- b. Select **Extended Configuration** —> **Extended Verification and Synchronization**.

The default options as shown in Figure 4-15 on page 128 are the ones that you may choose in normal situations such as our scenario.

For more information, refer to Chapter 6 in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862.

## 4.2.4 HACMP post-configuration procedures

To start the cluster services on any of the nodes, issue the following command:

```
smit clstart
```

This starts the cluster daemon (CLSTRMGR) and brings up all the resources involved (IP address, file systems, OnDemand Library Server or Object Server depending on your configuration). On this screen, you can select the node or nodes that you want to start. You can leave all other options as the defaults. Pay special attention to “Reacquire resources after forced down” option because if the value is set to true it may fallback any resources from another node before you are ready; it could lead to unexpected circumstances.

To stop the cluster services on any of the nodes, issue the following command:

```
smit clstop
```

On this screen, you can select the node or nodes that you want to start. You can leave all other options as the defaults. Shutdown mode should always be forced

in normal situations. Remember that stopping the cluster server implies that the applications, file systems, and IP addresses will be deactivated or unconfigured.

Also, you can bring a resource online or offline from a node. You can even move that resource from one node to another. This could be accomplished by using the C-SPOC menus for the resource group management as follows:

1. Enter `smit hacmp`.
2. Select **System Management (C-SPOC) —> HACMP Resource Group and Application Management —> HACMP Resource Group and Application Management**.

**Important:** For more information, refer to Chapter 13 “Managing Resource Groups in a Cluster” in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862.

## Configure application monitoring

HACMP can monitor specified applications and take a desired action if failure or the death of a process is detected. You can choose *Process Application Monitoring* to detect the death of one or more OnDemand processes. This option uses the RSCT Event Manager which is a component that is embedded into the HACMP software. No custom scripts or additional configuration are required if you choose this option. However, this might not fit your needs. The other option, *Custom application monitoring* might fit. It checks the health of an application with a custom monitoring method at a specified polling interval.

In our examples, configuring the process monitoring option should be fine because the database (DB2 UDB) processes are always up if the application is in good health. We choose this option.

When a problem is detected by the monitor, HACMP attempts to restart the application a specified number of times. If the application cannot be restarted within this retry count, one of two actions can be taken:

- ▶ *Failover*: Causes the resource group containing the application to fall over to the node with the next highest priority according to the resource policy.
- ▶ *Notify*: Causes HACMP to generate a `server_down` event to inform the cluster of the failure. You can add pre and post events scripts to the resource group definition to control the cleanup or application stop.

First thing to do is to identify the correct process or processes names to be monitored. Refer to the Configuring Application Monitoring on Chapter 3 in *Administration and Troubleshooting Guide for HACMP V5.1*, SC23-4862. The manual guides you in choosing the correct process names.

You should use the processes that are listed in response to the **ps -e1** command, and not with **ps -f** command.

### ***Identify the processes to monitor***

If there is any doubt about the correct names, the following is a recommended short procedure to identify all the process names for your list:

1. Enter the following command:

```
ps -e1 | cut -c72-80 | sort > list1
```

2. Run the application server.

3. Enter the following command:

```
ps -e1 | cut -c72-80 | sort > list2
```

4. Compare the two lists by entering:

```
diff list1 list2 | grep \>
```

The result should be a list of all the processes spawned by the application server. You may choose not to include all of them in your process list, but you now have a complete and accurate list of possible processes to monitor.

### ***Set up the application monitor***

After the processes that need to be monitored are identified, perform the following steps to set them up to be monitored:

1. Enter **smitty hacmp**.
2. Select **Extended Configuration** —> **Extended Resources Configuration** —> **Configure HACMP Application Monitoring** —> **Define Process Application Monitor** —> **Add Process Application Monitor** —> Select the desired application server in which the application is started or stopped.
3. Enter the appropriate smit field values. A detailed explanation of each field is in the previously referenced chapter of the HACMP manual.

After you configure the monitor, you need to have the scripts on all the cluster nodes and synchronize HACMP.

You can suspend or restart the monitoring for maintenance purposes.

## **4.2.5 Failover tests and results**

After we performed all the configuration steps as explained earlier, we had our system up and running, ready to service user requests and to stand the failure of the Library Server and/or Object Server node(s).

## Client search and retrieve

Using the OnDemand Client for Windows, if performing a search when an outage occurs, cancel the search and resubmit the request. The client will not need to log on to the OnDemand server again. If the failover has completed prior to the search being resubmitted, the query will complete. Similarly, if retrieving a document when an outage occurs, the request will fail and can be resubmitted (again without performing another log on). As long as the failover has successfully completed, the document will be available to the client for retrieval.

## Data loads

When loading with the ARSLOAD program, there is a possibility that either the indexing or the load phase is interrupted when a node fails. If this happens, some “clean-up” will be necessary. If partially indexed files are present, they are likely the truncated files that will need to be removed in order to avoid the ARSLOAD daemon from trying to process the files again, which otherwise would cause an error. If a partial load has occurred, rows inserted during the partial load would need to be unloaded to avoid duplicate rows being inserted, and data objects need to be removed from storage. After that, the temporary load files would be removed so that the ARSLOAD daemon can begin processing the original input files again.

We recommend that these steps be manually performed after a failover. However, if you do decide to add these steps as part of your start scripts, make sure you have thoroughly tested all scenarios specific to your environment. These scripts must be tailored for each installation. Your HACMP and OnDemand infrastructure specialists need to spend time working together on these scripts as a critical part of the OnDemand HACMP implementation. The scripts we used in our test environment are provided only as a *reference*.

Refer to Appendix A.3, “OnDemand load daemon cleanup script” on page 329 for information about automating this process.

## OnDemand system maintenance

Should a failure occur when scheduled maintenance program (ARSMaint or ARSDB) is running, it is probable that the job has not run to completion. These jobs are generally scheduled and can either be run at the OnDemand system administrator’s discretion, or the job will run again the next time it is scheduled to run.

## 4.3 Business continuity strategies and options

The focus of business continuity is to provide a process to implement a disaster recovery plan aimed at *protecting against data loss* caused by a catastrophic system failure or natural disaster.

Geographic diversity is a factor greatly affecting the likelihood that businesses can recover from unplanned interruptions in service. Business continuity plans must account for power disruption and other natural or man-made disastrous events and architect alternate sites at a sufficient distance separation.

### 4.3.1 Multi-site solutions overview

Similar to high availability configuration options, business continuity solutions can be run in the following modes:

- ▶ *Active/standby mode* - These are the traditional redundant hardware configurations where one or more standby sites stand idle, waiting for an indication that the primary site is unavailable. The advantage of the active/standby configuration is steady performance. The disadvantage is that redundant hardware is needed.
- ▶ *Active/active mode* - In this configuration, all sites process part of the workload. There are no standby sites. Active/active configurations use hardware resources more efficiently than standby configurations since there are no idle systems. Performance can degrade after one site is removed from the scenario, however, since the load on remaining the system increases in that situation.

In both cases, a significant challenge is in maintaining database and storage object synchronicity between sites. Some strategies to overcome this challenge are:

- ▶ Dual load
- ▶ Enterprise storage disk mirroring

#### ***Dual load***

Using this solution, data/content is distributed and loaded twice to independent systems. Load files are transferred from the source to both sites and indexed and loaded into applications/application groups whose properties are identical. The administrator can ensure parallelism by using the OnDemand Administrative Client to export OnDemand objects, such as users, application groups, and storage sets from one system to another:

There are a few issues that need to be taken care of or considered:



- ▶ Make sure the OnDemand objects are synchronized on the two independent systems. If you forget to export an application group, load will fail on that system and data will not be in agreement on the two systems.
- ▶ Make sure that objects are exported in the correct logical order. For example, because the folder contains a logical grouping of at least one application group, if the folder is exported before the application group, there will be no way for the system to associate the folder with the application group because the application group object does not yet exist. The export order that will maintain permissions and object ownership is:
  - a. Printers
  - b. Users
  - c. Groups
  - d. Storage sets
  - e. Application groups
  - f. Folders
- ▶ Make sure to monitor the loads to validate data parallelism. You can use the OnDemand tools, such as ARSDOC of the system log, with query or get parameter, to verify that both loads agree in number and content. Customized scripts can be written to automatically parse, compare, and summarize ARSLOAD output to help you managing this task.
- ▶ This solution does not facilitate the replication of OnDemand annotations. Annotations are stored in their own OnDemand database table rather than with the documents to which they refer. If an annotation is added on one system by users, it will not be automatically replicated on the second system. In addition, the annotation table cannot be exported from a source system and imported to a target system as with a regular database table because the annotation table contains internal references to the document metadata that cannot not be translated to the target system. If you use dual load and need to use annotation, customizing would be required to replicate annotations from one system to the second system. One possible solution is to use the OnDemand Web Enablement Kit (ODWEK) Java™ APIs to write a program to extract the annotation text values and permissions from the source system and use these values to create a new annotation on the target system. This solution preserves the annotation permissions and text but not the time stamp of the original annotation creation, color, and position.

### ***Enterprise storage disk mirroring***

Storage products such as the IBM TotalStorage® Enterprise Storage Server (ESS) typically configured to provide RAID 5 volumes enable synchronous or asynchronous mirroring of data from one storage subsystem to another. The secondary storage subsystem can be located in the same site or at another site some distance away. These types of solutions are application independent. The copy functions occur at the disk subsystem level, and the application has no

knowledge of its existence. These are more expensive solutions because of the additional hardware, software, networking, and skill resources required.

### 4.3.2 Business continuity configuration examples

In this section, we provide some examples of possible business continuity (disaster recovery) configuration for use with an OnDemand solution:

- ▶ Example 1: Multi-site active/standby configuration
- ▶ Example 2: Multi-site active/active configuration
- ▶ Example 3: Multi-site primary/secondary with disk replication configuration

#### **Business continuity example 1: Multi-site active/standby**

This setup uses a second site located at a different data center to take over the operation of the primary system in case of a hardware, software, or data center failure. Systems are completely independent of one another in that they maintain a separate database and separate storage. Figure 4-19 on page 151 shows this configuration.

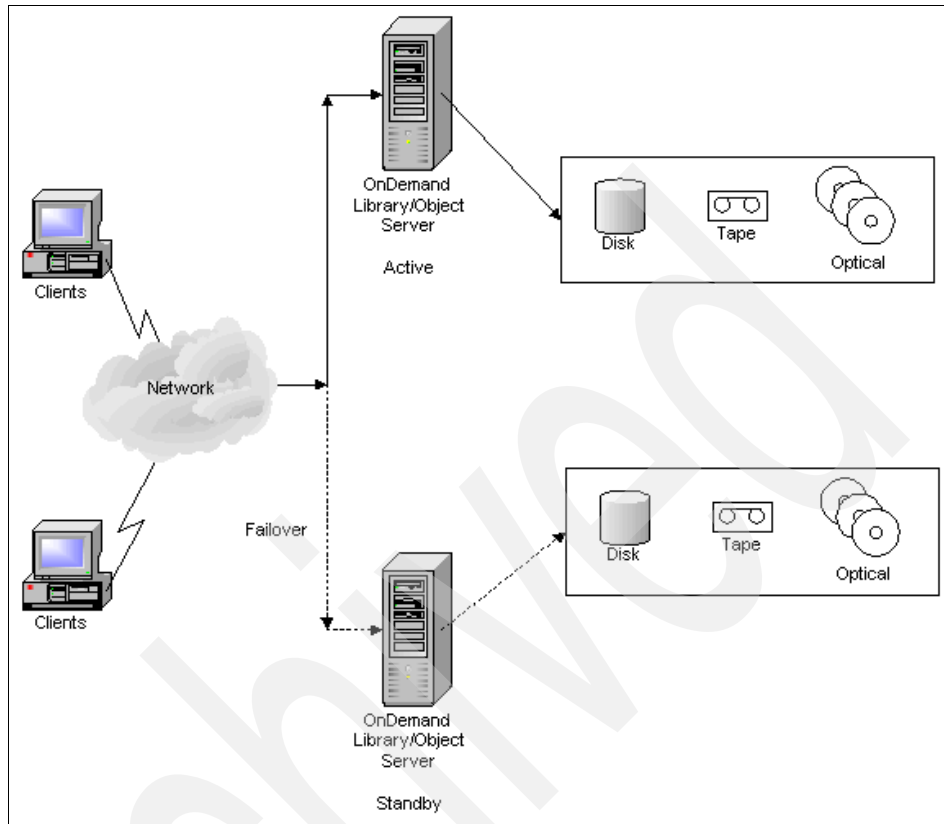


Figure 4-19 Business continuity active/standby configuration

You can setup the system with either an automated failover or a manual failover. An *automated failover* would be transparent to the clients trying to access the system. An automated failover would involve network monitors evaluating the health of a system and redirecting client requests to the backup system in the event that the primary system is unavailable. A *manual failover* would not be transparent to the client and would involve the clients trying to access the system, failing, and then redirecting their request to a second system.

Dual loading is used to simulate mirroring of ingested data content on both systems. Some monitoring of loads would be required in order to be assured that loads on both systems were successful and comparable. Custom scripts could be written to query the OnDemand system log for successful and unsuccessful load reports, and to compare their output.

The following are some dual loading solution options on a multi-site active/standby configuration:

- The primary system receives load data files and executes an across-the-network load on both the primary and standby systems. A custom script would be useful to copy the temporary index files to each download directory monitored by different Library Servers, thus eliminating the need for redundant indexing.

We offer the following sequence as a possible scenario:

- a. The primary system runs three ARSLOAD daemons (or services); one for requesting indexing only, one for specifying the target Library Server locally (for the primary system) and one for specifying the target Library Server remotely (for the standby system).
- b. A load data file - loadfile - is transferred to the download directory monitored by the first ARSLOAD daemon that requests indexing only.
- c. The first ARSLOAD daemon would request to have the files indexed only and the temporary index files are created: loadfile.ind, loadfile.out, and, if AFP data is present, loadfile.res.

**Note:** The first ARSLOAD daemon's job is simply to create the temporary index files. These files would then be distributed (by copy or move) to directories monitored by the other two ARSLOAD daemons.

- d. Upon completion of indexing by the first ARSLOAD daemon, a custom script, that has been checking the directory for index completion, would *first copy* the temporary index files to the download directory being monitored by the ARSLOAD daemon specifying to load to the backup Library Server (standby system), and *second move* the files to the download directory monitored by the ARSLOAD daemon specifying to load to the primary Library Server. Finally, the script creates a loadfile.ARD or loadfile.PDF (using touch) to initialize the load process.
  - e. The two remaining ARSLOAD daemons would recognize that the files are already indexed (by the presence of the ind, out, and res files). They would skip the indexing step and move directly to database loads and storage object loads. The remaining daemons would load the data to their respective servers (one daemon for each server). These occur independently at this point to their respective Library Servers.
- The standby system could receive load data files and execute an across-the-network load on both the primary and standby systems. This procedure would be identical to the procedure listed in the previous option, except that the standby system would be the one receiving the load data files and running the three ARSLOAD daemons. This option would be favorable to the previous option because the processor that executes the indexing (the

processor on the backup system) would otherwise be idle. Also, the primary system is then almost fully utilized to handle client requests.

- The load data files could be routed to both destination Library Servers and indexed and loaded on their respective systems. In this case, each system would have its own ARSLOAD daemon(s) performing indexing and loading to the database and storage manager to the local Library Server.

Because the systems are completely independent, the OnDemand configuration files need to be maintained separately on both systems. The OnDemand objects such as users, groups, applications, application groups, folders, storage sets, and printers can be exported using the OnDemand administrative client to push changes to both systems.

### **Business continuity example 2: Multi-site active/active**

This configuration uses a second system located at a different data center to share the load of OnDemand users. Systems are completely independent of one another in that they maintain a separate database and separate storage. Figure 4-20 on page 154 shows this configuration setup.

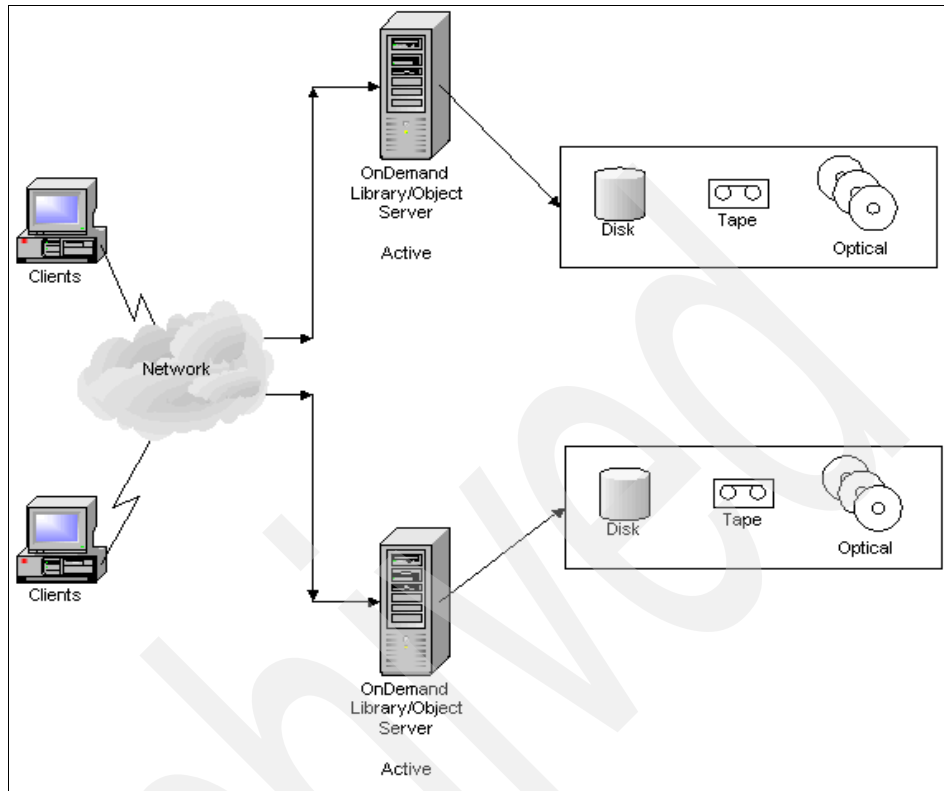


Figure 4-20 Business continuity shared load, active/active configuration

This configuration can also be setup with an automated failover or a manual failover as described in previous example.

Dual loading can be used to simulate mirroring of ingested data content on both systems. Again, as in the previous example, load monitoring would be required in order to assure that loads on both systems are successful and comparable. Custom scripts could be written to query the OnDemand system log for successful and unsuccessful load reports, and to compare their output.

Of the dual loading strategies mentioned on page 152 for the previous example, only number 5 on page 153 would be recommended for an active/active configuration if the user load is also evenly distributed to both servers. That solution option involves sending each server its own copy of the load data files and process on its own. That way, the load will be evenly distributed across both servers.

If the user load is heavier on one system than the other, it might be preferable to concentrate the task of indexing on the less used system.

Administration of the OnDemand system would be very similar to the previous “Business continuity example 1: Multi-site active/standby” on page 150, with the exception that the annotations could potentially be added to both working systems causing the synchronization of the annotations tables to be more complicated.

### **Business continuity example 3: Multi-site primary/secondary with disk replication**

This setup differs from the previous two examples because dual loading is not used to synchronize the data at the two different sites; rather, a dual write is performed.

ESS volumes that use Peer-to Peer Remote Copy (PPRC) can be used to copy data to a remote site for disaster recovery purposes. It is effectively, a hardware mirroring technique. PPRC allows mirroring to be suspended and restarted without affecting data integrity because data is only accessed at one site (the primary site) at a time; discontinuing data mirroring via PPRC will not affect users accessing data on the primary site.

Depend on your business needs, many sites also use this kind of configuration to make “hot” and very fast backups for their systems. For example, you have configured and have made PPRC over your data. Once you have the mirrored data on the second site, you can then suspend the PPRC services over that volumes and execute a “flashcopy” operation on the secondary site volumes to another set of volumes on the same site (secondary). With this, you have a backup of your entire system in a very short time. One thing to remember, with this setup, is that it requires three times the normal storage space. After the “flashcopy” operation, PPRC must be restarted again and the difference between the sites will be reconciled, so data integrity is guaranteed. It is beyond the scope of this redbook to cover all the tasks and configuration steps involved to accomplish this. Different areas and specialists should be involved to carry out this configuration.

Figure 4-21 on page 156 shows this configuration setup.

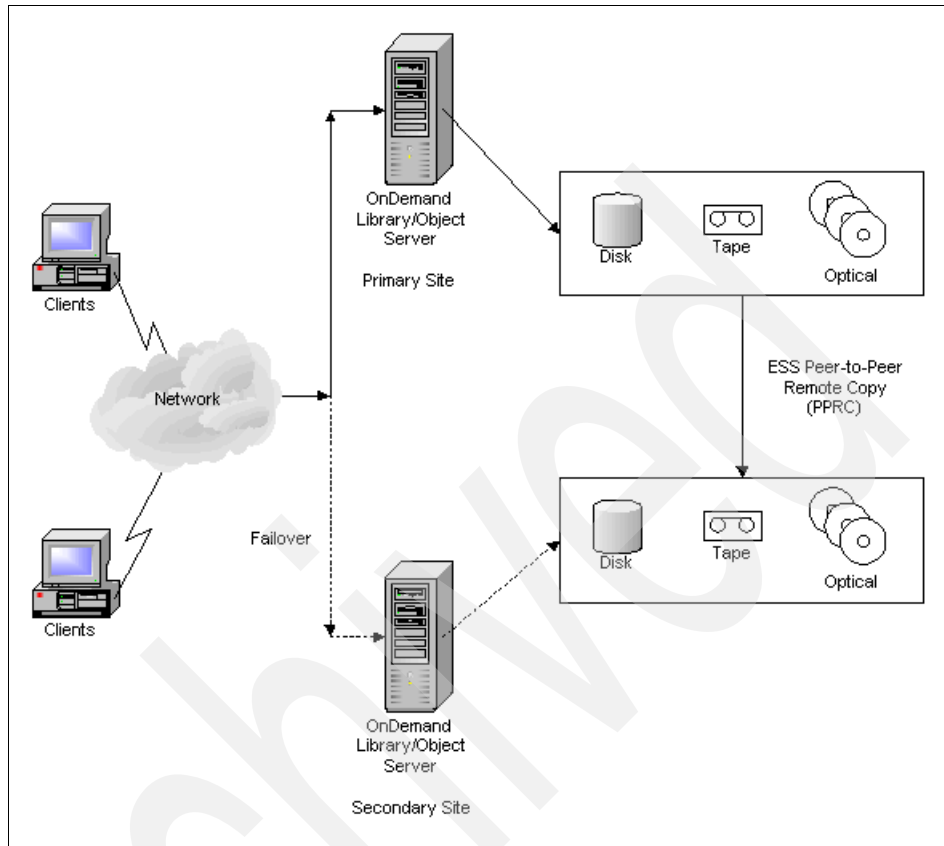


Figure 4-21 Business continuity primary/secondary with disk replication

Should a failure occur on the active system, the second site will take over the operation. A system is configured on the second site to acquire the copied data and continue the normal business processing using the copied data.

There are several ways to set up data failover to a secondary site using PPRC services. Depend on your business needs, you may have different configuration requirements. What we suggest here should be used as reference only.

High availability using HACMP software could be configured on the primary site in case you have more than one node. This is the more common situation because many sites have their operational and main systems on a primary site and on the secondary or standby site, less resources. All the configuration examples presented in this chapter should apply for a valid configuration for a primary site. Then, on the secondary site, you should have only one system configured to acquire the storage and to be able to connect to the network of the former one so that it will be ready to continue working.



Another option is to use HACMP/XD which takes care the different steps involved in migrating the resources from one site to another. For example, it will establish the PPRC freeze, consistency and terminate on the primary site, make the mirrored volumes available on the secondary one, and start the application.

**Note:** If you are interested, refer to *HACMP Remote Copy: ESS PPRC Guide for HACMP for AIX*, SC23-4863, as the first reference.

You can download it from the following Web site:

<http://publibfp.boulder.ibm.com/epubs/pdf/c2348630.pdf>

We recommend performing the following steps before failover to the backup site (again, you may change it according to your system setup and business requirements):

1. Establish path between the primary site A and the secondary (backup) site B (PPRC\_PATH\_AtoB).
2. Establish PPRC synchronization between the primary site A and the secondary (backup) site B (PPRC\_SYNC\_AtoB).
3. Setup HACMP cluster at the primary host.
4. Assign the backup site ESS volume to backup the host which is part PPRC target volume.

Refer to Example B-1 on page 336 for the sample script that needs to be executed before failover at backup site. It includes the tasks to establish the paths and their relationships between the set of disks (LUNs) from the primary site A to the secondary site B. Data will be copied and synchronized. Note, on the secondary site, the data will not be available.

For the failover, the PPRC copy services of the devices might be freezed to stop I/O operations and keep synchronization between pairs. After that, consistency task must take place. Once this is done, terminate the PPRC services to have the volumes on the secondary (backup) site B available to use.

We recommend performing the following steps during failover to backup site provided that both sites are in PPRC synchronization:

1. Establish PPRC freeze at the primary site A (PPRC\_FRZ\_AtoB).  
Make sure the primary host running no or minimum I/O.
2. Establish PPRC consistency created at the primary site A (PPRC\_CONST\_AtoB).

If not, the primary host I/O will fail due to write inhibit by ESS due to freeze operation.

3. Establish PPRC terminate between A and B (PPRC\_TER\_AtoB).
4. Import backup ESS volumes at backup host.
5. Mount file systems at backup host.
6. Setup HACMP cluster at backup host.
7. Valid data integrity.

Example B-2 on page 338 shows the sample script that need to be executed after failover at backup site.

### 4.3.3 eRCMF (enterprise Remote Copy Management Facility)

eRCMF is intended as a multi-site disaster recovery solution for open systems. It is capable of automatically repair inconsistent PPRC pairs. It is a scalable, flexible open systems ESS solution that protects the business data and can be used for both:

- ▶ The planned outages (hardware and software upgrades).
- ▶ The unplanned outages (disaster recovery, testing a disaster).

It simplifies the disaster recovery implementation and concept. Once eRCMF is configured, it monitors the PPRC states of all specified LUNs/volumes.

eRCMF provides following functions:

- ▶ When a site failure occurs or maybe occurring, eRCMF splits the two sites to allow the backup site to be used to restart the applications. This needs to be fast enough so that, when the split occurs, operations on the production site are not impacted.
- ▶ Manage the states of the PPRC and flashcopy relationships so that the customer knows when the data is consistent and can control on which site the applications are to be run.
- ▶ Offer easy commands to place the data in the state it needs to be in. For example if the data is out of synchronization, the command resync causes eRCMF to scan the specified volumes and issue necessary commands to bring the volumes back into synchronization.
- ▶ Offer tool to execute eRCMF configuration checks to verify if the eRCMF configuration matches the physical ESS setup. It is required to discover customer ESS configuration changes which affect the eRCMF configuration as well. Regular checks keeps the eRCMF configuration up-to-date with its actual environment. Otherwise, full eRCMF management functionality is not given.

eRCMF uses two levels of management:

- ▶ Enterprise level: Actions affect the complete monitored environment.
- ▶ VolumeSet level: Actions affects only monitored volumes of a VolumeSet. Managing site splits will be at the enterprise level because the freeze function works against LSS pairs and several VolumeSets may share the same LSS pair. However, managing the states (consistency) will be at VolumeSet level. The reason behind this is two fold: Open systems users want to manage by server or server group and HACMP wants to manage by volume group.

eRCMF V2 is designed as a three-site disaster recovery solution and runs on dedicated Productivity Center Machines (PCMs), which run AIX with WebSphere® and incorporate the Copy Services Server (CSS).

As a result, the following simplifications have been made:

- ▶ Only one CSS pair (PCM pair) per installation, preferably on different physical sites. The size of the PCM can be scaled as needed to handle more ESSs.
- ▶ Reduce the number of predefined tasks required for disaster recovery with PPRC. eRCMF is capable of directly invoking CSS commands without having to predefine CSS tasks.

#### 4.3.4 Business continuity summary chart

Table 4-9 shows a summary of the high availability strategies and options discussed in this chapter. Refer to Figure 1-3 on page 19 for a description of the levels of availability.

Table 4-9 Business continuity summary chart

| Example   | Description                                     | Level of business continuity |
|-----------|-------------------------------------------------|------------------------------|
| Example 1 | Multi-site active/standby configuration         | Tier 5                       |
| Example 2 | Multi-site active/active configuration          | Tier 5                       |
| Example 3 | Multi-site active/standby with disk replication | Tier 6                       |



## Case studies for OnDemand Multiplatforms

In this chapter, we apply some of the technologies, strategies, and options, discussed earlier in this redbook, to two case studies. For each case study, we describe its background information, the backup procedures, the high availability configuration, and disaster recovery plan implemented.

The two case studies covered in this chapter include:

- ▶ Global voice and data communications company
- ▶ International financial services company

## 5.1 Global voice and data communications company

Our first case study is a study of a global voice and data communications company.

### 5.1.1 Background

A global communications company has a requirement to make electronic statements available to their customers online. They also want to provide their customer service representatives with the most recent statements received by their customers, an archive repository of statements, and reports from years past.

They implemented OnDemand for Multiplatforms for their company need.

Since the implementation, the company's database has grown to approximately 550 GB. At any given time, they have approximately 4 TB of data objects stored in the OnDemand managed cache file systems. Most of the customer statements are stored in the Advanced Function Presentation (AFP) data format and most of the internal reports are stored as line data. They also store image and Portable Document Format (PDF) data but to a much lesser extent.

Approximately 12 TB of data files are stored on TSM managed media, and an additional 12 TB reside in TSM copy storage pools. TSM provides an immediate "backup" of data loads for OnDemand since they load data to TSM at the same time they load data to the database and OnDemand managed cache. Their average load volumes are approximately 240 GB per month.

This company's peak volume support close to 3,000 concurrent users extrapolating to 50,000 to 70,000 logged-on users at a glance. 60% of their users access the OnDemand system using a Web application. The other 40% are internal Windows client users. They process just under 10 million retrievals per month with peaks at 500,000 per day.

### 5.1.2 Backup, recovery, and high availability approach

The global voice and data communications company has the following backup procedure, high availability configuration, and disaster recovery plan.

#### **Backup procedure**

A full offline OnDemand database backup is performed every weekend on the Library Server and is stored in TSM. Because the full offline backup takes a snapshot of the database at a point in time, archive logs are not required to

restore the database. The TSM database is also backed up in its entirety while the OnDemand application is offline.

Because application groups are configured to store data to cache and a TSM defined nodes at load time, the cache file systems defined in the ars.cache file are not backed up.

## High availability configuration

This telecommunication company is configured to run OnDemand in a distributed Library Server and Object Server system configuration with TSM using a standby node for high availability. (Refer to 2.2.6, “Distributed OnDemand system with TSM” on page 34 for more information).

Table 5-1 lists the software installed on the Library Server, Object Server, and the standby node.

*Table 5-1 Software installed on the active nodes and the standby node.*

| System role             | Software                                                                                               |
|-------------------------|--------------------------------------------------------------------------------------------------------|
| OnDemand Library Server | AIX 5.2<br>HACMP 4.5<br>DB2 EEE 7.2<br>Content Manager OnDemand 7.1.1                                  |
| OnDemand Object Server  | AIX 5.2<br>HACMP 4.5<br>TSM Server and Client API 5.2<br>Content Manager OnDemand 7.1.1                |
| Standby node            | AIX 5.2<br>HACMP 4.5<br>DB2 EEE 7.2<br>TSM Server and Client API 5.2<br>Content Manager OnDemand 7.1.1 |

The Library Server, Object Server, and the standby node all have a physical attachment to the IBM ESS subsystem (shared disk) via RS232 cable. The standby node is inactive. It waits for either the Library Server or the Object Server to fail and leave the cluster, and then the standby node takes over the role of the failed node.

This company's high availability plan is visualized in Figure 5-1 on page 164. When all nodes on the system are healthy, the Library Server owns the resource group A, which consists of the OnDemand database and database log files. The primary Object Server owns the resource group B, which consists of the OnDemand managed cache file systems, TSM database, and TSM logs. Should the Library Server node fail and leave the cluster, the standby node assumes

control of the resource group A and functions as the Library Server. Should the Object Server node fail and leave the cluster, the standby node assumes control of the resource group B and functions as the Object Server.

This configuration is a variation of the “High availability example 1: Standby configuration” on page 92. The added benefit here is that the redundant hardware is minimized by having the standby node act as failover for both the Library Server and Object Server. The standby node has three network interfaces and separate physical connections to each server node’s external disk. Therefore, the standby node can, if necessary, take over for both servers concurrently. The cluster’s performance, however, would most likely degrade while the standby node functions in both roles.

Figure 5-1 shows the system configuration for this case study.

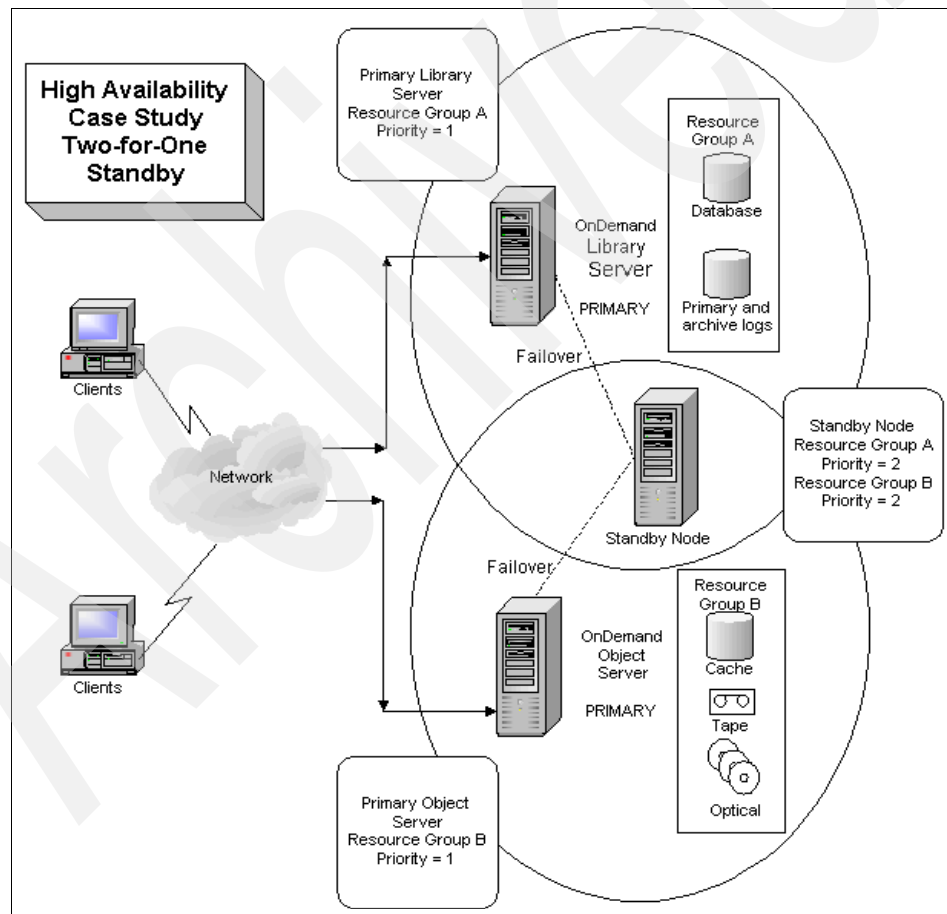


Figure 5-1 High availability configuration two-for-one standby



### **Disaster recovery plan**

The disaster recovery plan is to store backups at an offsite location. There is no mirrored site, but rather an expectation that if a catastrophic event was to occur at their data center that an identical hardware configuration could be created, and backup and storage volumes restored.

## **5.2 International financial services company**

Our second case study is a study of an international financial services company.

### **5.2.1 Background**

A major financial services company has implemented OnDemand to manage their electronic data. More than 50% of their users access the OnDemand system via a company Web site. The others are internal users who access the system using the OnDemand Windows client.

The database is approximately 275 GB. Approximately 1 million stored data objects reside in cache which equates to approximately 1.3 TB of data spread across nine OnDemand managed cache file systems.

Approximately 3 TB of data are stored on TSM managed media, primarily to optical platters but some data is also stored on LTO tape drives (for example, backups). The company has configured TSM copy storage pools in order to have backup platters of their data in TSM. Approximately 2,000 loads are performed each day. TSM stores data to seven 3995 optical jukebox libraries.

### **5.2.2 Backup, recovery, and high availability approach**

The international financial services company has the following backup procedure, high availability configuration, and disaster recovery plan.

#### **Backup procedure**

A full online OnDemand database backup is performed twice each week on the Library Server during the night when the system is least accessed. Incremental backups are performed all other nights. Database backups and logs are managed by TSM and stored to LTO tape drives. The TSM database is backed up at the same time as the OnDemand database.

Because application groups are configured to store data to cache and a TSM defined node, the cache file systems defined in the ars.cache file are not backed up.

## High availability configuration and disaster recovery plan

Their implementation involves two independent Library Server and Object Server configurations each with TSM. (See 2.2.6, “Distributed OnDemand system with TSM” on page 34 for more information). The independent systems are located at separate geographical sites providing a standby system for disaster recovery. Both systems are active in production and dual loading is conducted to keep the systems equivalent in content.

This company implemented a distributed Library Server and Object Server system with TSM (See 2.2.6, “Distributed OnDemand system with TSM” on page 34 for more information) using two active sites to address both their high availability and disaster recovery requirement. This solution involves two independent Library Server and Object Server, each with TSM. The independent systems are located at separate geographical sites providing a standby system for disaster recovery. Both systems are active in production. They use dual loading (defining multiple route destinations from the mainframe that generates their report files) to keep the systems equivalent in content. This customer does not use annotations in their implementation; so consequently, there is no concern with synchronization of the annotation table. Loads are verified on each system by an OnDemand administrator to ensure congruency in content.

Because this company has two independent active systems, each system effectively act as a hot standby for each other and the system downtime is far less likely. The switch over to the healthy system, however, is not automatic and is not monitored by any high availability software. If a failure occurs at one site, the clients simply log on to the healthy site and work from the healthy site.

Figure 5-2 on page 167 shows the system configuration for this case study.

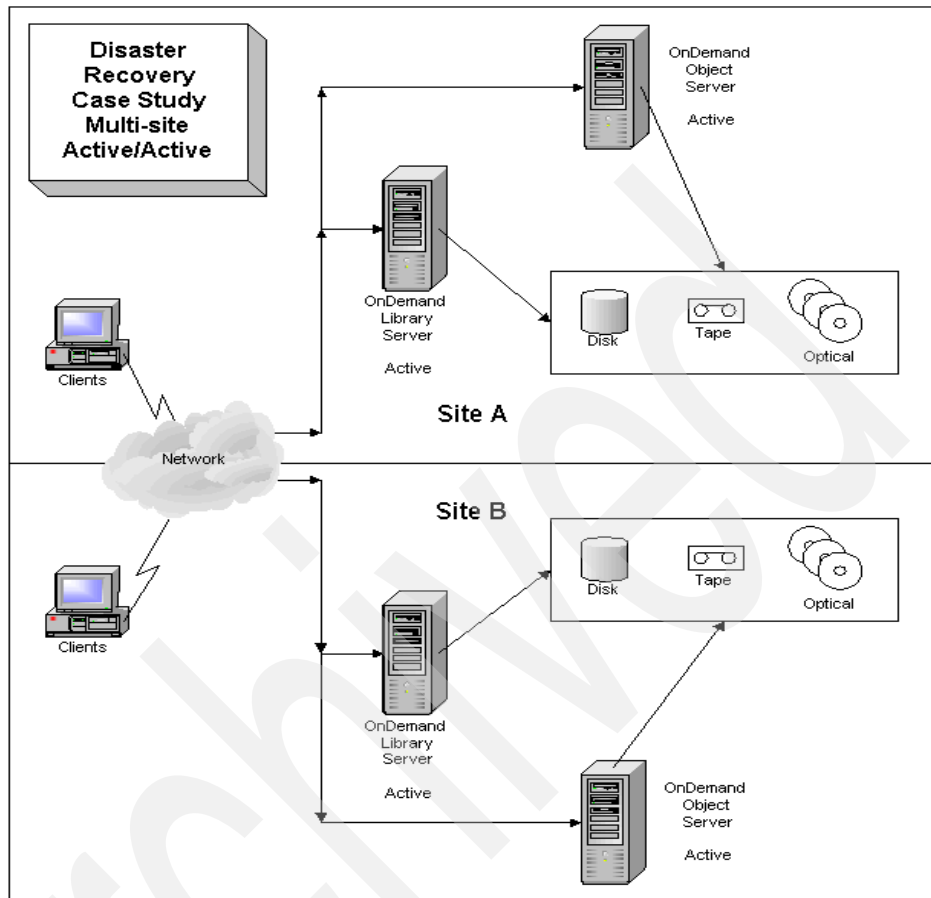


Figure 5-2 Disaster recovery configuration multi-site active/active





## Part 3

# iSeries

In Part 3, we focus on OnDemand for iSeries. We provide an introduction to the overall iSeries architecture and an overview of OnDemand for iSeries. In addition, we discuss backup and recovery strategies, and introduce some high availability options and strategies to assist you in selecting an appropriate solution to satisfy your business requirements and to suit your operating environment.

IBM i5/OS is the next generation of OS/400. Sections in this redbook may refer to i5/OS as OS/400.



## iSeries architecture

The inherent reliability, ease-of-operation, and robustness of the iSeries makes it the ideal server for your Content Manager OnDemand system. In this chapter we provide a brief overview of some of these strengths and design features.

The following topics are covered:

- ▶ iSeries success factors
- ▶ iSeries architecture overview

## 6.1 iSeries overview

One of the key factors that differentiates the iSeries from other systems is the level of hardware and software integration. Unlike most other systems, the iSeries does not require you to select software components from different vendors (such as operating system, relational database, security, system management software) and integrate them in order to build a robust working environment for your business applications.

The iSeries is an integrated system that offers an out-of-the-box, ready-to-run approach. OS/400 includes a complete range of licensed programs (middleware) that offer the highest level of integration. By reducing the extent of integration required during implementation, the iSeries approach minimizes implementation costs and increases reliability. The iSeries also provides customers with the highest level of ease-of-use in today's market. The initial ease of implementation and the on-going ease of use, combined with its reliable integration, make the iSeries a high-performing, highly available and low-cost business solution.

### 6.1.1 iSeries success factors

The iSeries server has a long and successful history worldwide. The reason for this success is founded in six basic factors:

- ▶ Architecture
- ▶ High level of integration
- ▶ Interoperability
- ▶ Client/server capability
- ▶ Scalability
- ▶ Price and performance

#### ***Architecture***

One of the key factors contributing to the commercial success of the iSeries server is its integrated architecture. Several architectural features distinguish the the iSeries from other servers. These features include:

- ▶ Layered architecture
- ▶ Technology-independent machine interface (TIMI)
- ▶ Object-based system
- ▶ Single-level storage
- ▶ Separate I/O processors
- ▶ Multiple data busses
- ▶ High degree of integration
- ▶ Open standards
- ▶ Logical partitions and multiple operating systems



For more information, refer to 6.1.2, “iSeries architecture overview” on page 176.

### ***High level of integration***

The iSeries offers the highest integration of both its hardware and software components. Hardware, microcode, the operating system, and IBM middleware are tightly interlaced, allowing maximum exploitation of all available computing resources. Integration of input/output processors (IOPs) and direct access storage devices (DASD) yields valuable benefits, such as an extremely high level of reliability and availability. Some of the features that are integrated into the iSeries include:

- ▶ System availability:
  - Battery backup unit (BBU)
  - Continuously powered main storage (CPM)
  - Uninterruptable power supply (UPS)
  - Protection against system failures
  - Backup and recovery
  - Mirroring and RAID-5 (both at minimal performance cost)
  - Menu-driven backup and recovery
  - Journaling
  - Commitment control
  - Auxiliary storage pools (ASPs)
  - Save-while-active functionality
  - Clustering support
  - Logical partitioning
- ▶ Standard ease-of-use functions for:
  - System customization
  - Automatic procedures, startup programs, and so on
  - System values
  - System tuning (managing memory, disk)
  - Graphical user interface (GUI) to system functions through iSeries Navigator.

For more information about iSeries Navigator, refer to:

<http://www.ibm.com/servers/eserver/iseries/navigator>

- ▶ Database management system integrated with the operating system:
  - No additional cost for database software
  - Integrated database administration tools and automated self-managing database administration functions
  - Excellent performance through microcode-embedding, fine tuned with hardware and OS/400
  - Support for parallel database operations, symmetric multiprocessing (SMP), and parallel I/O processing

- DB2 Universal Database (DB2 UDB) for iSeries supports the storing, managing, and indexing of all forms of information, including binary objects such as spreadsheets, word processing documents, and multimedia objects
- ▶ Easy system management through:
  - Easy hardware configuration and reconfiguration
  - Automatic configuration of devices
  - Integrated file system (IFS) accessible through a standard iSeries interface (including the PC and UNIX file system). For more information about IFS, refer to *OS/400 Integrated File System Introduction Guide*, SC41-5711.
  - Minimal database installation, management, and operations activity
  - Simple menu-driven functions
  - Balancing data across disk units
  - Native performance optimization of DB2 UDB for iSeries
- ▶ A single, integrated security model which is shared by the operating system, database, systems management, Internet, mail and communications functions. The iSeries and its predecessors have always been a platform where the operating system and features are developed with a focus on security. In the past, the iSeries have received several security certifications, including a C2 certification from the US Department of Defence. The IBM 4758 PCI Cryptographic Coprocessor that is available on iSeries was the first product in the market to have received certification for the National Institute of Standards and Technology (NIST) Federal Information Processing Standards (FIPS) 140-1 Level 4. The 4758-023 Cryptographic Coprocessor is certified for FIPS 140-1 Level 3.
- ▶ Currently, OS/400 is being evaluated by the Common Criteria Evaluation and Validation Scheme (CCEVS) for EAL 4 Augmented ALC\_FLR.2 and Controlled Access Protection Profile certification.

**Note:** CCEVS is a set of criteria set by NIST and NSA to:

- Meet the needs of government and industry for cost-effective evaluation of IT products.
  - Encourage the formation of commercial security testing laboratories and the development of a private sector security testing industry.
  - Ensure that security evaluations of IT products are performed to consistent standards.
  - Improve the availability of evaluated IT products.
- ▶ Internet serving using IBM HTTP Server for iSeries (powered by Apache) and IBM WebSphere Application Server for IBM @server iSeries.

For more information about WebSphere Application Server for iSeries, refer to:

<http://www.ibm.com/servers/eserver/series/software/websphere/wsappserver>

For more information about IBM HTTP Server for iSeries (powered by Apache), refer to:

<http://www.ibm.com/servers/eserver/series/software/http/docs/doc.htm>

- ▶ **Printer management.** iSeries print jobs can be sent or mailed to other iSeries servers and users. Plain text print output can be viewed on the iSeries server before being printed. When using Advanced Function Printing™ (AFP), a user can view the actual printed output (text, forms, or graphics) before it is printed or distributed. Through Infoprint® server, the iSeries provides full support for PDF documents, intelligent routing of documents, print consolidation, segmentation and indexing, Web APF and image transforms.

For more information about IBM Infoprint Server for iSeries, refer to:

[http://www.printers.ibm.com/internet/wwsites.nsf/vwwebpublished/ipserverhome\\_i\\_ww](http://www.printers.ibm.com/internet/wwsites.nsf/vwwebpublished/ipserverhome_i_ww)

- ▶ **Electronic customer support** enabling electronic fix retrieval, problem reporting, remote support and proactive maintenance.

### ***Interoperability***

The iSeries server offers a wide range of communication capabilities and functions that enable the iSeries server to communicate with IBM and non-IBM systems.

### ***Client/server capability***

The iSeries server can operate with almost any client in any communication environment.

### ***Scalability***

The iSeries server product line covers a wide range of performance capacities. With flexible Capacity on Demand options, additional resources can be activated as and when required, without having to reconfigure or restart the server.

### ***Price and performance***

Many independent analysts have confirmed that the iSeries server represents a cost-effective platform in the long run. Its extensive integration yields significant cost advantages, high availability, easy system management, and significant investment protection. This has been the basis for its success in the dynamic world of information technology. The iSeries delivers high computing performance at a low cost of ownership and, therefore, scores high in customer satisfaction.

## 6.1.2 iSeries architecture overview

The iSeries integrated architecture distinguishes itself from other servers with the following features:

- ▶ Layered architecture
- ▶ Technology-independent machine interface (TIMI)
- ▶ Object-based system
- ▶ Single-level storage
- ▶ Separate I/O processors
- ▶ Multiple data busses
- ▶ High degree of integration
- ▶ Open standards
- ▶ Logical partitions and multiple operating systems

### Layered architecture

The iSeries has a layered architecture that is divided into the actual user interface (OS/400) and a *Technology Independent Machine Interface (TIMI)*.

There are two components to the operating system software on an iSeries server. This important distinction is unique in the industry in its completeness of implementation. The two components are *System Licensed Internal Code (SLIC)* and *Operating System/400® (OS/400)*.

*SLIC* is a robust, high-performance layer of software at the lowest level which provides the Technology-Independent Machine Interface (TIMI), process control, resource management, integrated SQL database, security, communications, file system, temporary storage, Java Virtual Machine (JVM™) and other primitives.

*OS/400* provides the functions to work with the above services and to present these services to the users and applications. OS/400 also provides a vast range of high-level language (such as C/C++, COBOL, RPG, FORTRAN) runtime functions and interacts with the client-server graphical user interface and iSeries Navigator.

### ***Technology-Independent Machine Interface (TIMI)***

In the iSeries, changes to processor hardware and firmware do not affect the operating system, middleware, or business applications. The heart of this ability to change without disrupting customers and their applications is the Technology-Independent Machine Interface (TIMI). Figure 6-1 on page 177 shows a simplified view of how TIMI fits into the overall iSeries design.

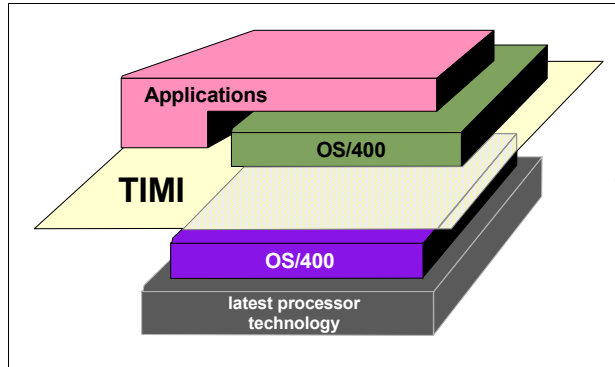


Figure 6-1 Technology-Independent Machine Interface (TIMI)

Applications do not have to be rewritten or recompiled to exploit new hardware technologies. This permits iSeries servers to exploit advances in hardware technology, such as storage, memory, and processor technology, without modifying the existing applications.

All iSeries servers share the same software architecture. This permits applications to scale across the entire product line, from the smallest to the largest models.

### Object-based system

OS/400 treats all information as objects. This is significantly different from the simple byte-string, file-based manipulations used by many other systems. Object-based design enables a powerfully, yet manageable level of system integrity, reliability, and authorization constraints.

All program and operating system information, such as user profiles, database files, programs, and printer queues, have their associated object types stored with the information. In the iSeries architecture, the object type determines how the object can be used and by which methods. For example, it is not possible to corrupt a program object by modifying its code sequence data, as though it was a file. Because the system knows the object is a program, it will only allow valid program operations. Figure 6-2 on page 178 shows an example of a few object types and their associated valid access methods.

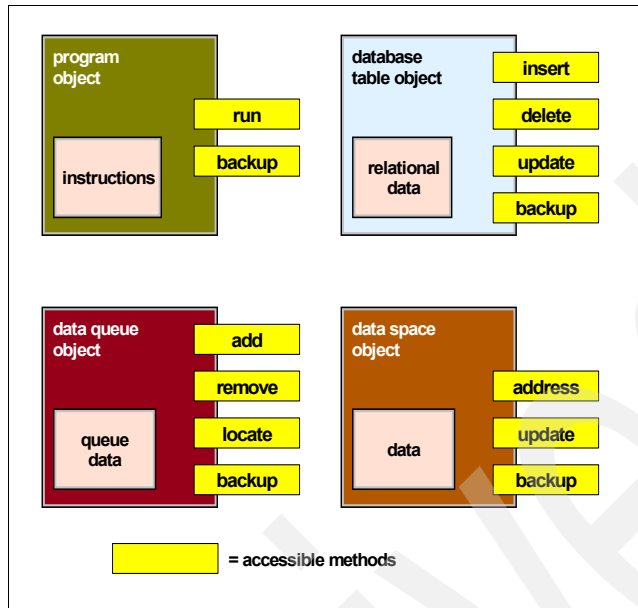


Figure 6-2 Object types and associated operations

As you can see from Figure 6-2, the only valid access methods for a program object is *Run* and *Backup*. There is no associated write method to a program object; this is the *key* in making iSeries programs immune to viruses. Additionally, OS/400 ensures that all secondary components of a particular object remain associated with that object.

Simple stream data files, such as image files and audio files, are stored as stream-file objects with open, read, and write operations.

## Single-Level Storage

The iSeries architecture specifies a single, very large virtual address space known as Single-Level Storage (SLS). All objects, such as programs, files, users, data, working space, and database tables reside in this storage space. This storage space is addressed using 128-bit pointers. Figure 6-3 on page 179 depicts this concept.

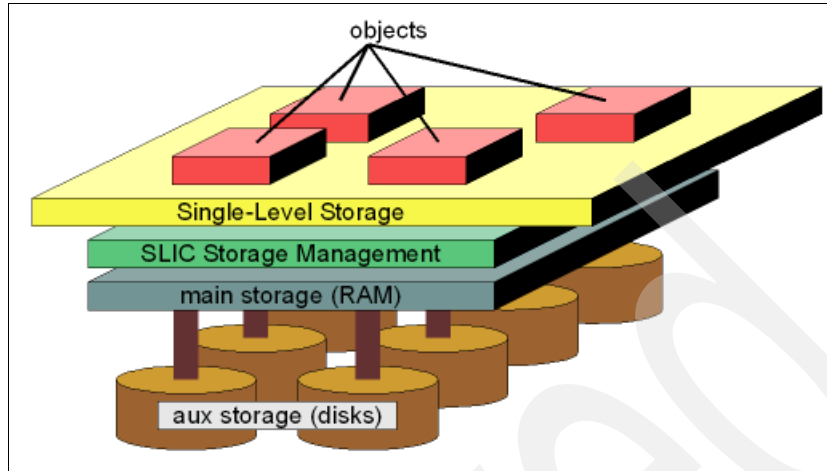


Figure 6-3 Single-level storage architecture

A single page table maps all virtual addresses to physical addresses, which makes task switching very efficient. Furthermore, most SLS addresses contain a real address, thereby eliminating the need for address translation. This in turn speeds up data access. Additionally, OS/400 automatically spans data objects across all available disks arms. This improves the speed of paging and persistent object retrieval and eliminates the need for a database administrator or systems administrator to manage data spaces.

### Separate IO processors and multiple data busses

The iSeries architecture emphasizes the use of specialized, intelligent IO processors.

System tasks, such as disk, networking, tape and terminal IO, are off-loaded from the main application processors and handed to a variety of dedicated processors. This allows the main processors to concentrate on application execution.

### Open standards

The iSeries heritage is based on integration. Emphasis has always been on providing an integrated solution, but with a strong focus on “open” system applications architecture. OS/400 provides numerous industry standards, which facilitate application portability from other platforms and interoperability with other hardware platforms.

## Logical partitions and multiple operating systems

Dynamic *logical partitioning* (LPAR) is a system architecture approach that provides the capability of virtualizing hardware resources that can be shared by multiple independent operating environments. Originally developed for mainframe computers, LPAR allows the division of a single server into several completely independent virtual servers or logical partitions. Figure 6-4 depicts this concept.

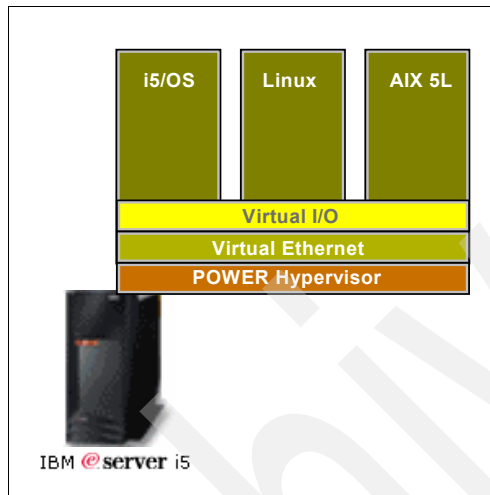


Figure 6-4 Dynamic LPAR

LPAR on iSeries is supported by i5/OS, AIX 5L and by the Linux operating system. Since its introduction in 1999, LPAR on the iSeries and now @server i5 has re-energized server consolidation and IT cost optimization strategies. You can employ LPAR to achieve quick consolidation of multiple workload environments, time zones, and foot prints within a single or fewer iSeries or @server i5 systems.

Introduced first on the @server i5, the IBM Virtualization Engine™ system extends innovations such as micro-partitioning to AIX 5L, automatic CPU balancing with uncapped processor partitioning, and further exploits virtual IO capabilities. Additionally, the previous limit of up to 32 partitions has been significantly increased to a maximum of 254 partitions.

### 6.1.3 Summary

The iSeries architecture is different from most other machines in the computing industry. It is a flexible architecture that is entirely focused on business computing.



The iSeries family is equipped with a range of dynamic workload management features designed to enable businesses to adjust workloads and performance dynamically and automatically in order to meet constantly shifting business priorities and increase server utilization rates. The workload management tools incorporated into iSeries servers can give you the option to run multiple subsystems, allowing administrators to enhance productivity by balancing the processing priorities for different applications running within the same operating system image.

With the latest generation of IBM @server iSeries servers, you have the flexibility to run on demand computing environments for i5/OS (the latest generation of OS/400), IBM AIX 5L, Microsoft® Windows, Linux, WebSphere, Lotus Domino® and Java solutions on a single highly integrated, powerful server. Extensive growth options, resource virtualization and intuitive management tools mean that iSeries servers can provide not only the power and capacity to run core business applications, but also the freedom and scalability to add new e-business applications on the same server.



## OnDemand for iSeries overview

Chapter 2, “Content Manager OnDemand overview” on page 27 provides a general overview of IBM DB2 Content Manager OnDemand. Although the fundamental concepts are identical on all hardware platforms, there are, however, certain aspects of OnDemand for iSeries that are different.

In this chapter, we provide an overview of OnDemand on iSeries. We look at some of the features and benefits of deploying Content Manager OnDemand on the iSeries and also discuss a few implementation considerations.

The following OnDemand for iSeries topics are covered:

- ▶ Installation and configuration
- ▶ Administration
- ▶ User interface

**Note:** This chapter focuses on the Common Server implementation of Content Manager OnDemand for iSeries, as opposed to the Spool File Archive (SFA) feature.

## 7.1 Introduction

Managing the information needed to comply with the growing number of regulatory and statutory requirements is a real challenge for any modern organization. IBM content management software delivers the most comprehensive and powerful platform today for managing and delivering all forms of information on demand. It can help you to improve productivity, enhance responsiveness, and streamline regulatory compliance.

IBM DB2 Content Manager OnDemand (OnDemand) for iSeries is part of the IBM DB2 content management software for enterprise content management product suite. Implemented on the iSeries as a licensed program, it could transform your iSeries into a very powerful archive server to support the native OS/400 applications, as well as business applications running on other host systems.

With the release of OS/400 V5R1, OnDemand for Multiplatforms was ported to the iSeries and became what we today commonly refer to as OnDemand Common Server. Certain architectural enhancements have been made to take full advantage of the inherent strengths and design features of the iSeries, many of which are discussed in Chapter 6, “iSeries architecture” on page 171.

With Common Server, many new features and enhanced functionality have been added to OnDemand. These include:

- ▶ Client/server archival and retrieval of OS/400 spooled files and data, with integration capabilities for scanned images and other non-iSeries files, including Binary Large Objects (BLOBS)
- ▶ Enhanced indexing capabilities, including:
  - A maximum of 32 indexes
  - Maximum field length of 254 bytes
  - Additional field data types
  - Full text search capability
  - Flexible field mapping options
  - Portable Document Format (PDF) indexing support
- ▶ Policy-driven life cycle management with automation
- ▶ Improved disaster recovery options through automatic media duplication, including dual-write capabilities for optical and tape media
- ▶ Powerful and flexible full graphical Windows administration client, in conjunction with the iSeries Navigator interface
- ▶ Ad-hoc CD mastering functionality to provide offline access to archived data and indexes
- ▶ Ability to archive PC files in OnDemand through the optional Store OnDemand service offering or through the use of e-document support in Kofax Ascent Capture

- ▶ Highly customizable graphical end-user interface through the OnDemand Windows client
- ▶ Access to the archive repository through a Web browser interface with the OnDemand Web Enablement Kit (ODWEK)
- ▶ Significant security enhancements, including the option not to associate OnDemand users with corresponding OS/400 user profiles
- ▶ Application Programming Interface (API) to launch the OnDemand end-user client from 5250 applications
- ▶ 5250 command interfaces for selected operational functions, for example output queue monitoring and report archiving
- ▶ Support for multiple OnDemand environments on the same iSeries server

Common Server shares a common code base with OnDemand for Multiplatforms which enables you to implement common and consistent OnDemand applications on different hardware platforms. In V5R3, OnDemand for iSeries provides Common Server as well as full Spool File Archive and AnyStore support. In addition, V5R3 has a built-in migration utility to help you move your Spool File Archive indexes and report definitions to OnDemand Common Server so that you can take advantage of the new features and functions.

## 7.2 Installation and configuration

OnDemand for iSeries is installed as an OS/400 licensed program. Product 5722-RD1 with \*BASE feature is a requirement. Depending on the functionality that you require, you should install additional features. For example, if you require Spool File Archive and Common Server functionality, then you would need to install Option 1 and Option 10 in addition to the \*BASE feature.

For example, if you require Spool File Archive or Common Server functionality, then you would need to install the \*BASE feature and Option 1 or Option 5.

Figure 7-1 on page 186 shows some of the licensed programs and features that you may want to install depending on your business requirement.

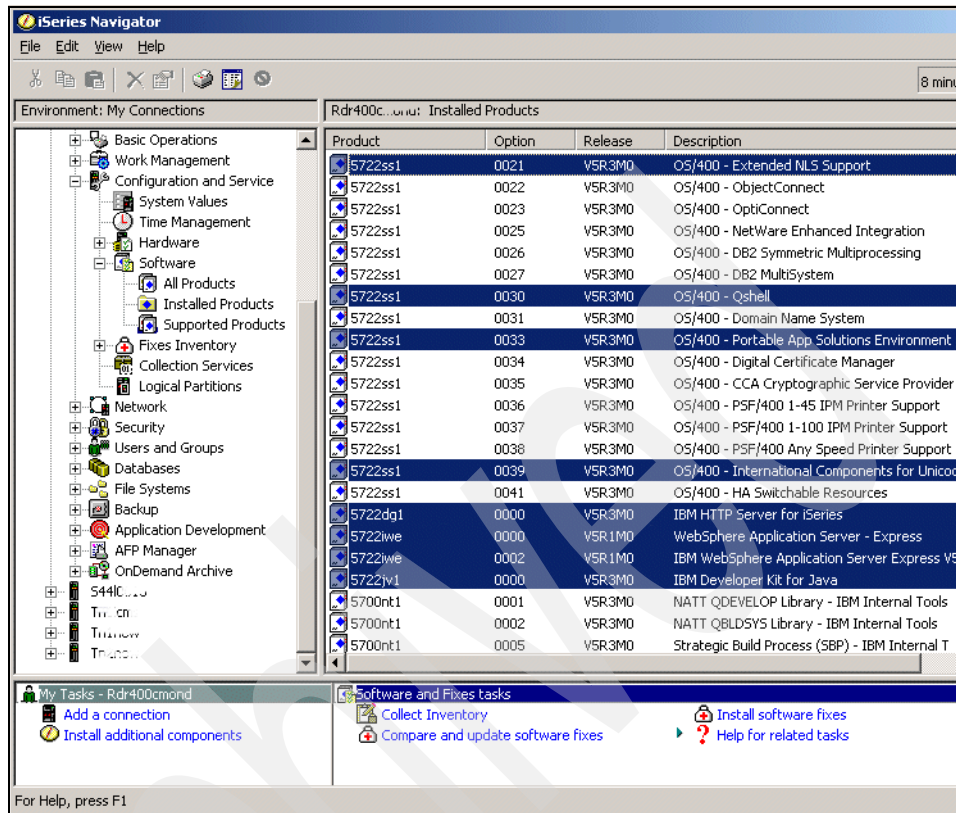


Figure 7-1 Some of the OnDemand product options

During the installation and configuration of OnDemand, certain OS/400 objects are automatically created, including some of the following OS/400 object types:

- ▶ Libraries
- ▶ Integrated File System (IFS) directories
- ▶ Job descriptions
- ▶ Tables (Files)
- ▶ Application programs
- ▶ Journals and Journal Receivers
- ▶ User spaces
- ▶ Data queues
- ▶ Data areas
- ▶ Output queues
- ▶ Authorization lists
- ▶ User profiles

OnDemand's user-defined configurations and business data are stored in database tables and in IFS directories. They include:

- ▶ Application group definitions
- ▶ Application definitions
- ▶ Folder definitions
- ▶ OnDemand user information
- ▶ Report index information
- ▶ Cached archived objects
- ▶ Migration policies

For more information and documentation on installing OnDemand for iSeries, refer to the following URL:

<http://www.ibm.com/software/data/ondemand/400/library.html>

## 7.3 Administration

OnDemand for iSeries offers a full graphical administration client for managing users, application groups, applications and folders in multiple environments. Figure 7-2 shows an example of the OnDemand administration client.

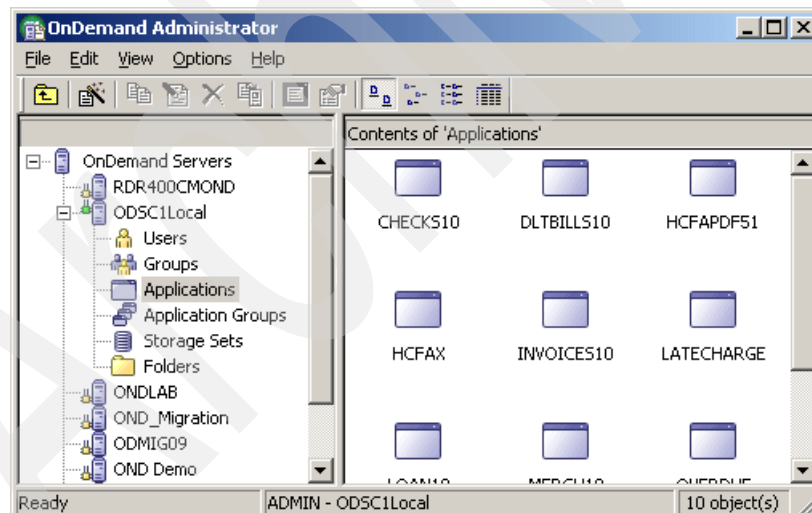


Figure 7-2 OnDemand administration client

The administration client further offers a graphical utility known as the report wizard. The report wizard enables you to access spool file data directly from an OS/400 output queue and graphically define your application group, application, and folder in a series of easy steps.

**Note:** You may also use the graphical interface to define your report indexes without using the report wizard.

Figure 7-3 shows how to initiate the report wizard function.

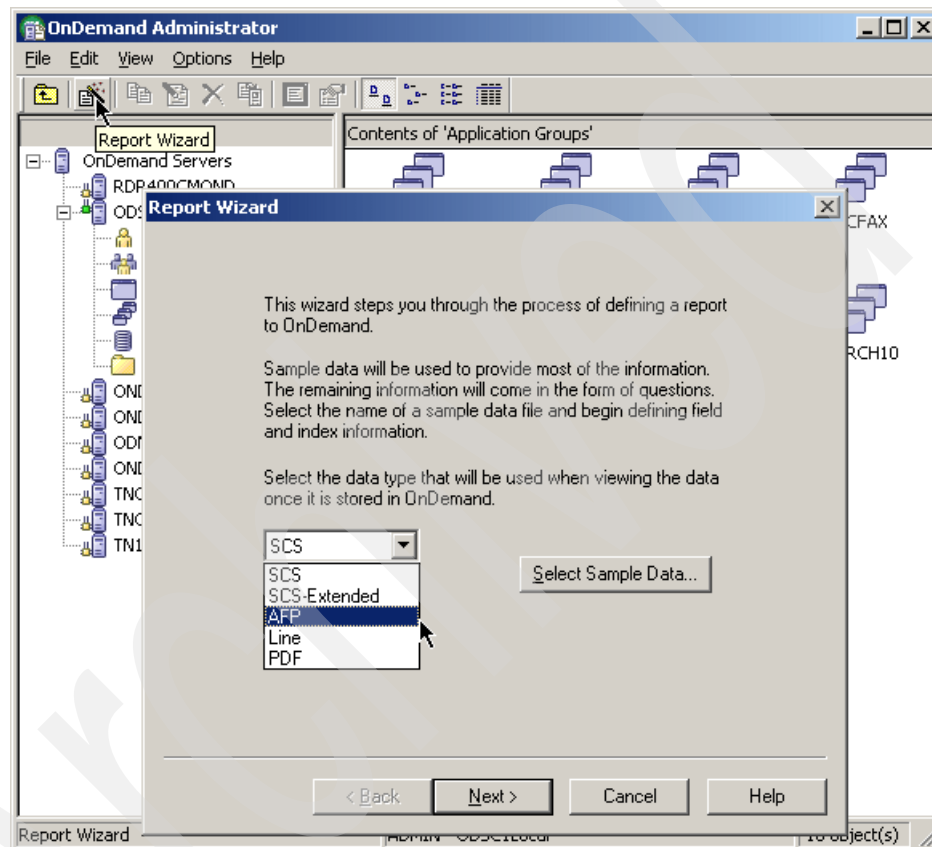


Figure 7-3 Report wizard

Other Common Server administration functions are managed through the iSeries Navigator interface. By installing the iSeries Navigator OnDemand plug-in, you will be able to graphically manage your migration policies, output queue monitors, tape and optical storage devices, storage volumes, and disk pools.

Figure 7-4 on page 189 shows an example of the options available through the iSeries Navigator plug-in.



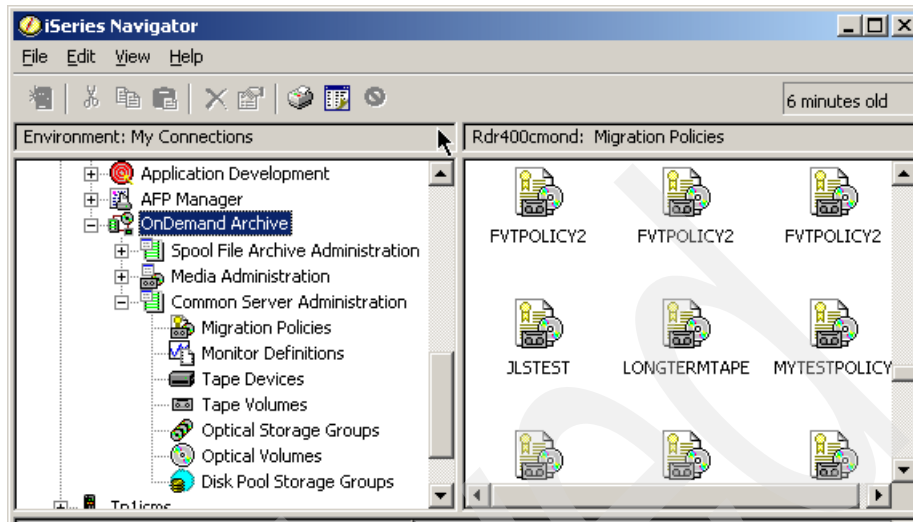


Figure 7-4 iSeries Navigator OnDemand plug-in

Note that from Figure 7-4, the OnDemand Navigator plug-in provides you with separate administration functions for Spool File Archive and Common Server.

## 7.4 User interface

OnDemand for iSeries provides you with the facility to grant your end-users access to the system through the Windows client, or through a Web browser interface.

Figure 7-5 on page 190 shows an example of a Windows client interface.

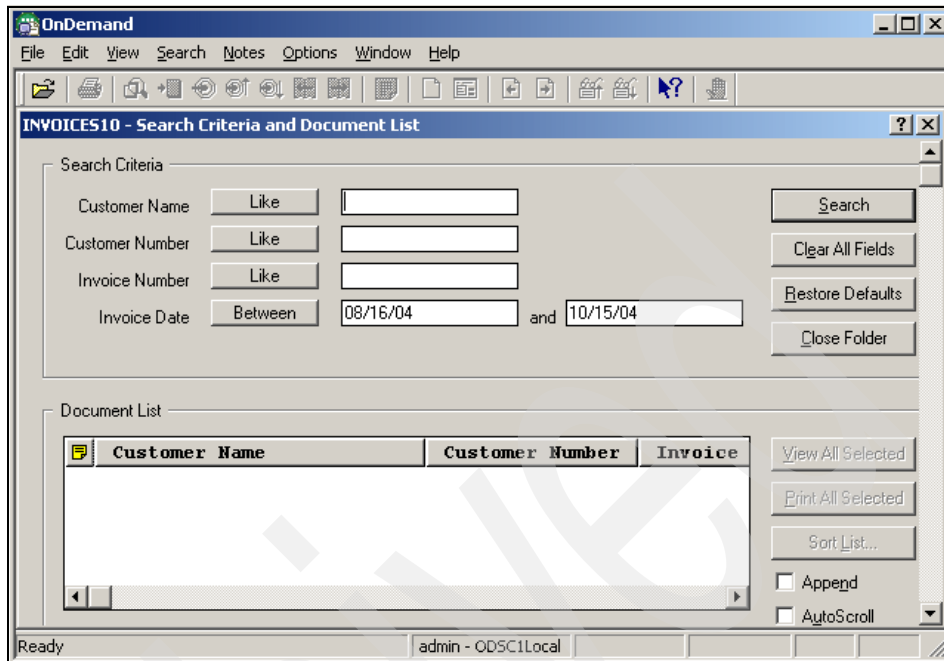


Figure 7-5 Windows end-user client interface

You need to install the OnDemand Windows client on each of your users' local workstations.

The Web browser interface requires that you implement ODWEK on your iSeries. Users can access the OnDemand system by using a standard compatible Web browser. There is no need to install additional OnDemand software on their workstations. Figure 7-6 on page 191 shows an example of the browser interface.

Depending on the data stream of the reports you need to work with, you may need to install the AFP plugin, the Image plugin, the Adobe Acrobat PDF plugin, and a Java runtime environment.



Figure 7-6 OnDemand Web browser interface

## 7.5 Summary

OnDemand for iSeries provides rich features with ease-of-use functionality.

For more information about the implementation and configuration of OnDemand for iSeries, including user manuals and guides, refer to:

<http://www.ibm.com/software/data/ondemand/400/library.html>



## Backup and recovery for OnDemand iSeries

In this chapter, we describe the backup and recovery strategies and options available for OnDemand for iSeries. We also discuss the system configuration files, system directories, and database libraries to backup, data recovery, general considerations, and the Backup, Recovery and Media Services (BRMS).

The procedures covered in this chapter include:

- ▶ Backup of OnDemand database, system files, and system directories
- ▶ Optical media backup
- ▶ Database recovery

## 8.1 Overview

Losing the information contained in your Content Manager OnDemand system could be detrimental to your enterprise. The loss of this data could cost your organization money and in some cases you could even be contravening certain laws and legal requirements.

Recovering lost data is a time-consuming exercise and you run the risk of not being able to recover all the required data. More time is spent verifying the integrity and completeness of the information, than the amount of time it actually takes to recover the data.

Establishing an effective backup and disaster recovery plan is of vital importance to ensure that your business-critical data remains secure and that your enterprise can function normally after a disaster.

Data loss can occur as a result of a total disaster (for example, the loss or destruction of your server) or as a result of a system failure, or data can be lost to human error, such as someone accidentally deleting the data. Irrespective of the cause of the data loss, it is imperative that you are able to recover your data in the shortest period of time in order to resume normal business operations as quickly as possible.

The appropriate disaster recovery strategy is usually based on two basic factors: how long the business can afford to operate without some or all of its data and how much it will cost to implement a particular disaster recovery plan.

Data backup and recovery is generally considered to be the least costly option but it could take a long time before your data is available again to your end-users. In this chapter, we provide some information to assist you in defining an effective backup and recovery strategy for your Content Manager OnDemand for iSeries system. We provide guidelines to help you in defining which objects to backup, what backup methods to use, and advise on certain recovery options.

A backup and recovery plan implies that you make a copy of your data at regular intervals.

You would typically save your data onto some sort of removable media (for example, tape cartridges) and store the copied data in a safe and secure environment. Then in case of a disaster, you would be able to use this saved data to restore your system to a consistent state and to a particular point in time prior to the disaster.

There are also other precautions that you can take in order to safeguard your data. Some of them include:

- ▶ Physically securing your server and server environment
- ▶ Installing redundant power supplies and an Uninterrupted Power Supply (UPS)
- ▶ Implementing disk mirroring or employing RAID technology
- ▶ Implementing an effective OS/400 data security model
- ▶ Making use of the OS/400 journaling feature
- ▶ Implementing a regular, planned system maintenance schedule
- ▶ Scheduling regular data backup operations

In the following sections we discuss backup and recovery of your OnDemand for iSeries system in more detail.

## 8.2 Database and system files backup

As explained in Chapter 7, “OnDemand for iSeries overview” on page 183, OnDemand for iSeries is implemented as an OS/400 licensed program and it stores its configuration objects and user data in OS/400 libraries and in IFS directories.

It is therefore essential that all these objects are saved in a valid and consistent state to ensure full recoverability of your OnDemand system.

### 8.2.1 System save

Although OS/400 licensed programs can be re-installed from the OS/400 shipped media, we recommend that you schedule regular system save operations to ensure that you can easily and quickly recover your system after a disaster. This is important since the original media does not include any PTFs. If you re-install from the shipped media, you would have to find and re-apply all the PTFs, which can take a considerable amount of time.

Any system modifications and configuration changes that you might have implemented in any of the OS/400 system libraries and directories will be lost if you re-install your system from the IBM-supplied OS/400 media. In addition, certain OS/400 objects, including user profiles and authorization lists, are automatically created when you configure your OnDemand instances. These objects cannot be saved individually.

You can initiate an entire system save by selecting **Option 21** on the OS/400 SAVE menu. You can schedule this save operation to run at a pre-determined time and let it run unattended, that is it does not require the attention of an operator.

**Note:** Option 21 requires that your iSeries system is in a restricted state for the duration of the save operation.

Table 8-1 shows a list of objects that are automatically saved when you use Option 21, together with the corresponding OS/400 commands to save the particular individual objects.

*Table 8-1 System save commands*

| Item saved                                  | Save commands                                                                                                                       |
|---------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| Licensed internal code                      | SAVSYS                                                                                                                              |
| All objects in QSYS                         | SAVSYS                                                                                                                              |
| User profiles including private authorities | SAVSYS or SAVSECDTA                                                                                                                 |
| Authorization lists                         | SAVSYS or SAVSECDTA                                                                                                                 |
| Configuration objects                       | SAVSYS or SAVCFG                                                                                                                    |
| IBM-supplied directories                    | SAV                                                                                                                                 |
| OS/400 optional libraries                   | SAVLIB<br>Using the command, SAVLIB *IBM, this saves all IBM supplied libraries except those that typically contain user data.      |
| Licensed program libraries (libraries)      | SAVLIB<br>Using the command, SAVLIB *IBM, this saves all IBM supplied libraries except those that typically contain user data       |
| IBM-supplied libraries containing user data | SAVLIB<br>Using the command, SAVLIB *ALLUSR, this saves libraries that typically contain user data, including some supplied by IBM. |
| User libraries                              | SAVLIB<br>Using the command, SAVLIB *ALLUSR, this saves libraries that typically contain user data, including some supplied by IBM. |
| Documents and folders                       | SAVDLO                                                                                                                              |
| User objects in directories                 | SAV                                                                                                                                 |
| Distribution objects                        | SAVDLO                                                                                                                              |



To restore your entire system from the backup media created by the Option 21 Save operation, you would select **Option 21** on the OS/400's RESTORE menu.

## 8.2.2 OnDemand libraries

As mentioned in 7.2, "Installation and configuration" on page 185, OnDemand for iSeries makes use of OS/400 libraries and IFS directories to store system data, user data, and configuration data.

Your OnDemand system typically contains the following OS/400 libraries:

- ▶ QRDARS
- ▶ QUSRRDARS
- ▶ OnDemand instance libraries

### ***QRDARS***

The *QRDARS* library is automatically created when you install your OnDemand system. It contains IBM-supplied application programs and other OS/400 objects that are required for your OnDemand system to function effectively. Although this library can be re-created by reinstalling the OnDemand licensed program, we recommend that you take a backup copy of this library on a regular basis so you can restore it in case of a hardware failure or disaster.

The following OS/400 command shows an example of how you can save this library and all its contents:

```
SAVLIB LIB(QRDARS) DEV(<device name>)
```

**Important:** It is important to realize that if you reinstall your OnDemand system from the original installation media, you will lose all IBM-supplied OnDemand PTFs (fixes) that you have implemented since the original installation. We recommend you save your system using the OS/400 Save Option 21. This ensures that all PTFs would be saved and your system can be restored with all your PTFs implemented.

### ***QUSRRDARS***

The *QUSRRDARS* library is also automatically created when you install your OnDemand system. However, it contains your customized OnDemand system configuration data, including details about your storage groups, migration policies, archive media, user settings and system default settings. Most of this information is contained in database tables, user spaces and data areas. It is therefore very important that you have an up-to-date backup copy of this library in order to be able to fully recover your OnDemand system.

The following OS/400 command shows an example of how you can save this library and all its contents:

```
SAVLIB LIB(QUSRRDARS) DEV(<device name>)
```

The QUSRRDARS library also contains the OS/400 journal QRLCJRN, together with its associated journal receivers. This journal is used for commitment control and to journal changes to some of the objects contained in this library. It is therefore important that all objects contained in this library are saved at the same time and in a consistent state.

### ***OnDemand instance libraries***

When you create an OnDemand instance, a library with the same name as the OnDemand instance is automatically created. This library is used to store the configuration information for the instance, including information relating to your application groups, applications, folders, and archived reports. The index information required to store and retrieve your archived data is also stored in tables in this library.

The tables in this library are journaled to an OS/400 journal called QSQJRN which is created at the time the instance is created. This journal and its associated journal receivers are also contained in the instance library.

In order to ensure that you can restore your OnDemand instance to a consistent state after a disaster, it is important that you save all the objects in your instance library, as well as all the objects in the QUSRRDARS library at the same time.

The following OS/400 command shows an example of how you can save your OnDemand instance libraries and the QUSRRDARS library:

```
SAVLIB LIB(<library name>) DEV(<device name>)
```

**Note:** Starting at OS/400 V5R3, the default OnDemand instance QUSROND is no longer automatically created when you install OnDemand for iSeries.

We recommend that you make regular and frequent backup copies of your OnDemand libraries, or at least every time you update application group, application or folder definitions and after any report data has been archived or expired.

## **8.2.3 OnDemand directories**

When you install and configure your OnDemand system and instances, a number of IFS directories and subdirectories are automatically created. We discuss these directories in more detail in this section.

## OnDemand system directory

The OnDemand system directory contains the OnDemand system configuration data, templates to assist you to configure your OnDemand instances and symbolic links to the application programs in the QRDARS library. Figure 8-1 shows an example of this directory.

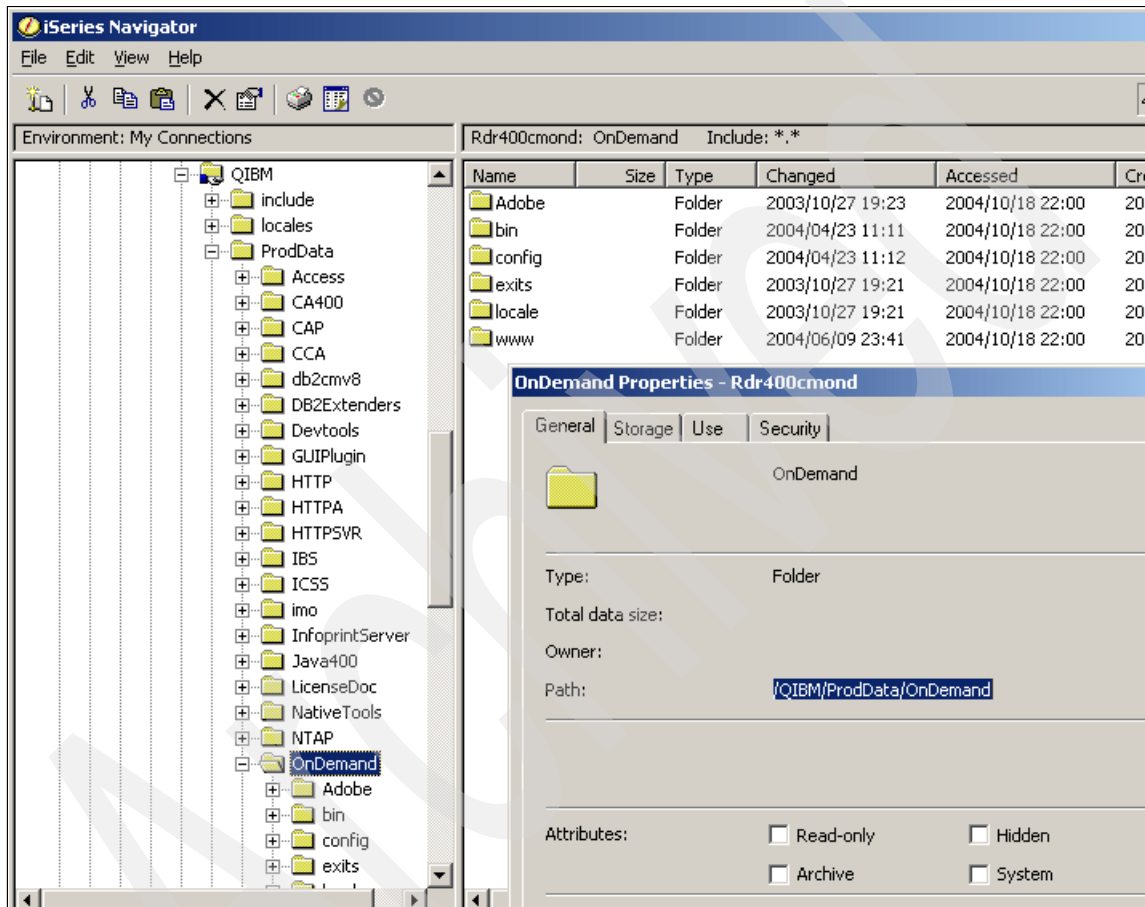


Figure 8-1 OnDemand system directory

You should make a backup copy of this directory and especially after implementing changes that affect your overall OnDemand system, for example after applying PTFs.

**Note:** We do not recommend that you modify any of the objects contained in this directory. User changes might be lost during future upgrades or when applying IBM-supplied PTFs. Instead, you should copy templates and other objects contained here into your instance directory and modify these to suit your requirements.

The following shows an example of how you could make a backup copy of this directory and the contents of all its sub-directories:

```
SAV DEV('/QSYS.LIB/<device name>.devd') OBJ('/QIBM/ProdData/OnDemand'
*INCLUDE))+ SUBTREE(*ALL)
```

### OnDemand user directories

The OnDemand user directory contains information relevant to your particular OnDemand installation and instances, including your customized configuration and settings.

The configuration directory is under /QIBM/UserData/OnDemand/CONFIG. The ARS.INI file, within this directory, contains important configuration data.

When you create an OnDemand instance, a sub-directory with the same name as the instance is automatically created. Figure 8-2 on page 201 shows an example of a typical OnDemand instance directory.

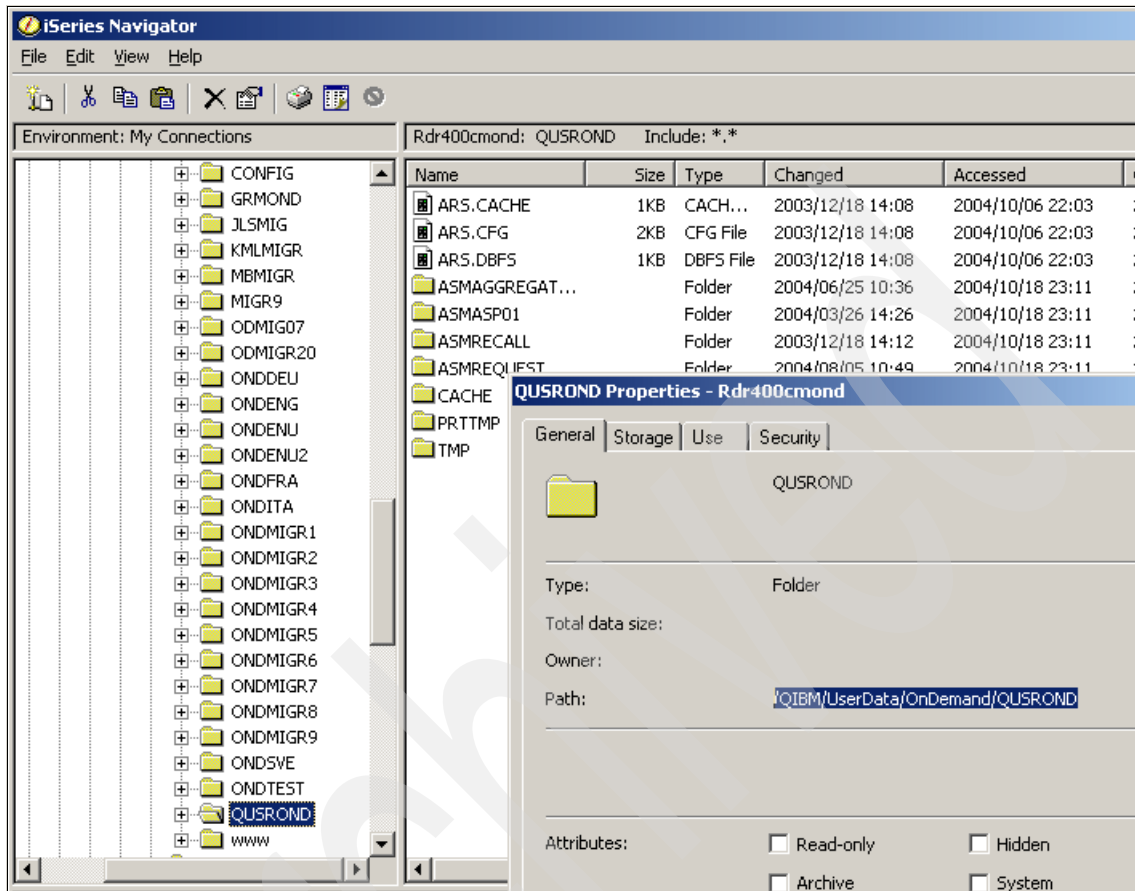


Figure 8-2 OnDemand instance directory

This instance directory contains configuration data pertaining to the particular OnDemand instance, as well as data stored by the Archive Storage Manager (ASM) and report data stored in cache.

We recommend that you frequently make a backup copy of this directory and at the same time that you save the OnDemand instance libraries and the QUSRRDARS library. The following shows an example of how you can make a backup of this directory:

```
SAV DEV('/QSYS.LIB/<device name>.devd') OBJ('/QIBM/UserData/OnDemand'
*INCLUDE)) SUBTREE(*ALL)
```

To make a backup copy of any particular OnDemand instance directory, you could specify the OnDemand instance name, for example:

```
SAV DEV('/QSYS.LIB/<device name>.devd')
OBJ('/QIBM/UserData/OnDemand/<instance_name>' *INCLUDE)) SUBTREE(*ALL)
```

As a general rule of thumb, you should make a backup copy of your instance directory if any of the following situations have occurred:

- ▶ You have archived new report data or other objects.
- ▶ You have run Archive Storage Manager (ASM). Note, ASM typically changes the contents of directories, even if data is not expired.
- ▶ You have manually deleted report data.
- ▶ You have installed or configured a new OnDemand instance.
- ▶ You have modified any settings of your existing OnDemand instances.

To facilitate the management of system performance and backup requirements, you can configure your servers to contain multiple basic user pools, commonly known as user Auxiliary Storage Pools (ASPs). This is typically done by grouping together a number of disk units and assigning that group to a disk pool. Basic disk pools can contain libraries, documents, and certain other types of objects. The system automatically creates the system disk pool (ASP1) which comprises of disk unit 1 and all other configured disks that are not assigned to a basic disk pool.

The system disk pool contains all system objects for the OS/400 licensed program and all user objects that are not assigned to a basic or independent disk pool.

The data in a basic user pool is always accessible whenever the iSeries server is up and running. For more information about basic user pools, refer to the iSeries Information Center at:

<http://publib.boulder.ibm.com/pubs/html/as400/>

When you make use of ASPs in your migration policy, the OnDemand Archive Storage Manager automatically creates and mounts a User Defined File System (UDFS) when the disk pool is required for the first time.

To ensure easy recoverability of your OnDemand system, you should take a backup of the relevant UDFS together with your OnDemand instance directory. We recommend that you execute the following steps to save your UDFS:

1. Unmount the UDFS, using the system-supplied program *QRLCASMUFS*. For example:

```
CALL QRDARS/QRLCASMUFS PARM('<instance name>') or
CALL QRDARS/QRLCASMUFS PARM(*ALL)
```

**Attention:** The QRLCASMUFS program will end the specified OnDemand instance(s) and unmount the relevant UDFS.

You may also use the **UNMOUNT** OS/400 command to unmount your UDFS, for example:

```
UNMOUNT TYPE(*UDFS) MNTOVRDIR('/QIBM/UserData/OnDemand/<instance name>
/ASMASPnn/PRIMARY')
```

2. Save the directory structure /dev/QASPnn/ONDEMAND\_<instance name>\*, for example:

```
SAV DEV('/QSYS.LIB/tap01.devd') OBJ(('dev/QASP03/ONDEMAND_QUSROND*'))
```

**Note:** You should make sure that you have the latest PTFs installed which will allow ASM to automatically mount the UDFS when it calls any process that requires the UDFS. The required PTFs are as follows:

- ▶ V5R1 - SI14504
- ▶ V5R2 - SI14486
- ▶ V5R3 - SI14507

To restore your UDFS, we recommend that you execute the following steps:

1. Unmount the UDFS, by using the system-supplied program *QRLCASMUFS*. For example:

```
CALL QRDARS/QRLCASMUFS PARM('<instance name>') or
CALL QRDARS/QRLCASMUFS PARM(*ALL)
```

**Attention:** The QRLCASMUFS program will end the specified OnDemand instance(s) and unmount the relevant UDFS.

You may also use the **UNMOUNT** OS/400 command to unmount your UDFS, for example:

```
UNMOUNT TYPE(*UDFS) MNTOVRDIR('/QIBM/UserData/OnDemand/<instance name>
/ASMASPnn/PRIMARY')
```

2. Restore the directory structure /dev/QASPnn/ONDEMAND\_<instance name>\*, for example:

```
RST DEV('/QSYS.LIB/tap01.devd') OBJ(('dev/QASP03/ONDEMAND_QUSROND*'))
```

## OnDemand Web Enablement Kit (ODWEK)

If you have implemented ODWEK, it is important to ensure that you also have a valid backup copy of your HTTP configuration so that you can easily recreate your HTTP server after a disaster has occurred. Figure 8-3 on page 204 shows an example of this directory.

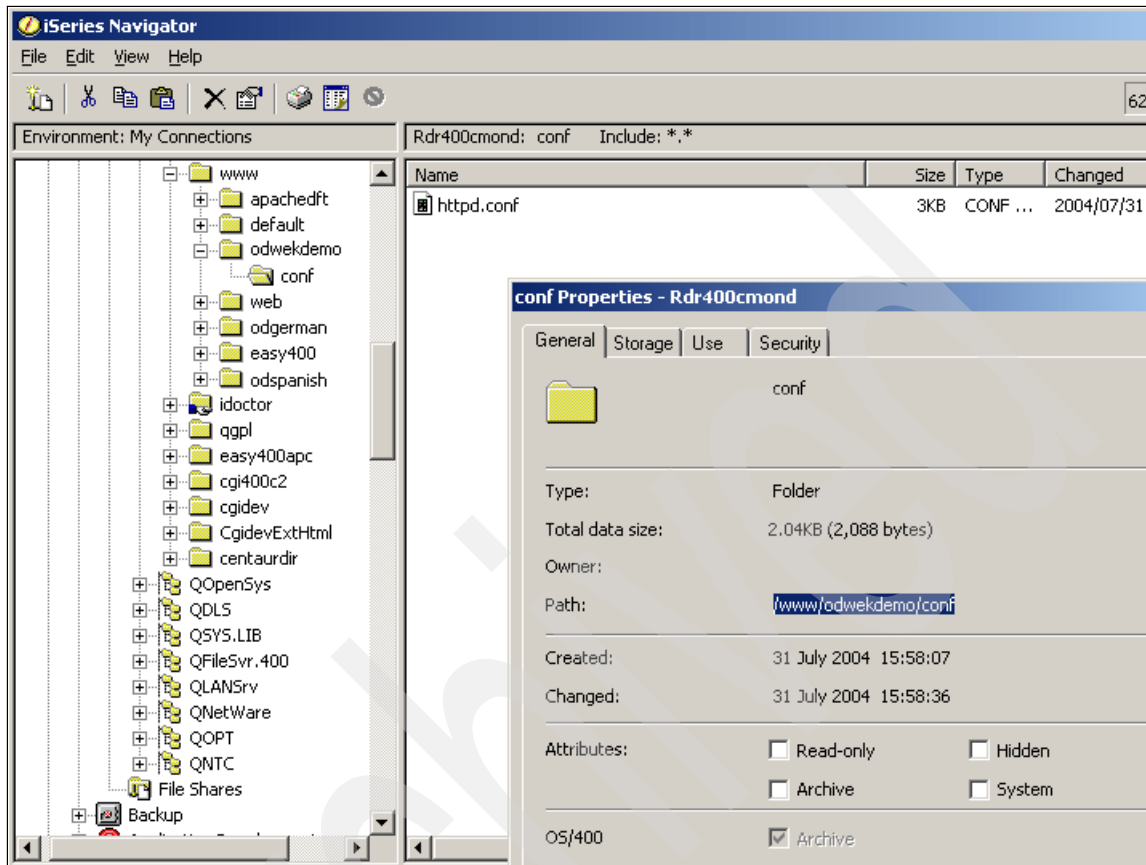


Figure 8-3 HTTP configuration directory

You can make a backup copy of this directory by using the SAV OS/400 command, for example:

```
SAV DEV('/QSYS.LIB/<device name>.devd') OBJ('/www/<http server name>'
*INCLUDE)) SUBTREE(*ALL)
```

With ODWEK installed and configured, both the OnDemand system directory and OnDemand instance directory also contain information relevant to ODWEK. These directories will be saved automatically if you perform the save operations as explained in “OnDemand system directory” on page 199 and “OnDemand user directories” on page 200.



## Changed objects

If you have a limited amount of time each day in which to make backup copies of your OnDemand libraries and directories, you may want to consider the following approach.

Make a full backup copy of your libraries and directories once every week using the OS/400 commands **SAVLIB** and **SAV**. You could then save only the changes that might have occurred in the libraries and directories each day for the rest of the week. To save only those objects that had changed in each library since the last time the library had been saved using the SAVLIB OS/400 command, you would specify the OS/400 **SAVCHGOBJ** command as shown in the following example:

```
SAVCHGOBJ OBJ(*ALL) LIB(<library name>) DEV(<device name>)OBJJRN(*YES)
REFDATE(*SAVLIB) UPDHST(*YES)
```

Similarly, to save only the changed objects in your OnDemand directories, you would specify the OS/400 **SAV** command as shown in the example below:

```
SAV DEV('/QSYS.LIB/<device name>.devd') OBJ(('<directory name>' *INCLUDE))
SUBTREE(*ALL) CHGPERIOD(*LASTSAVE) UPDHST(*YES)
```

## 8.3 Optical media backup

Securing your archived data that reside on your optical disks is as important as securing your Content Manager OnDemand database tables and directories. There are a number of options available to you for making backup copies of your optical volumes. These include some of the following:

- ▶ Duplicate optical media
- ▶ Copy objects
- ▶ Copy optical media contents
- ▶ Migration policy backup options

### Duplicate optical media

The OS/400 command **DUPOPT** allows you to create a duplicate copy of your optical volumes. The duplicated optical volume is identical to the original volume, except the volume identifier and the creation time would be different. The following shows an example of how to use this command:

```
DUPOPT FROMVOL('<source volume identifier>') TOVOL('<target volume
identifier>') FROMDEV(<source device>) TODEV(<target device>)
```

The following are some considerations that you should bear in mind when using the **DUPOPT** command:

- ▶ You need at least two drives in your optical Library Server.

- ▶ The two allocated drives of the optical library server will not be accessible to other applications while the **DUPOPT** command is active.
- ▶ The **DUPOPT** command does support duplicating Write Once Read Many (WORM) format optical volumes to Rewritable (REWT) format volumes.
- ▶ If your target volume is of WORM format, the most efficient utilization of the target volume occurs when the source volume is completely full.
- ▶ The source and target volumes may be of different media densities.
- ▶ The amount of time it takes to create a copy of your optical volume when using the **DUPOPT** remains virtually constant, irrespective of the number of files or the size of the files contained on your optical volume.
- ▶ It is not possible to use the **DUPOPT** command to make a partial copy of the data contained on your optical volume. In other words, all the data contained on the source volume is copied to the target volume.

## Copy objects

The OS/400 command **SAV** also enables you to make volume level backup copies of the data contained on your optical volumes.

You can use this command to save your optical volume data to another optical volume, or to a tape device, or a savefile. The following shows an example of how you could use this command:

```
SAV DEV('QSYS.LIB/<device name>') OBJ(('*' *INCLUDE)) SUBTREE(*STG)
OPTFILE('<optical directory path>')
```

If you backup to a savefile (\*SAVF) which is on disk, the savefile can then be transmitted electronically to a remote system. Once there, it could be saved to tape, or restored onto another optical volume, thus providing an off-site backup volume.

The following are some considerations that you should bear in mind when using the **SAV** command:

- ▶ The **SAV** command allows you to duplicate data from a source optical volume to a target optical volume even if you only have a single drive in your optical Library Server.
- ▶ You may use the **SAV** command to backup your optical volume data to a tape device or a savefile whenever new data had been archived to your optical volume. Then, only when your optical volume is completely full, can you use the **SAV** or **DUPOPT** OS/400 command to save the entire contents of the optical volume to a backup optical volume.

- ▶ The amount of time it takes to create a copy of your optical volume when using the **SAV** remains virtually constant, irrespective of the number of files or the size of the files contained on your optical volume.
- ▶ You can take the backup optical volume off-site.

## Copy optical media contents

The **CPYOPT OS/400** command enables you to copy the contents of one optical volume to another optical volume. When using this command to copy data between optical media, the directory names and file names do not change during the copy operation. The creation dates and modification dates of the directories and files are also copied to the target disk and remain the same as on the source disk. The following shows an example of how to use this command:

```
CPYOPT FROMVOL('<source volume>' *PRIMARY) FROMPATH('/')
TOVOL('<target volume>' *BACKUP) SLTFILE(*ALL) CPYSUBDIR(*YES) ALWCPYOPP(*NO)
```

You can also use this OS/400 command to create duplicate *Primary* optical volumes by specifying the parameter **FROMVOL('<source volume>' \*PRIMARY)**, or to make incremental backup copies of the data on your optical media by specifying the **SLTFILE(\*CHANGED)** parameter.

## Migration policy backup options

Your OnDemand system has built-in functionality that can automatically create backup copies of your archived data to a tape storage device. All you have to do is specify the appropriate settings in your migration policies. Figure 8-4 on page 208 shows an example of how you can do this.

**Migration Policy Properties - Rdr400cmond**

Policy name: D90OPTICAL

Description: DASD for 90 days then Optical

☒ Enable aggregation

Maximum size: 1000 kilobytes

☒ Close aggregate only when maximum size reached

☐ Close aggregate after specified time period

Time period: 0 days

☒ Tape backup requested

Media type: QIC2GB, QIC13GB, QIC25GB, QIC50GB, REEL, 8MM, 3570CART, 3570CARTE

Instance:

| Level | Disabled | Media   | Days | Primary Group | Description |
|-------|----------|---------|------|---------------|-------------|
| 0001  | Yes      | Optical | 900  | RDARSD        | Optical     |

Buttons: Add Before..., Add After..., Change..., Remove, OK, Cancel, Help

Figure 8-4 Migration policy backup option

If you select the option to make a backup copy to tape, you need to specify the appropriate tape media type that you will be using. By specifying this option, a one-time backup copy of your archived data is automatically created for data that is archived using this migration policy.

OnDemand also gives you the ability to create a primary, as well as a backup storage group. Figure 8-5 on page 209 shows an example of primary and backup optical storage groups.

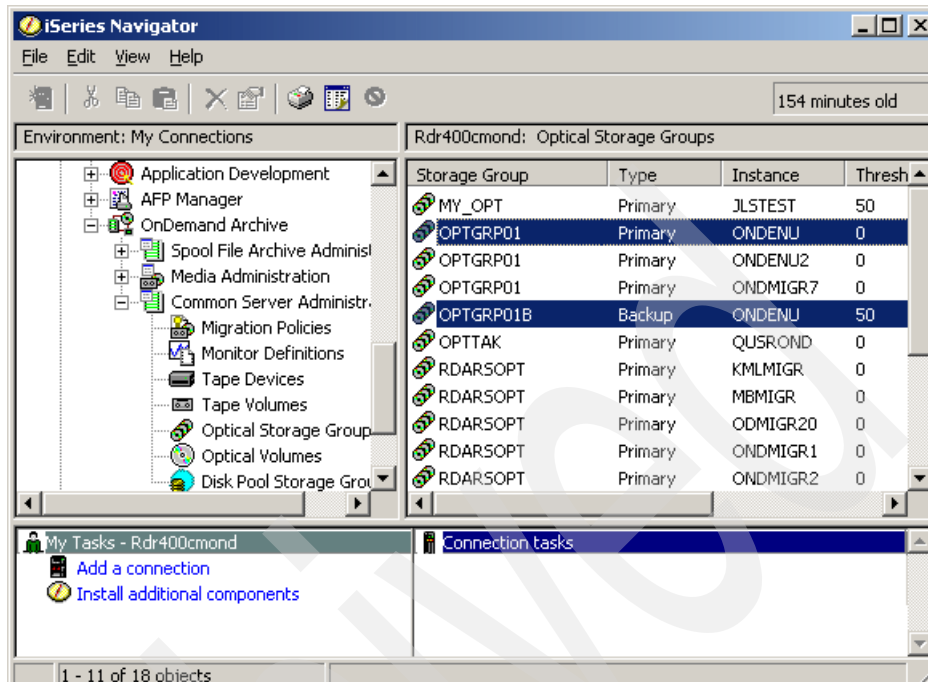


Figure 8-5 Optical storage groups

OnDemand also supports backup storage groups for disk media and tape media.

This feature enables your OnDemand system to perform dual write operations. In other words, your data will be written to your primary storage media and a copy will automatically be written to the backup media.

Be cautious. Dual write is not guaranteed to produce two identical volumes, whether tape or optical. There are reasons that they may not be identical, including drive errors, media errors, volume type, and capacity differences. It may not necessarily be the best method for creating backup copies of removable media volumes.

To make use of this functionality, you need to specify the backup storage group in the storage level of your migration policy. Figure 8-6 on page 210 shows an example of this.

| Level | Media     | Days   | Primary Group | Backup Group | Description     |
|-------|-----------|--------|---------------|--------------|-----------------|
| DISK  | Disk Pool | 30     | ASP01         |              | Disk Pool       |
| OPT   | Optical   | 999999 | OPTGRP01      | OPTGRP01B    | Optical Library |

Figure 8-6 Storage level backup

You will notice that in this particular example, a backup storage group had been specified for the OPT storage level. This is a storage group that had previously been defined as a backup storage group, as depicted in Figure 8-5 on page 209.

## 8.4 Optical media considerations

The following are some optical media related recommendations that will assist you in securing the data on your optical media, as well as some performance guidelines (note, some are already mentioned earlier in the chapter but are repeated here due to their importance):

- ▶ Make use of volume level backup whenever possible by using the **DUPOPT** or **SAV OS/400** commands.
- ▶ If you must use file level backup (**CPYOPT**), use the **SLTFILE(\*CHANGED)** parameter in conjunction with the **FROMTIME(mmddyy)** parameter.
- ▶ An advantage of file level backup is that incremental backup is possible.
- ▶ To optimize file level optical backup:

- Configure each optical library with a minimum of two optical drives.
  - Perform copying operation during periods of low activity for optical applications.
  - Do not specify the source volume's opposite side volume as the target volume.
  - Specify the **COPYTYPE(\*IOP)** parameter of **CPYOPT**.
  - Mount the source and target optical disks in the same optical Library Server when performing copy operation.
  - Specify incremental file copying using the **SLTFILE(\*CHANGED)** parameter of **CPYOPT**.
- ▶ Store your backup optical disks and tape volumes in a safe and separate location from your source disks.
  - ▶ The primary determinant of copy performance, when using the **CPYOPT** command, is the size and quantity of files selected to be copied: The more and smaller the files that are to be copied, the longer the time to complete the copy.
  - ▶ The speed and performance of the OS/400 commands **DUPOPT** and **SAV** are not relative to the file size or the number of files contained the source optical disk.
  - ▶ The **DUPOPT** OS/400 command does not allow you to make partial or incremental copies. It produces an exact replica of the source optical disks.
  - ▶ The **DUPOPT** OS/400 command requires at least two optical drives to be configured in the same optical Library Server. It also requires exclusive use of the optical drives for the duration of the copy operation.
  - ▶ You may duplicate WORM optical volumes to re-writable optical volumes when using the **DUPOPT** OS/400 command. When duplicating WORM media, the best utilization efficiency is obtained when the source volume is completely full.
  - ▶ The **SAV** OS/400 command allows you to copy optical media, even if you have only one optical drive. It also supports copying optical volumes to tape devices and savefiles and allowing incremental backup of your data.
  - ▶ If you have created a backup optical volume from a primary optical volume by using the **CPYOPT** OS/400 command, you need to convert the backup volume to a primary volume and reset the volume ID to match that of the original primary optical volume before you will be able to retrieve the data. The following shows an example of the OS/400 command that you can use to do this:

```
CVTOPTBKU BKUVOL('<backup volume ID>') PRIVOL(*PRVPRIVOL)
```

- In the case of destruction of the \*PRIMARY cartridge after using **CPYOPT**, the backup cartridge should be returned to the optical library. Then use the **CVTOPTBKU** command to change the cartridge type to **\*PRIMARY** and to change the volume ID to match the original cartridge volume ID.

More information about optical backup is available at the following URL:

<http://www.ibm.com/servers/eserver/series/optical/backup/backup.htm>

Other sources of reference include:

- *iSeries Backup and Recovery Guide*, SC41-5304
- *V5R3 OS/400 Optical Support*, SC41-5310-04

## 8.5 Database recovery

The method that you will employ to restore your data after a disaster largely depends on how you originally saved your data.

If you want to recover your system from the media that had been created by the OS/400 **Save Option 21**, you would typically use **Option 21** from the OS/400 **RESTORE** menu. You can restore any library that had been saved by the OS/400 **SAVLIB** command by using the OS/400 command **RSTLIB**, for example:

```
RSTLIB SAVLIB(<library name>) DEV(<device name>)
```

Any directory that you have previously saved using the OS/400 command **SAV**, can be restored using the OS/400 command **RST**, for example:

```
RST DEV('/QSYS.LIB/<device name>.devd') OBJ(('<directory name>' *INCLUDE))
SUBTREE(*ALL)
```

**Important:** To recover your OnDemand system to a consistent state after a system failure, it is important that you restore the libraries and directories that had been backed up at the same time, that is during the same backup windows while no archive operations were taking place. If the SAVLIB was executed at a different time from the SAV, then the libraries and directories are probably out of sync and you may not be able to properly recover your system.

It is also important to avoid restoring individual objects into any of your OnDemand libraries or directories, as this may cause inconsistencies in your system.

If you need to restore your OnDemand system and you have used the option to save only those objects that have changed on a daily basis, you would typically proceed as follows:



1. Restore the last full save of your libraries using the OS/400 command **RSTLIB**. For example:

```
RSTLIB SAVLIB(<library name>) DEV(<device name>)
```

2. Restore the last complete save of your directories by using the OS/400 command **RST**. For example:

```
RST DEV('/QSYS.LIB/<device name>.devd') OBJ(('<directory name>' *INCLUDE))
```

3. Restore the last backup set of your changed library objects using the OS/400 command **RSTOBJ**. For example:

```
RSTOBJ OBJ(*ALL) SAVLIB(<library name>) DEV(<device name>)
```

4. Restore the last backup of any User Defined File Systems (UDFS). For example:

```
RST DEV('/QSYS.LIB/<device name>.devd') OBJ(('dev/QASPnn/ONDEMAND_<instance name>*'')).
```

“OnDemand user directories” on page 200 described the process for saving and restoring UDFS in more detail.

5. Restore the last backup of your changed directories using the OS/400 command **RST**. For example:

```
RST DEV('/QSYS.LIB/<device name>.devd') OBJ(('<directory name>' *INCLUDE))
```

Following the successful recovery of your libraries and directories, we recommend that you execute the following program:

```
CALL QRDARS/QRLCSTRJ PARM(RLC)
```

This ensures that all the required OnDemand tables are properly journaled.

You may also need to re-synchronize your optical library and the optical index database after the restore operation is completed. You can do this by issuing the following OS/400 command:

```
RCLOPT MLB(<optical library name>) OPTION(*SYNC) VOL(*ALL)
```

If the disaster occurred while any archiving jobs were active, for example **STRMONOND** jobs, these jobs would have terminated and it is possible that not all your report data had been fully archived. If this is the case, you should delete any reports that were being archived by the terminated job, using the **RMVRPTOND** OS/400 command and then recreate the report data and re-archive these reports.

## 8.6 Backup, Recovery and Media Services (BRMS)

Backup, Recovery and Media Services (BRMS) for iSeries is an OS/400 licensed program that can help you manage your backups and backup media through a structured approach. It also enables you to retrieve lost or damaged data easily.

With BRMS, you can easily manage your backup and recovery, including online backups and recovery of your entire iSeries system in the event of a disaster or failure. In addition, BRMS provides facilities to keep track of the backup media from creation to expiration. You no longer have to manually keep track of which backup items are on which volumes. It helps to eliminate the risk of accidental deletion or overwriting active data. It can also manage and perform the daily maintenance activities associated with the backup routine.

With a plug-in to the iSeries Navigator interface, BRMS makes managing your backups even easier. There are several wizards you can use to simplify the common tasks you need to perform. These include creating backup policies, adding media to BRMS and preparing it for use, adding items to backup policies, creating a move policy, restoring saved items and reclaiming media.

More information about BRMS is available at:

<http://www.ibm.com/servers/eserver/iseries/service/brms/>



## High availability strategies and options for the OnDemand iSeries

In this chapter, we describe strategies and options to achieve high availability for an IBM DB2 Content Manager OnDemand (OnDemand) implementation for iSeries.

The following topics are covered in this chapter:

- ▶ Journaling
- ▶ Remote journaling
- ▶ High Availability Business Partner solutions

## 9.1 Introduction

The concept of high availability refers to a business strategy designed to ensure minimal disruption to an organization's business processes by reducing the impact of computer system failures on its business processes.

Because the focus is to prevent system failures from occurring in the first place, the tendency is often first to address and reduce potential system hardware failures by eliminating any single point-of-failure (SPOF). This is normally achieved by configuring redundant hardware components and servers. From a software perspective, you can make use of clustering and coupling to reduce the single-point-of-failure of your mission-critical applications. Very often, you make use of third party software solutions to ensure that your business applications remain highly available to your end-users.

Implementing an high availability solution, however, can prove to be expensive. You should therefore ensure that the cost of your solution is proportional and relevant to the savings and benefits that it will afford your business in the event of a disaster.

OnDemand for iSeries relies on a number of built-in features that iSeries has in providing a more highly available service to your end-users, including journaling and remote journaling.

## 9.2 Journaling

*Journaling* is a feature of OS/400 that ensures data integrity and recoverability. This support can further be extended to reduce the duration of an abnormal system end. Journaling also serves as the cornerstone upon which many data replication and data warehouse refresh-schemes are built.

Journaling can also provide a mechanism to minimize the possibility of the loss of critical enterprise data. You can achieve this by journaling not only your user database tables but also other supported objects, such as data areas, data queues, and your IFS contents.

### **Journaling and OnDemand**

OnDemand on the iSeries makes use of database triggers, referential integrity and commitment control to ensure database consistency and recoverability. To facilitate this, many of the more critical database tables of the OnDemand system are by default automatically journaled. Table 9-2 on page 219 shows some of the journals that are automatically created when you install and configure your OnDemand system.

Table 9-1 Default OnDemand journals

| Journal | library                     |
|---------|-----------------------------|
| QRLCJRN | QUSRRDARS                   |
| QSQJRN  | QUSROND                     |
| QSQJRN  | <OnDemand instance library> |

**Note:** Starting at OS/400 V5R3, the default OnDemand instance QUSROND is no longer automatically created when you install OnDemand for iSeries.

You can easily extend this journal functionality to your other OnDemand objects. The following OS/400 objects types currently support journaling:

- ▶ Database tables
- ▶ Data areas
- ▶ Data queues
- ▶ Integrated File System (IFS) objects

Figure 9-1 on page 218 shows an example of how you can activate journaling on a database table by using the iSeries Navigator interface.

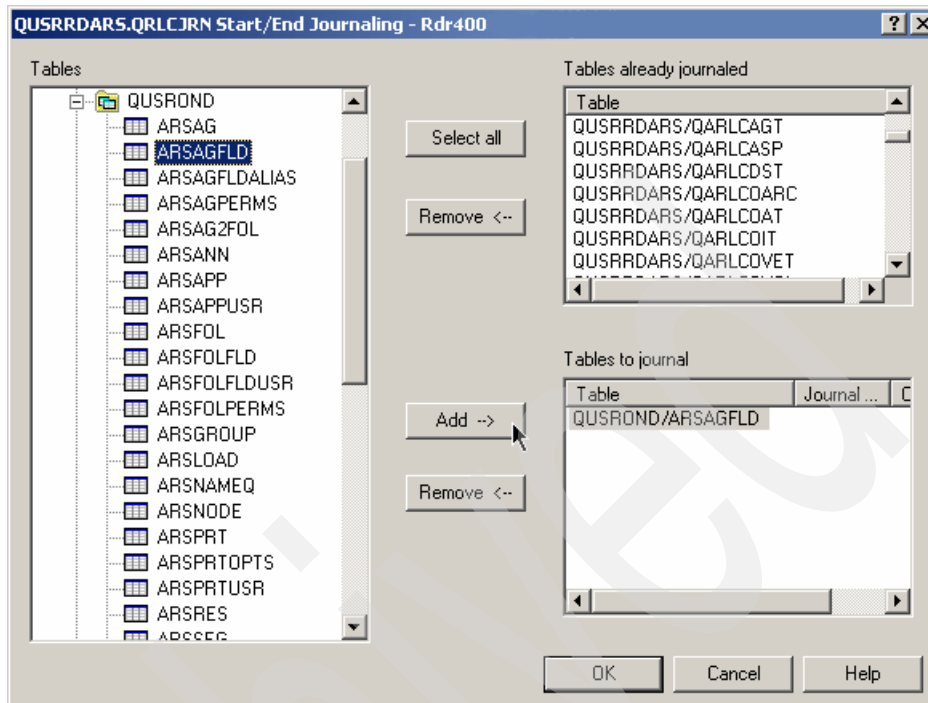


Figure 9-1 Journaling a database table

Alternatively, you could also use the **STRJRNPF** OS/400 command as shown below:

```
STRJRNPF FILE(<schema name>/<database table name>)
JRN(<library_name>/<journal_name>) IMAGES(*BOTH) OMTJRNE(*OPNCLO)
```

The OnDemand system makes extensive use of the IFS to store objects and data. Recovery of your IFS objects is therefore as important as recovery of your database tables. Figure 9-2 on page 219 shows an example of how you can activate journaling on an IFS object using the iSeries Navigator interface.

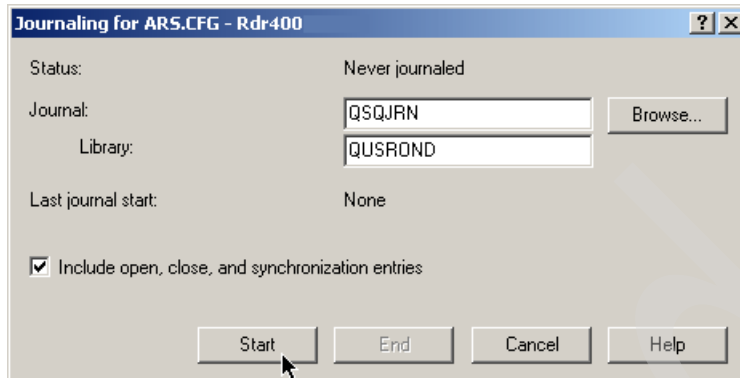


Figure 9-2 Journaling an IFS object

You could also use the OS/400 command **STRJRN** to activate journaling on IFS object. For example:

```
STRJRN OBJ ((' /QIBM/UserData/OnDemand/QUSROND/'))
JRN (' /QSYS.LIB/QUSROND.LIB/QSQJRN.JRN') SUBTREE (*ALL)
```

Similarly, you can use the iSeries Navigator interface or the **STRJRNOBJ** OS/400 command to start journaling on any data area or data queue.

## Recovering your system using journal changes

Once you have started journaling your objects, changes made to your objects are recorded in the journal receiver attached to the journal.

It is important that you regularly and frequently save your journal receivers onto the backup media so that you can recover your data to a specific point in time. You can use the **SAV0BJ** OS/400 command to save your journal receivers. We present the following hypothetical scenario to help you better understand the concept.

Your current business processes only allow you to backup your data every Saturday. You are journaling all your required OnDemand information such as database tables and IFS directories. In addition, you are also making backup copies of your journal receivers onto the tape media every day at 1 PM and again at 4 PM. You encounter a system disaster at 3 PM on a Friday. In order to recover your OnDemand system to the latest point in time prior to the disaster, you would first restore your system from the backup media which you made from the previous Saturday. After that, you would use your backup tapes to restore all the journal receivers that you saved since that Saturday, up to and including the last one that you have made on Friday at 1 PM. Following the restore operations, you would apply all the changes that occurred to your system such as the database tables and IFS objects up to the last journal receiver, using the OS/400 command

**APYJRNCHG.** On completion of this operation, your OnDemand system will be in a consistent state as at 1 PM on Friday. You only have to recapture or recreate the transactions that have occurred during the last 2 hours prior to the disaster. If, on the other hand you have not saved your journal receivers regularly or you used a backup strategy that only saved your data every evening, you would have lost all the changes that have occurred on your system since the last time (in this instance, last night) that you have saved your database tables and IFS objects.

From the above example, you can see that by making use of journaling, you can greatly reduce the amount of information that would potentially be lost in the event of a disaster.

For more information about backup and recovery concepts, refer to *iSeries Backup and Recovery Guide*, SC41-5304. You can find additional information about journaling and journal performance in the redbook *Striving for Optimal Journal Performance on DB2 Universal Database for iSeries*, SG24-6286.

## 9.3 Remote journaling

The *remote journal* function is a feature of OS/400 that offers you a reliable and fast method to transfer your journal entries to one or more remote iSeries servers. With remote journaling, you can establish and associate journals and journal receivers on a target system with specific journals and journal receivers on a source system. The original data remains on the source system and all changes to your database are still made on the source system. Once the remote journal function is activated, the source system continuously replicates journal entries to the target system.

The remote journal function is not a separate product or feature. It is part of the base OS/400 system, implemented at the licensed internal code layer. Some of the benefits of the remote journal function include:

- ▶ Lower CPU consumption on the source system. The processing required to harvest the journal entries is shifted from the source system to the target system.
- ▶ Fewer disk write operations and greater DASD efficiency on the source system. It eliminates the need to buffer copies of harvested journal entries to a temporary area before transmitting them from the source system.
- ▶ Greater replication performance and transmission efficiency when sending journal entries to the target system in real-time.
- ▶ If the remote journal synchronous mode is used, the journal entries are guaranteed to be in the main storage on the target system prior to the control being returned to the application on the source system.



- It allows you to transfer the backup operations of your journal receivers to the target system, thereby further reducing resource utilization on the source system.

Figure 9-3 shows a high level overview of the remote journaling function.

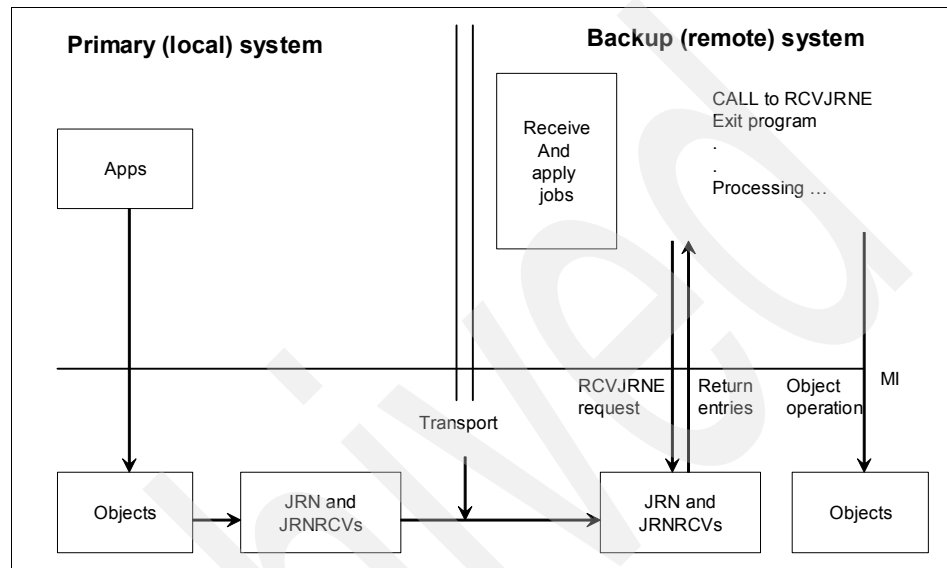


Figure 9-3 Remote journaling

All changes to your data continue to occur on the source system. Initially, you will need to create a replica of your data on the target system prior to establishing remote journaling. Afterwards, you would typically make use of application programs to access the journal receivers on the target and to replicate all the changes that have occurred on the source database, thereby keeping the source system and target system synchronized. In the event of a disaster occurring on your source system, you can continue normal business operations by simply switching your users over to your target system with minimal disruption to your business processes.

More information about remote journaling and performance considerations is available in the redbook *Striving for Optimal Journal Performance on DB2 Universal Database for iSeries*, SG24-6286.

## 9.4 High Availability Business Partner solutions

Currently, only certain OS/400 objects can be journaled. Depending on your business requirements, your high availability demands might well extend beyond those objects that can currently be journaled. This is where the High Availability Business Partner (HA BP) solutions come into play.

These solutions typically make use of standard OS/400 functionality, coupled with application programs to replicate changes that occur on your system to one or more target systems. In addition to replicating changes that are typically supported by the standard OS/400 journal functionality, they can also ensure that other objects, such as user profiles, job descriptions, spooled files, and output queues are replicated and synchronized on the source and target systems. Monitors are typically included in these solutions that enable you to check, verify and manage the replication and synchronization between your source and target systems.

Most HA BP solutions employ the journal asynchronous approach, where the changes on your source system are written asynchronously to the target systems. This ensures the lowest possible overhead and greater performance throughput on the source system. In addition, many HA BP solutions also offer a filtering facility which allows you to selectively replicate only certain journal changes from your source system to the target systems.

The dependence of your business on your system availability should dictate the extent of your high availability solution. HA BP solutions cost money and you might require additional infrastructure equipment, such as high speed links between your source and target systems in order to support the replication process at an acceptable level. We suggest that you consult and discuss your requirements with a HA BP before making a decision to implement a specific solution.

Your IBM representative will be able to advise you regarding HA BP solution providers in your area. For more information about HA BP solutions, refer to the redbook *Striving for Optimal Journal Performance on DB2 Universal Database for iSeries*, SG24-6286.

## Case studies for OnDemand iSeries

There are a number of factors that influence organizations in deciding on the best strategy to secure their OnDemand system and data. Some of these factors include:

- ▶ The extent to which the day-to-day operational activities of the organization is dependent upon on the OnDemand system
- ▶ Whether the OnDemand system and archived data is replicated onto a backup server
- ▶ The use of a High Availability Business Partner (HA BP) software solution
- ▶ Whether the OnDemand data is archived to optical disks
- ▶ The existence of a backup optical library server

In this chapter, we introduce you to a few techniques which some real-life customers employ to ensure recoverability and high availability of their OnDemand system and data. Not all of these methods and techniques will be applicable to your organization, but it will assist you in formulating a strategy to suit your particular requirements.

## 10.1 Scenario 1

In this example, the customer's OnDemand system resides on a single iSeries server and the migration policies specify that all archived data are permanently stored on the iSeries internal disks. The customer's backup and recovery strategy specifies that in the event of a system failure, the OnDemand system will be recovered using backup tape media.

To support this strategy, the OnDemand system is saved onto tape media at the end of each working day, using the **SAVLIB** OS/400 command to save the QUSRRDARS library and the OnDemand instance libraries. In addition, the entire contents of the IFS directory */QIBM/UserData/OnDemand/* is saved using the **SAV** OS/400 command. This ensures that the OnDemand system configuration data, as well as the archived data residing in cache are also saved. The backup tape media is then stored at a remote secure location.

In the event of a system failure, the QUSRRDARS and OnDemand instance libraries will be restored from the backup media using the **RSTLIB** OS/400 command. Similarly, the contents of the IFS directory */QIBM/UserData/OnDemand/* will be restored from the backup media, using the **RST** OS/400 command.

The customer further performs a full system save, using **Option 21** on the OS/400 **SAVE** menu, at the end of each working week.

## 10.2 Scenario 2

The scenario described here is often used by customers who employ a backup iSeries server as part of their disaster recovery and high availability strategy. The backup server remains in active stand-by mode and in the event of the primary server becoming inaccessible, the backup server would assume the role of the primary server. However, the use of an HA BP software solution does not form part of the high availability strategy. Any configuration changes to the OnDemand system on the primary server is manually replicated on the backup server.

As a once-off exercise, the entire OnDemand system, including all archived data residing on the iSeries internal disks, is replicated onto the backup server. This is typically done using a backup and restore operation to create a point-in-time replica of the primary OnDemand system on the backup server.

Next, a remote output queue is configured on the primary server which points to an output queue the backup server. The OnDemand system on the primary server is then configured to use this remote output queue as the processed queue for successfully archived reports. On the backup server, a **STRMONOND** job is

scheduled to archive reports that are placed on this output queue by the primary server. This ensures that reports that had been archived successfully on the primary server are also archived on the backup server, thereby keeping the two OnDemand systems synchronized.

Customers who employ this method may also choose to configure an optical Library Server on both the primary server and the backup server. By scheduling the Archive Storage Management (ASM) job on both servers, you can ensure that data that had been migrated to optical disks on the primary server are also migrated to optical disks on the backup server, thereby creating backup copies of your optical library data as well.

**Note:** If you have one optical Library Server that is shared by both the primary and the backup iSeries servers, then you should ensure that the Archive Storage Management (ASM) job is not scheduled to run on the backup server, as this would result in duplication of the same archived data on your optical disks.

## 10.3 Scenario 3

The following is an example of what customers do to secure their archived data that had already been written to optical media.

The technique involves using the **DUPOPT OS/400** command to make a backup copy of all the data contained on an optical volume. At regular intervals, for example at the end of each working day, a designated rewritable optical volume is added to the optical Library Server and is initialized as a primary volume with a unique volume ID. The **DUPOPT** command is then used to replicate the data that had been written to the source volume during that working day to the newly added target volume. Afterwards, the target volume is removed from the optical Library Server and stored at a remote secure location.

Customers who employ this technique typically make use of three or more target volumes and write data to these target volumes in a round-robin fashion. The following shows a simplified example of how this is achieved and the chronological order in which the volumes are written:

### **Day #1**

Perform the following steps:

1. Retrieve optical volume ONDV01B1 from the remote secure location.
2. Initialize ONDV01B1, for example **INZOPT VOL(ONDV01B1) DEV(OPTLIB1) CHECK(\*NO) TEXT('Backup Volume ONDV01B1') TYPE(\*PRIMARY)**.

3. Use the OS/400 command **DOPUOPT** to create a backup copy of the source volume, for example **DUPOPT FROMVOL(ONDV01A1) TOVOL(ONDV01B1)**.
4. Return the target volume ONDV01B1 to the remote secure location.

### ***Day #2***

Perform the following steps:

1. Retrieve optical volume ONDV01B2 from the remote secure location.
2. Initialize ONDV01B2, for example **INZOPT VOL(ONDV01B2) DEV(OPTLIB1) CHECK(\*NO) TEXT('Backup Volume ONDV01B2') TYPE(\*PRIMARY)**.
3. Use the OS/400 command **DOPUOPT** to create a backup copy of the source volume, for example **DUPOPT FROMVOL(ONDV01A1) TOVOL(ONDV01B2)**.
4. Return the target volume ONDV01B2 to the remote secure location.

### ***Day #3***

Perform the following steps:

1. Retrieve optical volume ONDV01B3 from the remote secure location.
2. Initialize ONDV01B3, for example **INZOPT VOL(ONDV01B3) DEV(OPTLIB1) CHECK(\*NO) TEXT('Backup Volume ONDV01B3') TYPE(\*PRIMARY)**.
3. Use the OS/400 command **DOPUOPT** to create a backup copy of the source volume, for example **DUPOPT FROMVOL(ONDV01A1) TOVOL(ONDV01B3)**.
4. Return the target volume ONDV01B3 to the remote secure location.

### ***Day #4***

Perform the following steps:

1. Retrieve optical volume ONDV01B1 from the remote secure location.
2. Initialize ONDV01B1, for example **INZOPT VOL(ONDV01B1) DEV(OPTLIB1) CHECK(\*NO) TEXT('Backup Volume ONDV01B1') TYPE(\*PRIMARY)**.
3. Use the OS/400 command **DOPUOPT** to create a backup copy of the source volume, for example **DUPOPT FROMVOL(ONDV01A1) TOVOL(ONDV01B1)**.
4. Return the target volume ONDV01B1 to the remote secure location.

In the above example, data had only been archived to the optical volume ONDV01A1 each day. However, in real life, data may well have been archived to more than one optical volume on any particular day. In such cases, you would have to repeat the process shown in the above example for each optical volume to which data had been written during that particular day.

The target optical volumes have to be in a rewritable format (REWT) and should have a storage capacity equal to or greater than the source volume. The source volumes may however be REWT or Write Once Read Many (WORM) format.

When both sides of a source disk is full or you have manually set the indicator to mark the volumes as full, you can use the **DUPOPT** command to make a final copy of both sides of the source disk to a designated target disk and then store the target disk at a secure remote location for an indefinite period of time. You should thereafter no longer use this target disk for future backup purposes.

In the event of an optical volume being lost or damaged, you would retrieve the duplicate volume from its secure location, add it to the optical Library Server and rename it to match the name of the lost or damaged optical volume. There is no need to convert the backup volume to a primary volume as it had not been initialized as a backup volume.

**Important:** If you decide to leave the target volumes in the optical Library Server, you must take special care not to include the target volume IDs in the list of optical volumes which you allocate to your OnDemand system. Otherwise, that target volume will be visible to the OnDemand system and data will be archived to these volumes when you run ASM.

## 10.4 Scenario 4

Customers who have an optical Library Server containing only a single drive cannot make use of the method described in Scenario 3 above, as the **DUPOPT** command requires more than one drive installed in the optical Library Server.

To secure their OnDemand archived data, these customers often configure their migration policies to specify that data written to the optical volumes should also be saved to a designated backup tape device. Figure 10-1 on page 228 shows how to specify this using the iSeries Navigator interface.

Migration Policy Properties - Rdr400cmond

Policy name: POLICY4

Description: Second Migration Policy

☒ Enable aggregation

Maximum size: 1000 kilobytes

☒ Close aggregate only when maximum size reached

☐ Close aggregate after specified time period

Time period: 0 days

☒ Tape backup requested

Media type: QIC50GB

Instance: ONDENU

Storage levels in this policy

| Level | Media     | Days   | Primary Group | Backup Group | Description     |
|-------|-----------|--------|---------------|--------------|-----------------|
| DISK  | Disk Pool | 30     | ASP01         |              | Disk Pool       |
| OPT   | Optical   | 999999 | OPTGRP01      |              | Optical Library |

Add Before...  
Add After...  
Change...  
Remove

OK Cancel Help

Figure 10-1 Migration policy tape backup option

When this option is specified, ASM will make a backup copy of the archived data to the specified tape device before moving the archived data to the first storage level defined in the migration policy.

**Note:** You must use the iSeries Navigator interface to define the tape device and the tape volumes that are to be used for the backup operations.

In the event of an optical volume being lost or damaged, the Tape Volume Recovery (TVR) process is used to locate and restore the data from the backup tape media. During this process, the QPRLCASMT1 report is generated which lists all the objects which were recovered successfully, as well as those which could not be recovered.

The following shows an example of how to initiate the TVR process:

```
CALL QRDARS/QLRCASMTVR PARM(<instance name> <volume to recover>)
```

Run the following to recover all objects for the QUSROND OnDemand instance that were located in the ASP01 disk pool:



```
CALL PGM(QRDARS/QRLCASMTVR) PARM(QUSROND *ASP01)
```

TVR considerations:

- ▶ You should run ASM immediately after successful completion of the TVR process so that the recovered objects are again archived to your archive media.
- ▶ Only primary optical, disk and tape volumes can be recovered.
- ▶ Your migration policies must have the tape backup option specified.
- ▶ The following OnDemand files must be available and up-to-date:
  - QARLCOIT (Object Inventory Table)
  - QARLCOAT (Object Archive Table)
  - QARLCTVET (Tape Volume Extended Table)
  - QARLCTDEV (Tape Device Table)
- ▶ TVR will overwrite any existing object with the recovered backup copy.
- ▶ After completion of the TVR process, the recovered objects are located in the /qibm/userdata/ondemand/<instance name>/ASMAGGREGATION directory if you enabled aggregation in your migration policy.
- ▶ If you have not specified the aggregation option in your migration policy, the recovered objects are located in the /qibm/userdata/ondemand/<instance name>/ASMREQUEST directory after completion of the TVR process.

Some customers also save the contents of their optical volumes to a backup tape device on a regular basis by using **SAV OS/400** command. In the event of an optical volume being lost or damaged, the saved archived data can be recreated from the backup tape to a new optical volume using the **RST OS/400** command.

## 10.5 Scenario 5

Some customers prefer to use the **CPYOPT OS/400** command to make duplicate copies of archived data that had been written to their optical volumes.

The advantage of this approach is that you can specify that only data that had been added or changed on the source volume should be copied to the target volume. The disadvantage of this technique is, however, that it could be a slow and lengthy process. In order to determine which data had been added or changed, a file level comparison is done between the data on the source volume and target volume before writing the data to the target volume.

It is therefore often less time consuming to initialize the target volume and use of the **DUPOPT OS/400** command to recreate the entire volume every time or to specify the **SLTFILE(\*ALL)** parameter if you use the **CPYOPT** command.

## 10.6 Scenario 6

Another technique used by customers to secure their optical volumes is to specify a backup optical storage group in their migration policies. Figure 10-2 shows how to create a backup optical storage group using the iSeries Navigator interface.

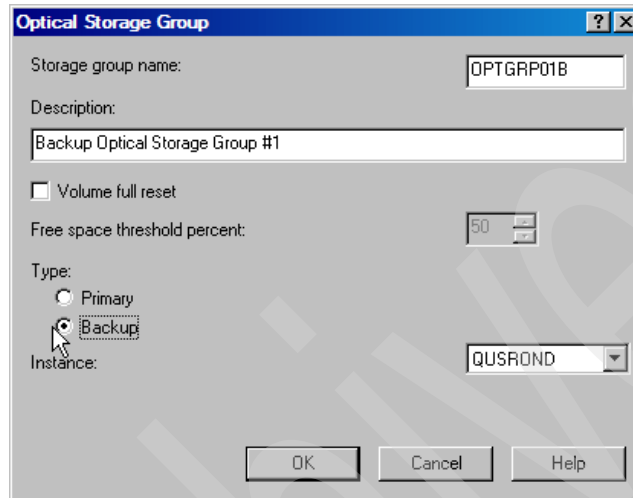


Figure 10-2 Create backup optical storage group

The migration policy is then configured to use this backup optical storage group. Figure 10-3 on page 231 shows how to specify the use of a backup optical storage group at the migration policy level in the iSeries Navigator interface.

**Policy Level Properties - Tr1new**

Policy name: MIGRATION POLICY #2

Level identifier: 0002

☐ Disabled

Description: Migrate to Optical Media

Media:

- ☒ Optical
- ☐ Tape
- ☐ Disk pool
- ☐ Expire

Duration at this level:

- ☒ No maximum
- ☐ 0 days

Primary storage group: OPTGRP01

☒ Create backup copy

Backup storage group: OPTGRP01B

☐ Stage to disk if retrieved from tape

Duration on disk: 0 days

OK Cancel Help

*Figure 10-3 Specify backup optical storage group*

When you specify this option, a duplicate copy of your archived data is created on the backup optical volumes associated with the specified backup optical storage group when the archived data is moved to this migration policy level.

If your primary storage group and your backup storage group are defined on a single optical Library Server, then both the primary volumes and the backup volumes will be contained in the same physical optical Library Server. You will therefore still need to make arrangements for securing your data at a secure remote location.

**Note:** The primary optical volumes and the backup optical volumes might not be completely identical. For example, bad sectors on the primary volume are not recreated on the backup volume and bad sectors on a backup volume cannot be written to. It is therefore possible that data from one primary volume might be located on two backup volumes. This does not, however, affect your ability to recover your data from the backup volumes.

## 10.7 Scenario 7

Often customers make use of HA BP software solutions to secure their OnDemand system and archived data.

These customers typically configure the high availability software to ensure near real-time replication of all changes to the QUSRRDARS and OnDemand instance libraries, as well as the contents of the `/QIBM/UserData/OnDemand/` IFS directory to the backup server. In addition, data placed on selected output queues on the primary server are also replicated to the backup server. In the event of the primary server becoming unavailable, the backup server will assume the role of the primary server and the OnDemand system and archived data will be readily available.

Generally, HA BP software solutions require that all data changes only occur on the source system. You should therefore ensure that you do not schedule any ASM jobs or **STRMONOND** jobs on the backup server. These jobs will cause data changes to occur on your backup server which will trigger out-of-sync conditions in your HA BP software solution. This could negatively affect your ability to switch over to your backup server in the event of a disaster.

In addition, these customers also employ suitable methods, similar to those described in the scenarios mentioned above, to secure their archived data contained on optical volumes.

## 10.8 Conclusion

As you can see from the above examples, customers use different methods to secure their data, depending on their specific requirements and infrastructures. Your unique requirements may very well require a combination of some of the scenarios described above.

There are a number of factors that you should consider when embarking on a strategy to secure your OnDemand system and archived data. Some of these include the length of time you retain your data in cache through the settings specified in your migration policies, the size of your OnDemand archives and the capacity of your optical cartridges. But, most importantly, your strategy must complement and align with your organization's dependence on your OnDemand system and archived data.



# Part 4

## z/OS

In Part 4, we focus on OnDemand for z/OS and OS/390. We provide an overview of OnDemand for z/OS, discuss the backup and recovery strategies and options, and describe different options for performing backup and recovery of OnDemand for z/OS system. In addition, we provide options and strategies to achieve high availability and business continuity to assist you in meeting your business requirements and your operating environment.



## OnDemand overview for z/OS

In this chapter we provide an overview of the IBM DB2 Content Manager OnDemand (OnDemand) for z/OS. The concepts covered include both OnDemand and ODWEK (OnDemand Web Enablement Kit).

We describe how OnDemand manages reports and index data, and how the OnDemand Library Server and Object Server works together to index, load, and retrieve documents.

The following topics are covered:

- ▶ System overview
- ▶ OnDemand terminology and concept
- ▶ OnDemand Web Enablement Kit (ODWEK)

## 11.1 Introduction

OnDemand is the leading offering for enterprise report management and electronic statement presentment solutions. It provides high volume capture of computer output, advanced out-of-the-box client applications for both desktops and standard Web browsers, and it enables automated storage management with advanced search and report-mining capabilities.

The OnDemand application allows an enterprise to capture, organize and store any printed output (such as reports, statements, or invoices) as well as e-mails and image documents. It is a computer output content management solution for leading Enterprise Report Management (ERP) and Customer Relationship Management (CRM) applications, and provides a platform for implementing leading electronic bill presentment and payment solutions.

OnDemand for z/OS supports structured and unstructured data types.

OnDemand provides users the ability to view documents, print, send and FAX copies of documents, and attach electronic notes to documents.

### 11.1.1 OnDemand features and functionalities

OnDemand for z/OS provides the following features and functionalities:

- ▶ Integrates data created by application programs into an online, electronic information archive and retrieval system.
- ▶ Provides the controlled and reliable access to all of an organization's reports.
- ▶ Retrieves data that is needed when it is needed.
- ▶ Provides a standard, intuitive client with features such as thumbnails, bookmarks, notes, and shortcuts.
- ▶ Provides indexing flexibility.
- ▶ Provides multiple levels of security.
- ▶ Supports OAM, VSAM, and TSM for archive data storage.
- ▶ Supports both the HFS and the zFS for cache data storage.
- ▶ Supports of complex print streams (including Xerox and PCL).
- ▶ Provides host based APIs that allow for further customization (including security, indexing and data presentation).
- ▶ Provides client based APIs (Java) that allow for the creation of custom clients and applications.

Using OnDemand, you have the following advantages:



- ▶ Easily locating data without specifying the exact report.
- ▶ Retrieving the pages of the report that are needed without processing the entire report.
- ▶ Viewing selected data from within a report.

This means that OnDemand can help customers quickly retrieve the specific page of a report that is need to provide fast customer service.

## 11.2 System overview

OnDemand for z/OS consists of client programs and server programs that communicate over a network running the TCP/IP protocol, a database manager that maintains objects, index data, and server control information, and storage managers that maintain documents on various types of storage devices.

The OnDemand server environment (known as an OnDemand Instance) includes a Library Server and one or more Object Servers residing on one or more systems connected to a TCP/IP network.

Figure 11-1 on page 238 highlights the OnDemand for z/OS logical model.

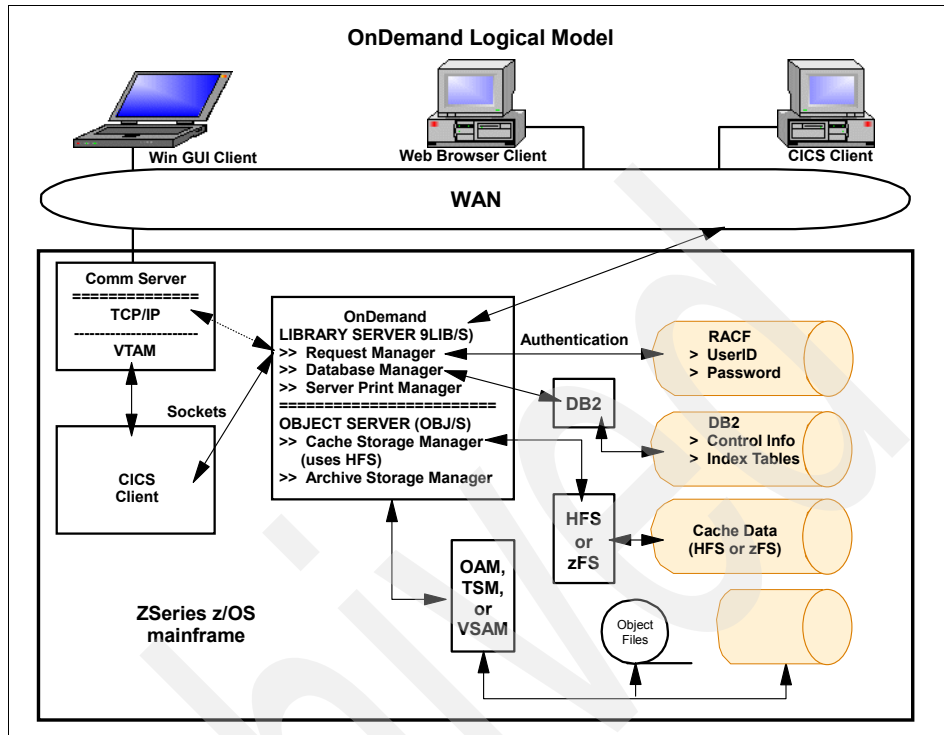


Figure 11-1 OnDemand for z/OS logical model

## 11.2.1 Library Server

The OnDemand *Library Server* is the central component of the OnDemand system. It uses a relational database manager to manage objects stored on one or more Object Servers. The OnDemand Library Server maintains index information, controls access to the stored objects and ensures data integrity. The objects it manages include users, user groups, storage sets, storage nodes, printers, applications, application groups, and folders.

The control files in the Library Server are used to initialize and operate the server.

The Library Server processes the logins from the clients, handles the requests from the clients, and maintains the database. The Library Server also routes requests to the appropriate Object Server to store, retrieve, and delete data objects.

An OnDemand instance is a logical server environment made up of a database, a Library Server, and one or more Object Servers. An *instance* is defined in the

ars.ini file by naming the instance, identifying the name of the database used by the instance, and identifying the Library Server on which the database will be maintained. Each OnDemand instance (the server, database and cache storage) has the following characteristics:

- ▶ Has its own folders, application groups, applications, users, groups, storage sets, and printers.
- ▶ Must run in a single code page.
- ▶ Has different security (for users, groups, folder and application group permissions).
- ▶ Must have its name specified in commands if it is not the default instance.
- ▶ Has its own system log.

## 11.2.2 Object Server

An *Object Server* maintains documents in cache storage nodes and/or archive storage nodes. A cache storage node can store documents as a collection of files in the Hierarchical File System (HFS) or the zFS (z/OS File System). An archive storage node can store documents in Tivoli Storage Manager (TSM), Virtual Storage Access Method (VSAM) files or Object Access Method (OAM) objects. An Object Server loads data, retrieves documents, and expires documents. The major functions that run on an Object Server are the cache storage manager, the OnDemand data loading and maintenance programs, and the archive storage manager.

OnDemand supports storing data on and retrieving data from more than one Object Server. In a distributed environment, users submit queries to the Library Server and OnDemand retrieves documents from the Object Server on which the data is stored. You can load reports on any of the Object Servers that are part of the system.

## 11.2.3 Server configurations

The two basic OnDemand configurations are the *standard* Library Server and Object Server configuration and the *distributed* Library and Object Server configuration.

### **Standard Library Server and Object Server configuration**

The standard Library Server and Object Server configuration includes the cache storage manager, the archive storage manager and the programs that are required to index reports and load data on the system.

Figure 11-2 on page 240 shows a standard Library Server and Object Server configuration.

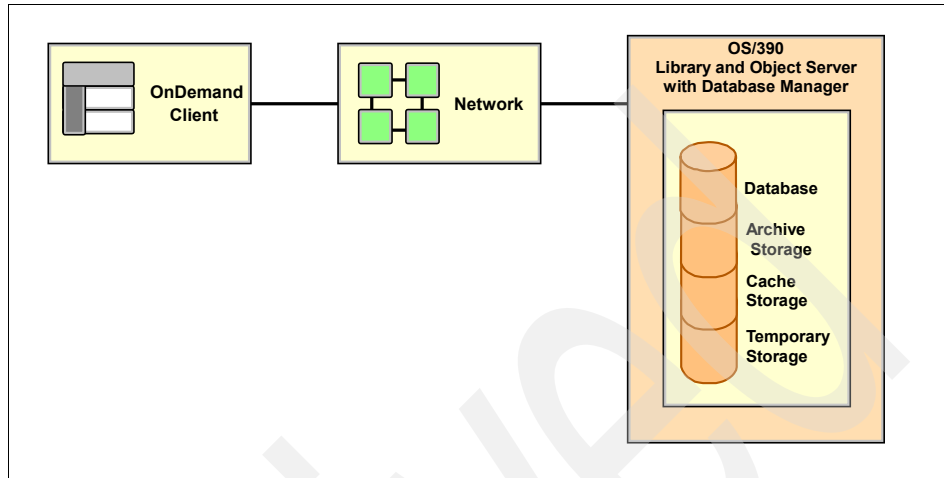


Figure 11-2 Standard Library Server and Object Server configuration

You can stage reports on temporary storage volumes for the data indexing and loading programs. This configuration is ideal if you need to run OnDemand on a single z/OS or OS/390 system.

The software required for this standard system configuration include:

- ▶ OnDemand base: Base OnDemand functions, such as enhanced ACIF (the OS/390 indexer is built in to the OnDemand product).
- ▶ Database manager (DB2): Database engine and administration.
- ▶ OnDemand client: Windows client program.

### Distributed Library Server and Object Server configuration

OnDemand supports storing data on and retrieving data from one or more Object Servers. In a distributed environment, users submit queries to the Library Server, the OnDemand software determines the location of the documents, retrieves them from the appropriate Object Server on which the data is stored and returns the documents to the users.

Figure 11-3 on page 241 shows a distributed Library Server and Object Server configuration.

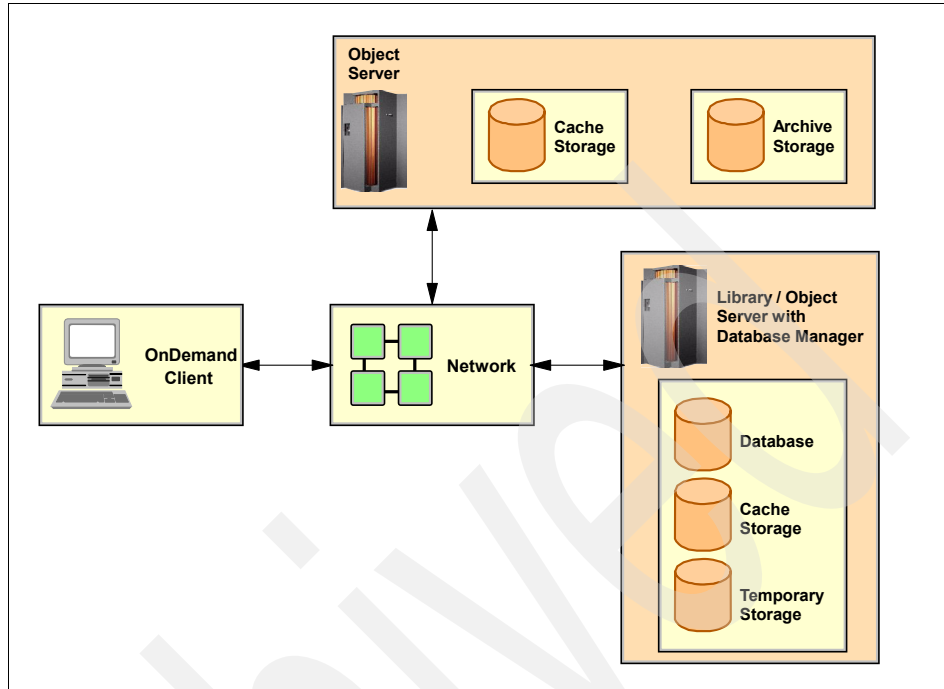


Figure 11-3 Distributed Library Server and Object Server configuration

In a distributed system configuration, you can load reports on any of the Object Servers that is part of the system. The index data is always stored on the Library Server. You can stage reports on the temporary storage volumes for the data indexing and loading programs. This configuration is ideal if you need to distribute the loading and accessing of reports over more than one LPAR. The Library Server and the Object Servers can reside on LPARs in a single z/OS or OS/390 system, or on separate systems in different physical locations. A distributed Object Server is ideal if you want the archive storage to be part of the system off of the Library Server.

The software required for the Library Server in a distributed system configuration include:

- ▶ OnDemand base: Base OnDemand functions, such as enhanced ACIF.
- ▶ Database manager (DB2): Database engine and administration.
- ▶ OnDemand client: Windows client program.

The software required for the Object Server in a distributed system configuration include:

- ▶ OnDemand base: Base OnDemand functions, such as enhanced ACIF.

## 11.3 OnDemand terminology and concept

The terms application, application group, and folder represent how OnDemand stores, manages, retrieves, views, and prints reports and index data. When defining a new report or type of data to OnDemand, an administrator must create an application and assign the application to an application group. (If an application group does not exist, the administrator must create one first.) In order for users to search for and to retrieve documents, an administrator must create or update a folder to use the application group and application for searching.

The stored report (and its resources) are stored in the specified Object Server in storage nodes. The storage nodes may be cache storage nodes (for short term data storage) or archive storage nodes (for long term data storage).

### 11.3.1 Applications

An OnDemand *application* describes the physical characteristics of a report, processing instructions for the indexing and data loading programs, and information about how OnDemand displays and prints pages of a report. You can specify default settings for viewing and printing pages of a report at the OnDemand application level. Typically you define an application for each different report that you plan to load into the system. The parameters defined specify information to be used by the indexing and data loading programs. These definitions specify the techniques that OnDemand uses to compress the report file, the parameter used to index the data, and information that OnDemand uses to process index data before loading index records into the database. OnDemand uses the indexing parameters, options, and data values that you specify to locate index data in and extract index data from the report.

### 11.3.2 Application groups

An *application group* is a collection of one or more applications that have the same or similar index fields and storage characteristics. The application group is the object that OnDemand uses to maintain the reports that you load into the system. The application group holds index data for reports, documents, management information, permissions for the groups and users authorized to access application group. When you define an application group, you specify the name and type of the database fields that hold the index data extracted from the reports that are loaded into the application group. You specify whether a database field is used to index or filter data, and specify other characteristics of the fields. We recommend defining all potential database fields as index data if these fields will be frequently used in searches.

When you define an application group, OnDemand creates an application group table structure in the DB2 database, with a column for each database field that you defined. When you load a report into the application group, OnDemand inserts rows into an application group DB2 table for each indexed item found in the report. An indexed item can be a logical item, such as a policy or statement, or a group of pages, depending on how the report is organized and how you decide to index the report. Users search for reports using one or more of the fields that you defined for the application group.

### 11.3.3 Folders

A *folder* provides users the means to search for and retrieve related reports stored on a system. Users open folders, construct queries, and retrieve reports from application groups. Users do not need to know or understand the concept of application groups.

When you create a folder, you define the search and display fields that appear when users open the folder. You map the folder fields to database fields in the application groups referenced by the folder. The *database fields* contain index values extracted from the reports that are loaded into the application groups.

For example, a folder search field, Customer Account Number, could be mapped to the ACCOUNT application group database field. OnDemand creates database records that include the index values for the ACCOUNT field when you load a report into the application group. When a user enters a query, OnDemand retrieves records from the database if the values of the ACCOUNT database field match the value that the user typed in the Customer Account Number search field.

Figure 11-4 on page 244 shows the relationship between applications, application groups, and folders.

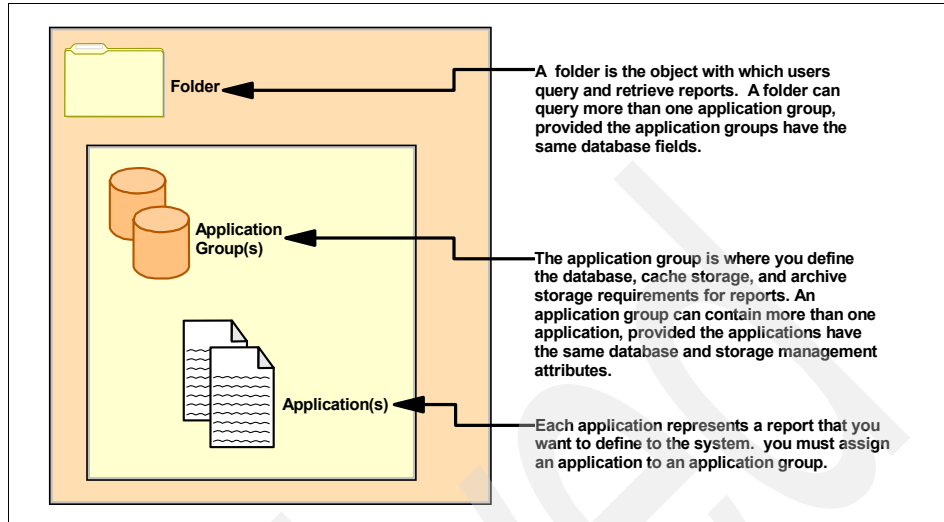


Figure 11-4 Application, application groups, and folders' relationship

### 11.3.4 Resources

If the report to be stored is AFP data, it may contain resources such as form definitions, page definitions, fonts, and overlays. OnDemand always stores a copy of the resources in cache storage, to provide fast retrieval when a user selects an item for viewing. The ARSLOAD program saves only one copy of a resource on the system, even if several reports use the same resource. When processing a resource group file, the ARSLOAD program checks the resource identifier to determine if the resource is already present on the system. If the storage node identifies a client node in TSM, OAM or VSAM, then the storage manager copies the resources to archive storage.

We recommend that you exclude fonts from the resource file. This reduces the number of bytes transmitted over the network when a document is retrieved for viewing.

### 11.3.5 Storage sets and storage nodes

A *storage set* is a named collection of storage nodes that support application groups with similar storage management characteristics. For example, a storage set can maintain data for several application groups that need to keep documents for the same length of time and store the data on the same type of media.

A *storage node* identifies an Object Server where the report data is stored. A storage node can specify cache storage, archive storage, or both. A storage set



can write data to one and only one storage node at a time (the active storage node).

If you configure your application groups to copy data to cache storage, then the storage manager copies the storage object to cache storage. The Cache Data for xx Days setting on the Storage Management page determines whether OnDemand copies documents to cache storage.

A storage node can identify a client node in TSM, OAM or VSAM. OnDemand uses archive storage to maintain storage objects for long-term storage and for backup copies of reports. The storage manager can copy the storage object to archive storage when the report is initially loaded into the system or at a later time, depending on how you configure your application groups. Most customers configure the system to copy report data to cache storage and archive storage at the same time.

### 11.3.6 Cache and archive storage

The report data may be stored in cache or archive storage nodes.

The primary purpose of *cache storage* is for short-term, high-speed retrieval of report data. Cache storage is located in either the HFS or in the safes. Reports are always stored in cache storage on the Object Server identified by the active storage node.

The primary purpose of *archive storage* is for the long-term storage and retrieval of report data. Archive storage may be located in TSM, OAM or VSAM. Each of these archive technologies utilize their own techniques for backup and restoration of data.

## 11.4 OnDemand Web Enablement Kit (ODWEK)

The OnDemand Web Enablement Kit (ODWEK) allows users to access data that is stored in an OnDemand server by using a Web browser. For example, from a Web browser, you can submit a search and ODWEK will retrieve the response from the OnDemand Library Server and display a Web page that contains a list of the documents that match your query. You can then select a document to view and ODWEK will retrieve the document from the OnDemand Object Server and send the document to the browser.

There are several components in ODWEK:

- ▶ OnDemand programming interface:
  - Common Gateway Interface (CGI)
  - Java servlet

- Java Application Programming Interface (API)
- ▶ Document viewing
  - OnDemand AFP Web Viewer
  - OnDemand Image Web Viewer
  - Line Data Java applet
  - AFP2HTML Java applet
  - AFP2PDF Transform

### 11.4.1 11.4.1 OnDemand programming interface

An instance of ODWEK (sometimes called an application) is ODWEK code that accesses data on an OnDemand server. An instance controls what can be done to the data, and manages the system resources that are assigned to it. Each instance is a complete environment. An instance has its own ASWWW.INI file and ODWEK programming interface, which other instances cannot access.

There are three ODWEK programming interfaces:

- ▶ Common Gateway Interface (CGI program)
- ▶ Java servlet
- ▶ Java Application Programming Interface (API)

The CGI program runs on a z/OS or OS/390 system that is running an HTTP server, such as the IBM HTTP Server.

The Java servlet runs on a z/OS or OS/390 system that is running a Java-enabled HTTP server (running as a Java application server), such as the IBM WebSphere Application Server. The servlet requires Java Version 1.2.2 or later. If you plan to use Java Version 1.3.1 to support the Java servlet, you must install Java Version 1.3.1 with Fix Pack 4 or later.

The Java Application Programming Interface (API) is a set of APIs that reproduce the CGI interface for Java developers. The Java APIs require Java Version 1.2.2 or later. The Java APIs do not require the HTTP server or an application server (such as WebSphere) to be installed.

These OnDemand programming interfaces use standard OnDemand protocols to access data stored in an OnDemand server. No additional code is needed on the OnDemand server to support ODWEK. You may use any one of the interfaces in your ODWEK application.

An instance may use only one programming interface. The programming interfaces are mutually exclusive. They cannot be used in the same instance at the same time. However, it is possible to run multiple instances of ODWEK on a single machine and have each instance using a different programming interface by configuring each instance to use a different port number. The most common

implementation of ODWEK is a single instance on a system. The single instance configuration is typically for developers or stand alone production computing, which involve a single application server instance operating independently of any other applications. Figure 11-5 illustrates a single ODWEK instance running on the mid-tier server using the java interface.

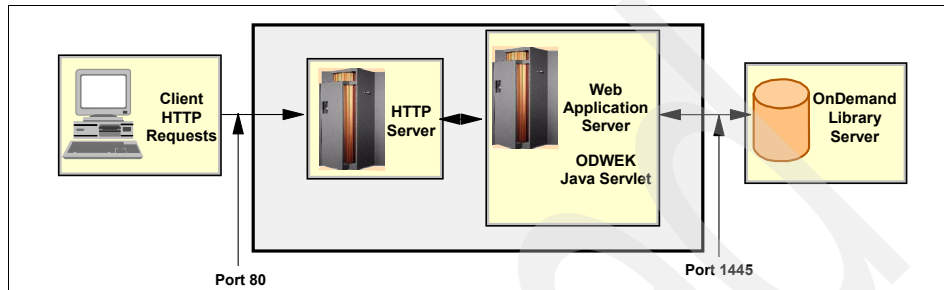


Figure 11-5 Single instance using the Java interface

Figure 11-6 on page 248 shows an example of three ODWEK instances running in the same mid-tier server, with each instance using one of the different programming interfaces. Each instance requires its own programming interface and ARSWWW.INI file, which specifies the unique port number over which communications between the programming interface and the OnDemand server take place. Each instance also requires its own storage and security. The multiple instance configuration is typically for customers that need to run one or more developer, testing or production applications on the same system. The instances operate independently of each other.

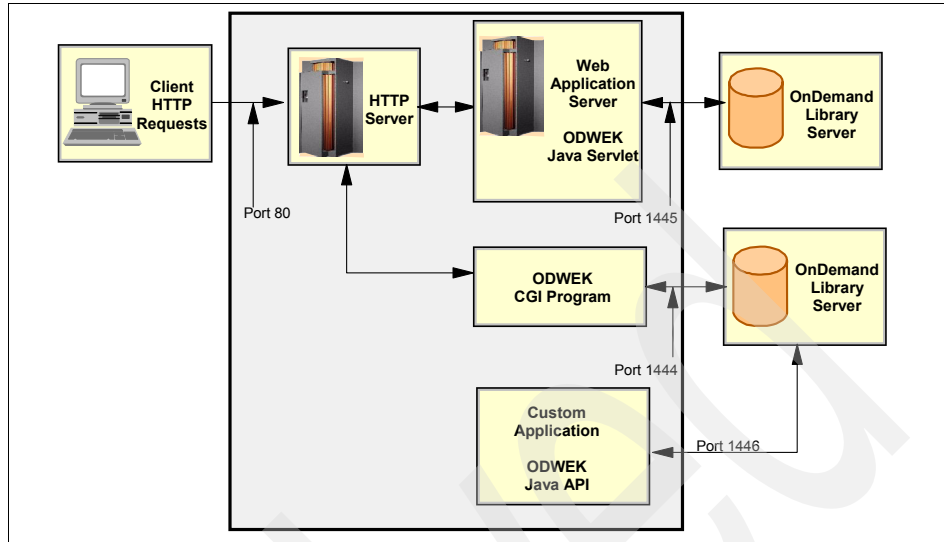


Figure 11-6 Three instance OWDEK topology

## 11.4.2 Document viewing

ODWEK provides the following viewers:

- ▶ OnDemand AFP Web Viewer
- ▶ OnDemand Image Web Viewer
- ▶ Line Data Java applet
- ▶ AFP2HTML Java applet
- ▶ AFP2PDF Transform

The *AFP Web Viewer* and the *Image Web Viewer* are software programs that extend the capabilities of a Web browser in a specific way.

The AFP Web Viewer lets users search, retrieve, view, navigate, and print AFP documents from a Web browser.

The *OnDemand Image Web Viewer* lets users search, retrieve, view, navigate, and print BMP, GIF, JPEG, PCX, and TIFF documents from a Web browser.

The viewers provide the capability to display documents in a browser. Each viewer adds a tool bar to the top of the display window, in addition to the browser's tool bar. The viewer tool bar provides controls that can help users work with documents. Each user who plans to use the Web viewers to view documents must install them on their local machines.

The *Line Data Java applet* allows users to view line data documents from a Web browser that are stored in OnDemand. The Line Data Java applet displays line data documents in the browser window and adds a tool bar to the top of the display window. The Line Data Java applet tool bar provides controls that can help users work with documents

The *AFP2HTML Java applet* allows users to view the output generated by the IBM AFP2WEB Transform service offering. The AFP2WEB Transform converts AFP documents and resources into HTML documents that can be displayed with the Java AFP2HTML Java applet. After installing and configuring the AFP2WEB Transform, an administrator enables the use of the Java AFP2HTML Viewer by configuring the ARSWWW.INI file. The AFP2HTML Java applet provides a tool bar with controls that can help users work with documents, including controls for large objects.

One advantage of the applets is that users never have to install or upgrade software on their local desktops to use them, unlike the AFP Web Viewer and the Image Web Viewer, which must be installed on the local machines. Also, if a new version of the AFP Web Viewer or the Image Web Viewer comes out, you must distribute the updated software to all users. When using the applets and viewers that are provided by IBM, the documents that are retrieved from an OnDemand server remain compressed until reaching the viewer. The viewer uncompresses the documents and displays the pages in a Web browser window. If a document was stored in OnDemand as a large object, then the viewer retrieves and uncompresses segments of the document, as needed, when the user moves through pages of the document.

The *AFP2PDF Transform* allows for the conversion of AFP documents retrieved from OnDemand into PDF documents that can be viewed with the Adobe Acrobat viewer

For more information, refer to *IBM DB2 Content Manager OnDemand for z/OS: Web Enablement Kit Implementation Guide*, SC27-1376.



# Backup and recovery for OnDemand z/OS

In this chapter, we describe different options for creating backups of the components of OnDemand for z/OS. We outline the backup strategy fitting your functional requirements, and we show you how to restore your data without losing data or leaving the components in an inconsistent state.

The following topics are covered:

- ▶ Backup and recovery overview
- ▶ Library Server backup and recovery
- ▶ Object Server backup and recovery
- ▶ Object Access Method (OAM)
- ▶ Virtual Storage Access Method (VSAM)

## 12.1 Backup and recovery overview

IBM DB2 Content Manager OnDemand on z/OS and OS/390 is an inseparable component of your mission critical business. Losing the information that is contained in your OnDemand system could have a major impact on your business. Today's environment generates higher customer expectations and faster anticipated response times. A business cannot afford any degree of disruption to its environment.

It is only when a data loss occurs that you realize that what is there was the most expensive constituent of your business. While you may not be able to predict all possible disruptions, you can be prepared to prevent their impact on your business.

In this chapter, we discuss backup and recovery strategies for OnDemand on z/OS and OS/390. We describe different options of how to perform backup and recovery of the components of OnDemand. We also provide options and strategies on how to achieve high availability so as to assist you in meeting your business requirements in Chapter 13, "High availability for OnDemand z/OS in a SYSPLEX environment" on page 281.

If you have not read Chapter 1, "Basic concepts" on page 3, please read it to gain knowledge of the basic concepts for backup and recovery before continuing.

An OnDemand system consists of many different parts. There is at least one Library Server and one or more Object Servers. You need to plan for backup of and recovery of the following critical OnDemand components:

- ▶ Library Server
  - OnDemand software
  - OnDemand server information, created or modified during installation, configuration, and on-going operation of OnDemand
  - OnDemand database
  - Other configuration and product files
- ▶ Object Server
  - Stored reports
    - Cache storage
    - Archive storage



## 12.2 Library Server backup and recovery

In this section, we address four area of backup and recovery for the OnDemand Library Server: OnDemand software, OnDemand server information, OnDemand database, and other configuration and product files.

### 12.2.1 OnDemand software

Some OnDemand programs are stored in the HFS while others are stored in MVST<sup>™</sup> Libraries. Both are managed by SMS.

If a media failure or some other unforeseen event occurs, you may be required to restore the OnDemand software programs, database software, archive manager software, server print manager software, and other application and user-defined software that you use on the system.

You can usually use the original product media as a basis to restore the software programs. It is important that you store the original product media in a safe location. We recommend that you register OnDemand as part of your business recovery plan, store the original product media in the same place where you store the other programs and files that are vital to the operation of your systems, and follow the same procedures used for the storage and recovery of application software. This would include the storage and recovery of the latest software updates, including all the applied PTFs.

### 12.2.2 OnDemand server information

You need to back up OnDemand server information, that are created or modified during installation, configuration, and on-going operation of OnDemand.

When you installed and configured OnDemand, you specified information that customizes OnDemand to operate in your environment. If you periodically make changes to the system, including the database, archive storage manager, and server print manager, it is helpful to back up the control files on a regular basis, perhaps once a week. The OnDemand control and data files are contained in the HFS.

See the operating system and device publications for your server for details about backup and restore concepts and commands for HFS and SMS.

### 12.2.3 OnDemand database

OnDemand supports storing index data in tablespaces and the incremental backup of tablespaces. tablespaces enhance the management of index data and

provide improved performance, especially for database backups. An incremental tablespace backup completes much quicker than a full database backup, providing you with increased flexibility in scheduling the loading of reports. Incremental backup images also require less storage space than full database backups.

DB2 provides means for recovering data to its current state or to an earlier state. The units of data that can be recovered are tablespaces, indexes, index spaces, partitions, and data sets.

The elements that DB2 manages can be divided into two broad categories:

- ▶ Data structures, which are accessed under the user's direction and by which the user's data (and some system data) is organized
- ▶ System structures, which are controlled and accessed by DB2

### **DB2 objects and structure hierarchy**

A DB2 database consists of DB2 objects and is structured hierarchically as shown in Figure 12-1 on page 255. The figure provides an overview of DB2. It introduces a hierarchy of structure from the most to the least inclusive.

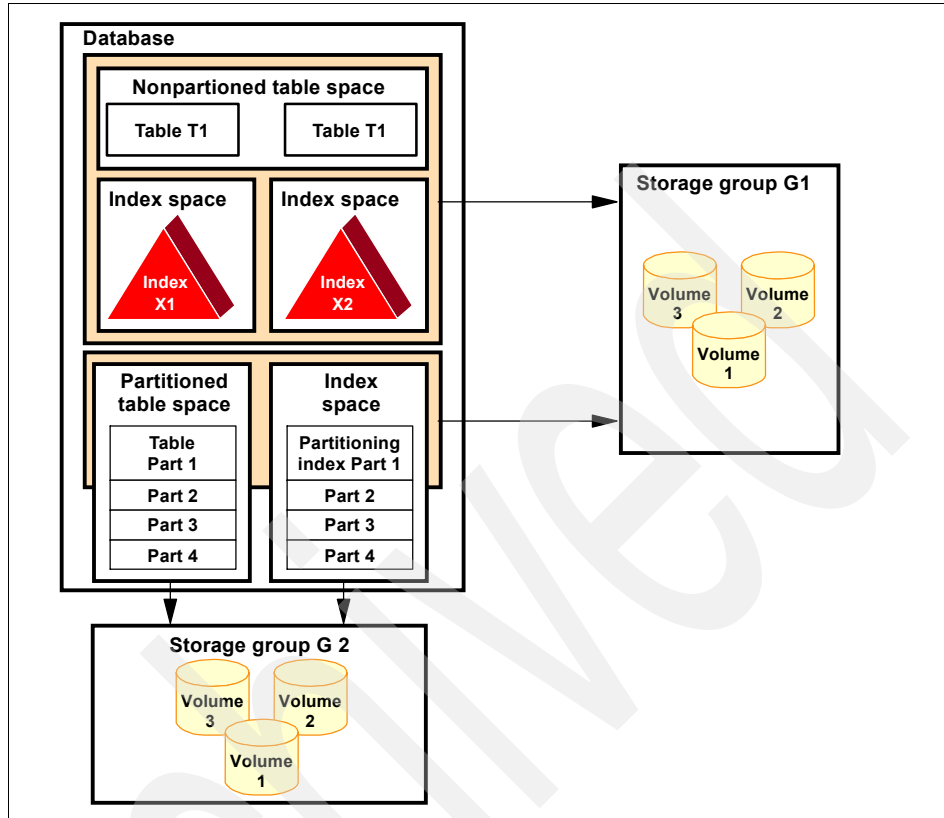


Figure 12-1 A hierarchy of DB2 structures

The basic DB2 objects as shown in Figure 12-1 include:

- ▶ Database
- ▶ Storage group
- ▶ Tablespace
- ▶ Tables
- ▶ Indexes

### **Database**

*Database* is a set of DB2 structures that include a collection of tables, their associated indexes, and the tablespaces in which they reside. A single database can contain all the data associated with one application or with a group of related applications. Collecting that data into one database allows you to start or stop access to all the data in one operation and grant authorization for access to all the data as a single unit. Assuming that you are authorized to do so, you can access data stored in different databases.

## ***Storage group***

*Storage group* is a set of volumes on disks that hold the data sets in which tables and indexes are actually stored. The description of a storage group names the group and identifies its volumes and the virtual storage access method (VSAM) catalog that records the data sets. The default storage group is created when you install DB2. All volumes of a given storage group must have the same device type. Parts of a single database can be stored in different storage groups.

## ***Tablespace***

*Tablespace* is a set of volumes on disks that hold the data sets in which tables and indexes are actually stored. A tablespace can consist of a number of VSAM linear data sets (LDSs). Tablespaces are divided into equal-sized units, called pages. One or more pages can be written to or read from disk in a single disk operation. You can specify page sizes (4k, 8k, 16k and 32k) for the data. The default page size is 4 KB.

When you create a tablespace, you can specify the database to which the tablespace belongs and the storage group it uses. If you do not specify the database and storage group, DB2 assigns the tablespace to the default database and the default storage group.

There are three types of tablespaces:

- ▶ **Partitioned:** Divides the available space into separate units of storage called partitions. Each partition contains one data set of one table. You assign the number of partitions (from 1 to 254) and you can assign partitions independently to different storage groups.
- ▶ **Segmented:** Divides the available space into groups of pages called segments. Each segment has the same size. A segment contains rows from only one table.
- ▶ **Simple:** Can contain more than one table. The rows of different tables are not kept separate (unlike segmented tablespaces).

## ***Tables***

When you create a *table* in DB2, you define an ordered set of columns. All data in a DB2 database is presented in tables. A table that holds persistent user data is a base table. A table that stores data temporarily is a global temporary table.

## ***Indexes***

An *index* is an ordered set of pointers to the data in a DB2 table. The index is stored separately from the table.

Each index is based on the values of data in one or more columns of a table. After you create an index, DB2 maintains the index. You can perform necessary

maintenance such as reorganizing it or recovering the index. Indexes take up physical storage in index spaces. Each index occupies its own index space.

## Planning for DB2 backup and recovery

Backup and recovery procedures is critical to avoid costly and time-consuming data losses. OnDemand provides the ARSDB program to create the backup images of the OnDemand database. This program is normally not used in a z/OS environment.

In a z/OS environment, typically, the database backups are the responsibility of the database administrator and are performed based on system workload, availability to users (24x7x365) and data volumes.

You should develop procedures to:

- ▶ Create a point of consistency.

This is the last point in time that the database will be recovered to in the case of a DB2 failure. Typically, this is accomplished through the use of incremental tablespace backups, with either online or offline full backups being run depending on system availability (where availability refers to backup availability, for example, when the system is not being accessed by users.)

- ▶ Restore system and data objects to a point of consistency.

In the event of failure, restoration is to the last point of consistency. It is important to keep both the Library Server database and the Object Server reports in synchronization. Any metadata that is lost requires that the reports that were pointed to through that metadata be reloaded into the system and new metadata be inserted into the DB2 tables.

- ▶ Recover from out-of-space conditions.

These are temporary conditions that prevent the loading of new data. They are resolved by assigning additional DASD to the tablespace and then restarting the process that failed due to the out of space condition.

- ▶ Recover from a hardware or power failure.

These are data center wide procedures. In the case of a Parallel Sysplex® spanning over two or more data centers, this should be transparent to the users.

- ▶ Recover from an MVS component failure.

In the case of the Parallel Sysplex, this should be transparent to the users. Recovery is normally achieved by restarting the failed component.

In addition, you should consider a procedure for off-site recovery in case of a disaster. This could include an establishment of two or more data centers for a Parallel Sysplex environment, or it can be as simple as storing copies of all data

and metadata off-site and having a support agreement with a data center provider that allows you to operate off of the provider's data center in the case of a major disaster.

To improve recovery capability in the event of a disk failure, it is advisable to use dual active logging and to place the copies of the active log data sets on different disk volumes. Backup DASD may also be located at remote locations and backed up using mirroring or flash copy type technologies.

The principal tools for DB2 recovery are the QUIESCE, REPORT, COPY, RECOVER, and MERGECOPY utilities.

The *QUIESCE* online utility establishes a quiesce point (the current log RBA or log record sequence number (LRSN)) for a tablespace, partition, tablespace set, or list of tablespaces and tablespace sets, and records it in the SYSIBM.SYSCOPY catalog table. A successful QUIESCE improves the probability of a successful RECOVER or COPY. You should run QUIESCE frequently between regular executions of COPY to establish regular recovery points for future point in time recovery.

The *REPORT* online utility provides information about tablespaces. Use REPORT TABLESPACESET to find the names of all the tablespaces and tables in a referential structure, including LOB tablespaces. Use REPORT RECOVERY to find information necessary for recovering a tablespace, index, or a tablespace and all of its indexes. The REPORT utility also provides the LOB tablespaces associated with a base tablespace.

The *COPY* online utility creates up to four image copies of any of the following objects:

- ▶ Tablespace
- ▶ Tablespace partition
- ▶ Data set of a linear tablespace
- ▶ Index space
- ▶ Index space partition

There are two types of image copies:

- ▶ A full image copy is a copy of all pages in a tablespace, partition, data set, or index space.
- ▶ An incremental image copy is a copy only of pages that have been modified since the last use of the COPY utility.

The copies are used by the RECOVER utility when recovering a tablespace or index space to the most recent time or to a previous time. Copies can also be used by MERGECOPY, RECOVER, COPYTOCOPY and UNLOAD.

The *RECOVER* online utility recovers data to the current state or to a previous point in time by restoring a copy, then applying log records. The largest unit of data recovery is the tablespace or index space. The smallest is the page. You can recover a single object, or a list of objects. RECOVER recovers an entire tablespace, index space, a partition or data set, pages within an error range, or a single page. You recover data from image copies of an object and from log records containing changes to the object. If the most recent full image copy data set is unusable, and there are previous image copy data sets existing in the system, then RECOVER uses the previous image copy data sets.

The *MERGECOPY* online utility merges image copies produced by the COPY utility or in-line copies produced by the LOAD or REORG utilities. It can merge several incremental copies of a tablespace to make one incremental copy. It can also merge incremental copies with a full image copy to make a new full image copy.

MERGECOPY operates on the image copy data sets of a tablespace, and not on the tablespace itself.

Refer to the *DB2 Administrator Guide* and *DB2 utility Guide Reference* for your level of DB2. Select backup and restore procedures that are in line with your operational procedures and needs.

**Note:** Make sure that database backup files are defined and backup procedures are established for your installation.

## 12.2.4 Other configuration and product files

It is important to backup other configuration and product files that are not part of OnDemand but that are needed to support the OnDemand environment. These additional files include:

- ▶ z/OS and configuration files
- ▶ z/OS component software and configuration files (for example, DB2, OAM, TCP/IP)
- ▶ sysplex and LPAR definition files
- ▶ Files that define the environment
- ▶ Any external Indexers used and their configuration files (for example, ACIF, XENOS)
- ▶ Any exits developed for use with OnDemand

Many but not all of the above files may already be part of the established backup procedures within your organization. If not, make sure that they are.

Note, data such as the access files for the Library Server, full text indexes, or the storage areas in the Object Servers are required for the operation of OnDemand. They must be backed up. These are external files that are not part of the database. They must be accessible for reading or editing when the database is down. The backup strategy must ensure that these files are also backed up using operating system or third-party tools.

## **12.3 Object Server backup and recovery**

In this section, we discuss backup and recovery for Object Server.

### **12.3.1 Stored reports**

OnDemand can store copies of reports in cache storage and/or archive storage.

The primary purpose of cache storage is short-term, high-speed storage and retrieval of reports. Cache storage consists of disk storage volumes maintained by OnDemand on one or more Object Servers.

The primary purpose of archive storage is for the long-term storage and retrieval of reports. Reports in archive storage can also be used as backup copies, in the event that cache storage becomes corrupted or unavailable.

### **12.3.2 Cache storage**

Cache storage is the primary, short-term storage location for reports. Cache storage consists of disk storage volumes maintained by OnDemand on one or more Object Servers. If you do not copy reports to archive storage when you store them in OnDemand, you need to consider how you can recover the reports in the event that you need to do so (for example, if a device fails). Cache storage can be protected by maintaining it on high-availability storage devices. If no high-availability storage is available, we recommend that backups of reports in cache storage (the HFS or ZFS datasets) be taken on a regular schedule.

### **12.3.3 Archive storage**

Archive storage is the primary long-term storage location. Reports in archive storage can also be used as backup copies, in the event that cache storage becomes corrupted or unavailable. Archive storage consists of optical or tape storage volumes managed by the archive storage manager, optionally OAM, VSAM or TSM.



Generally, a system is configured to copy reports to archive storage and to take a backup of that object on tape or optical at the same time, when the report is loaded into the system; however, you must configure the system to support multiple copies of reports. You configure OnDemand to use archive storage by defining VSAM files, OAM objects, or TSM objects in the administration client.

If you do not plan to copy reports to archive storage, we recommend that you take regular backups of the file systems that comprise cache storage. If a media failure occurs or cache storage becomes corrupted, users cannot retrieve reports until the file systems are restored.

OnDemand retrieves the primary copy of the report from archive storage after the report has been removed from cache storage. Customers with special business, legal, or performance reasons may want the system to maintain a backup copy of their reports in archive storage. The backup copy can also be used if the primary copy becomes corrupted or unavailable. You must configure the archive storage manager to maintain a backup copy of reports in archive storage. See your storage administrator for details about defining and managing multiple copies of reports, backup and recovery of data, and scheduling operations.

## 12.4 Object Access Method (OAM)

In this section, we provide an introduction to OAM and show its relationship with OnDemand in a z/OS environment. For more information about setting up OAM, refer to another redbook, *DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426.

OAM is the DFSMSdfp™ component that manages a class of data, called objects, in a z/OS environment. Objects are bit strings which are handled as one big byte string rather than processing them as records, as is done with data sets. The content of this byte string is not known to OAM. There are no restrictions on the data type of this object; it can be an image, compressed data, or coded data.

How to handle this data is left up to the application. OAM is designed to handle an unlimited number of objects, which can be stored on magnetic disk, magnetic tape, or optical storage. Objects are different from data sets, which are handled by existing access methods. The following characteristics distinguish objects from traditional data sets:

- ▶ **Lack of record orientation:** There is no concept of individual records within an object.
- ▶ **Broad range of size:** An object may contain less than one kilobyte of data or up to 150 megabytes of data.

- ▶ **Volume:** Objects are usually much smaller than data sets; however, they can use much more external storage, depending on the kind of application creating them, such as image applications.
- ▶ **Access time requirements:** Reference patterns for objects change over time, allowing less critical objects to be placed on lower cost, slower devices, or media.

### 12.4.1 OAM components and SMS terminologies

In this section, we cover three components of OAM and the SMS terminologies.

#### ***OAM components***

The functions of OAM are performed by three subcomponents:

- ▶ ***Object Storage and Retrieval (OSR) component:*** Provides an application programming interface (API) for OAM. All OAM API functions are requested via the OSREQ assembler macro. Applications use this interface to store, retrieve, query, and delete objects, as well as to change information about objects. OSR stores the objects in the storage hierarchy and maintains the information about these objects in DB2 databases. OSR functions invoked through the application programming interface require the OAM Thread Isolation Support (OTIS) application for administrative processing.
- ▶ ***Library Control System (LCS) component:*** Writes and reads objects on tape and optical disk storage. It also manipulates the volumes on which the objects reside. The LCS component controls the usage of optical hardware resources that are attached to the system.
- ▶ ***OAM Storage Management Component (OSMC):*** Determines where objects should be stored in the OAM storage hierarchy, manages object movement within the object storage hierarchy, manages expiration attributes that are based on the installation storage management policy that is defined through SMS, and creates the requested backup copies of the objects. OSMC also provides object and volume recovery functions.

#### ***SMS terminologies***

To provide a better understanding of OAM, we explain some SMS terms:

- ▶ ***SMS storage class:*** A storage class is a collection of performance goals and availability and accessibility requirements that are defined to SMS. It is used to select a device to meet those goals and requirements. Usually, three storage classes are set up for OAM where the names of the storage classes are set up by the storage administrator based on the naming convention in the enterprise. These storage classes are:
  - **OAMDASD:** Objects are stored in a DB2 table on fast magnetic disk.

- OAMTAPE: Objects are stored on magnetic tape including tape robots.
- OAMOPTIC: Objects are stored on a 3995 optical device.

Note: The cache storage on an HFS or a zFS file system is not part of these SMS constructs.

- ▶ *SMS storage group*: An SMS storage group is a collection of storage volumes and attributes that are defined by the installation. Storage groups, along with storage classes, help reduce the requirement for users to understand the physical characteristics of the storage devices which contain their data.

In an OAM environment, Object Storage Groups allow the storage administrator to define an object storage hierarchy. The object storage hierarchy classifies storage areas according to location and, therefore, according to retrieval response time. Each object storage hierarchy must contain an object directory, containing control information about each object. Additionally, the hierarchy can have:

- DB2 object storage tables on DASD
  - Optical volumes that are associated with optical libraries (real or pseudo), and stand-alone or operator-accessible optical disk drives
  - Tape volumes that are associated with tape libraries or stand-alone tape drives
- ▶ *SMS Management Class*: Management classes define the space and availability requirements for data sets. Class attributes control backup, migration, retention of data, and release of unused space. OSMC uses information from the management classes to determine which automatic management processes should be performed upon corresponding OAM objects.
  - ▶ *Automated Class Selection (ACS) routine*: ACS routines are used to assign class and storage group definitions to data sets and objects. ACS routines are written in the ACS language, which is a high-level programming language similar to that used for the construction of TSO CLISTs. The ACS translator is used to convert the routines to object form so they can be stored in the SMS configuration.
  - ▶ *OAM Collection*: A collection is a group of objects typically having similar performance, availability, backup, retention, and class transition characteristics. A collection is used to catalog a large number of objects, which, if cataloged separately, could require an extremely large catalog. Every object must be assigned to a collection. Object names within a collection must be unique. The same object name can be used in multiple collections. Each collection belongs to one and only one object storage group. Each storage group can contain one to many collections.

## 12.4.2 Establishing OAM recovery procedures

As part of disaster recovery plan, you need to establish and test the following procedures:

- ▶ Recovering DB2 databases
  - DB2 object storage databases
  - The optical configuration database
  - The OAM administration database
- ▶ Recovering single objects from removable media
- ▶ Recovering an entire optical or tape volume
- ▶ Accessing backup objects automatically
- ▶ Recovering collection name entries in a catalog

### Recovering DB2 databases

The recoverable structure of data in DB2 is the tablespace. To ensure recoverability, make an image copy when creating each tablespace in the optical configuration database, OAM administration database, and all the tablespaces in each of the object storage databases. For information regarding how to make these image copies, refer to *DB2 for OS/390 Administration Guide*.

Your installation determines how often to make backup copies, based on the usage of each tablespace. Use this original image copy as a base, and make subsequent periodic incremental image copies of each tablespace.

At specified intervals, (best defined based on the usage of each tablespace), perform a MERGECOPY on the base (original, full-image copy) and subsequent incremental image copies to establish a new base. After creating the new base level, perform subsequent incremental image copies in relation to this new base.

The main benefit of periodically using MERGECOPY to create a new base is the recovery time savings at the time of the failure. Because merge copies can be time-consuming, it is best to perform the task on a timely and convenient basis.

To recover a tablespace, merge the content of the DB2 recovery log with the most recent full image copy of the tablespace. Because each change made to the database is recorded in the DB2 recovery log, the merge restores the tablespace to its last point of consistency prior to system failure.

Note the following:

- ▶ In DB2, point of consistency is a term that designates a time when all recoverable data accessed by an application program is consistent with other data. It is also known as sync point or commit point. Refer to *DB2 for OS/390 Administration Guide* for more information.

- ▶ The entries within the DB2 collection name table are synchronized with a corresponding collection name entry in the catalog. Recovery of the DB2 collection name table must result in a table consistent with the catalog.
- ▶ If any action is taken such that it permanently removes an entry from the DB2 collection name table, the corresponding entry must be deleted from the catalog. After a collection entry is removed from the DB2 collection name table and the catalog, objects contained within the collection are no longer accessible or managed by OSMC.

## Recovering single objects from removable media

OAM contains a single object recovery utility for recovering a single object from removable media. The system creates a new primary copy from the backup copy (if one exists) using the following criteria:

- ▶ If the primary object resides on optical disk, the backup copy (could be on either optical disk or tape) is used to create a new optical primary copy.
- ▶ If the primary object resides on tape, the backup copy (could be on either optical disk or tape) is used to create a new tape primary copy.
- ▶ If the primary object resides on DASD, the backup copy (could be either on tape or optical disk) is used to create a new DASD primary copy.

The operator starts the single object recovery utility to copy the object.

### ***Procedure to recover a single object***

The following is the procedure to recover a single object:

1. Enter the following command:

```
F OAM,START,OBJRECV,collection-name,object-name.
```

2. The system issues the following message:

```
CBR1000I OAM START command execution scheduled.
```

If the backup volume is an optical volume and does not reside in an optical library, the system issues the following message:

```
CBR4400A Mount volume volser on drive drive-name. Shelf location is shelfloc.
```

If the backup volume is a tape volume, the system issues the following message:

```
IEC501A M drive-Addr,volser,label,,,data_set_name.
```

3. Mount the optical volume or tape volume identified by volser.
4. When recovery is complete, the system issues the following message:  
CBR9830I Single Object Recovery complete for collection collection-name, object object-name.

If any error occurs during the single object recovery process, additional messages may be issued identifying the error, and message CBR9830I is not issued.

Refer to *OS/390 MVS System Messages, Vol 2 (ASB-EWX)*, GC28-1785, for the appropriate action to be taken in response to error messages.

For further information about this procedure, refer to the section “Starting Object Recovery for Single Objects” in *DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426.

## **Recovering an entire optical or tape volume**

OAM contains a utility program that recovers the objects from an unusable optical or tape volume to an usable volume. This utility is called the Volume Recovery utility. The Volume Recovery utility is used in the event that an optical or tape volume is rendered unreadable, either because of physical damage, or the volume cannot be found.

The Volume Recovery utility is used for two types of volume recovery:

- ▶ Volumes containing primary objects belonging to the OBJECT storage group that contain backup copies of the objects.
- ▶ Backup volumes belonging to the OBJECT BACKUP storage group can be recovered from the primary copies of the objects (DASD, optical, or tape). All storage groups that contain objects that need to be recovered must be defined as part of the ACDS configuration.

To recover a primary optical or tape volume, all of the backup volumes containing backup copies of the objects on the primary volume are needed whether they are optical or tape. Although all backups are written to optical or all backups are written to tape, it is still possible to have backups on both optical and tape. For example, one storage management cycle for the storage group may have been run after OAM was initialized with a CBROAMxx PARMLIB member that contained a SETOAM statement specifying a tape unit name for the OBJECT BACKUP storage group that caused backups to be written to tape. Another time, a storage management cycle for the storage group may have been run after OAM was initialized, the START OAM command may be invoked with either one of the option:

- ▶ Without a CBROAMxx PARMLIB member
- ▶ With a CBROAMxx PARMLIB member that contains no SETOAM statements or a SETOAM statement that does not specify a tape unit name associated with the OBJECT BACKUP storage group

Either of these options will cause backups to be written to an optical volume. When recovering a backup volume, every OBJECT storage group must be searched for primary objects having backup copies residing on the backup volume being recovered. The primary copy for each of these objects can be on DASD, optical, or tape. As a result, the Volume Recovery utility must identify the optical volumes as well as the tape volumes needed for recovery. If both optical and tape volumes are requested for the recovery, the operator must reply that both types are available for the recovery to continue.

### ***Starting the OAM Volume Recovery utility***

The OAM Volume Recovery utility recovers only objects that reside on an unusable optical or tape volume. It does not recover objects to DASD volumes. The utility only retrieves copies of the objects stored on DASD and recovers them to optical or tape volumes when recovering an OBJECT BACKUP volume.

Typically, some of the objects are recovered to the OBJECT BACKUP or OBJECT storage group volume currently being written, and the rest of the objects are recovered to the next OBJECT BACKUP or OBJECT storage group volume assigned.

Note, the system may issue a message requesting that a scratch volume be mounted during recovery.

Once recovery is started and GO is issued to message CBR9820D, Volume Recovery can be stopped by issuing one of the following:

**F OAM,STOP,OSMC**

This command stops the OSMC process, and thereby, stops the volume recovery processing. We recommend that this command be used to stop the volume recovery process.

**F OAM,STOP,OAM**

This command stops all OAM processing, not just the volume recovery process. Caution should be used when issuing this command for that reason.

### ***Procedure to perform a volume recovery***

The following is the procedure to perform a volume recovery

1. Enter the following command:

**F OAM,START,RECOVERY, volser**

Where volser is the volume serial number of one of the volumes being recovered.

2. The system issues the following messages:

CBR1000I OAM START command execution scheduled.

CBR9800I OAM Volume Recovery starting for volumes volser-1 and volser-2.

CBR9824I OAM Volume Recovery. The following OPTICAL volumes are needed for recovery: volser-1 volser-2 volser-3 volser-4 volser-5 volser-6 volser-7 volser-8 volser-9.

CBR9827I OAM Volume Recovery. The following TAPE volumes are needed for recovery: volser-1 volser-2 volser-3 volser-4 volser-5 volser-6 volser-7 volser-8 volser-9

Message CBR9824I gives you a list of optical volumes to retrieve that are identified by volser-n. This message allows you to get the optical volumes needed for recovery processing.

Message CBR9827I gives you a list of tape volumes to retrieve that are identified by volser-n. This message allows you to get the tape volumes needed for recovery processing.

To recover a primary volume, all of the backup volumes containing backup copies of the object on the primary volume are needed whether they are optical or tape. Also, to recover a backup volume, every OBJECT storage group must be searched for objects which have a backup copy on the backup volume to be recovered. For each of these objects, the primary copy is used to recover the backup volume. The primary copy of these objects could be on DASD, optical, or tape. As a result, the Volume Recovery utility must identify the optical and tape volumes needed for the recovery.

3. If optical volumes are to be retrieved, the system issues the following message:

CBR9820D Reply 'QUIT' to terminate or 'GO' to proceed with recovery.

4. If the optical volumes are not available, reply QUIT to terminate recovery, and start again when the optical volumes have been retrieved.

If tape volumes are to be retrieved, the system issues the following message:

CBR9810D Reply 'QUIT' to terminate or 'GO' to proceed with recovery.

5. If the tape volumes are not available, reply QUIT to terminate recovery, and start again when the tape volumes have been retrieved.

If you reply QUIT to either CBR9820D or CBR9810D, the system issues the following message:

CBR9821I OAM Volume Recovery ENDING, n objects selected for recovery.

6. If the optical volumes are available, reply GO to CBR9820D. If the tape volumes are available, reply GO to CBR9810D.



**Note:** If some of the volumes are available and others are not, recovery can still be performed for objects from the available volumes. Not all the volumes must be available for volume recovery to proceed; however, for efficiency purposes, you may want to wait until all the volumes are available. This eliminates the need to rerun the utility when the other volumes become available.

7. The system issues the following message for each optical volume listed in message CBR9824I:

CBR4400A Mount volume volser on drive drive-name. Shelf location is shelfloc.

8. The system issues the following message for each tape volume listed in the message CBRR9827A:

IEC501A M drive-Addr,volser,label,,,data\_set\_name.

Message CBR9824I may identify volumes which are either library-resident or shelf-resident optical volumes. The system automatically mounts the library-resident optical volumes; therefore, a mount message is not issued for them. The mount message CBR4400A requests only shelf-resident optical volumes for recovery.

It is possible that no shelf-resident optical volumes are needed for the recovery of an optical disk. In this case, messages CBR9824I, CBR9820D, and CBR4400A are not issued.

9. Mount the optical volume identified by volser in message CBR4400A and any tape volumes identified by volser in message IEC501A.

When recovery is complete, the system issues the following message:

CBR9821I OAM Volume Recovery status, n objects selected for recovery.

Status is either ENDING or RESTARTING:

- ENDING: This indicates that the process is complete for the requested optical disk or tape volume.
- RESTARTING: This indicates that the capacity of the utility was exceeded, and the utility restarts to recover the remaining objects.

If any error occurs during the volume recovery process, additional messages may be issued identifying the error and message CBR9821I will not be issued in this scenario. Refer to *OS/390 MVS System Messages, Vol 2 (ASB-EWX)*, GC28-1785, for the appropriate action to be taken in response to error messages.

For further information about this procedure, refer to section "Recovering an Entire Optical or Tape Volume" in *DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426.

### **Accessing backup OAM objects automatically**

OAM allows your application to obtain the backup copy of an object when the primary copy of the object resides on a removable media under one of the following conditions:

- ▶ Removable media that is marked not readable (possibly damaged or destroyed)
- ▶ Removable media that is in a library that is offline or pending offline
- ▶ Removable media that is in a library that is not operational

When you activate this function for one or all of the above conditions, and that condition exists when retrieving an object, OAM attempts to obtain the backup copy, if one exists. If this function is inactive and the primary copy of the object is not available for any of the above reasons, an error returns and the reason code is passed back to the application. If no backup copy exists and the function is active, an error returns and the reason code is passed back to the application.

The operator activates and deactivates this function through an operator command. For further information about this procedure, refer to section "Starting Automatic Access to Backup Copies of Objects" in *DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426.

### **Recovering OAM collection name catalog entries**

OAM attempts to keep collection name catalog entries up to date. This cannot be accomplished if the catalog entry does not exist or if the catalog is unusable (for example, because of I/O errors). Recovery of the catalog may be required.

Standard catalog recovery procedures apply to recovering catalog entries for collection names. Those procedures usually involve making an image copy (for example, IDCAMS EXPORT) at certain intervals and restoring that copy (for example, IDCAMS IMPORT) to recover an unusable catalog.

If collection name catalog entries were added after the image copy was made, restoring an image copy does not complete the recovery; you must also recreate those added entries. When a collection name entry from the collection name table is lost, objects in that collection will not be processed in a storage management cycle. If you do not have a program or program product to apply a journal of those additions, you can use IDCAMS to recatalog those individual entries.

For further information about the use of IDCAMS with collection name entries in the catalog, refer to the *DFSMS/MVS Access Method Services for ICF* manual.

## 12.5 Virtual Storage Access Method (VSAM)

OnDemand only supports VSAM Linear Datasets.

In this section, we provide an introduction to DFSMS VSAM Linear Datasets and show its relationship with OnDemand in a z/OS environment.

For more information about VSAM Linear Datasets, refer to the *DFSMS MVS Guide*.

VSAM is the DFSMS MVS component that manages a class of data, called objects, in a z/OS environment. Objects, as mentioned earlier in this chapter, are bit strings which are handled as one big byte string rather than processing them as records, as is done with data sets. There are no restrictions on the data type of this object. It can be an image, compressed data, or coded data.

How to handle this data is up to the application. VSAM is designed to handle an unlimited number of files. Each VSAM data set equates to one OnDemand storage object, which can be stored on magnetic disk, magnetic tape, or optical storage. Each OnDemand storage object can contain multiple reports. The number of reports in the storage object is dependant on the report size, the object size and the compression method used. Also, each VSAM data set is an entry in the MVS catalog.

DFSMSHsm™ is the MVS component that is used to backup and restore VSAM Linear DataSets.

### 12.5.1 DFSMSHsm

DFSMSHsm is a functional component of the DFSMS/MVS® family, used for backing up, recovering data, and managing space on volumes in the storage hierarchy.

DFSMSHsm is a DASD storage management and productivity tool for managing low-activity and inactive data. It relieves you from manual storage management tasks and improves DASD use by automatically managing both space and data availability in a storage hierarchy.

DFSMSHsm cooperates with the other products in the DFSMSdfp family to provide efficient and effective storage management. DFSMSdfp provides a Storage Management Subsystem (SMS) that allows storage administrators to

control the use of storage. The Storage Management Subsystem provides storage groups, storage classes, management classes, and data classes that control the allocation parameters and management attributes of data sets. DFSMSHsm performs space management and availability management of each data set as directed by the management class attributes of that data set. In addition, the storage group controls the allocation of the data set when DFSMSHsm returns the data set to level 0 (L0) storage.

Your data is stored on level 0 volumes, either SMS-managed or non-SMS-managed. Only level 0 DASD volumes can be SMS managed. Migration DASD and backup DASD should never be managed by SMS. Migration, backup, and dump tapes can be SMS-managed in tape libraries.

By definition, level 0 (L0) volumes contain data sets that are directly accessible to you and the jobs you run. DFSMSHsm-managed volumes are those L0 volumes that are managed by the DFSMSHsm automatic functions. These volumes must be mounted and online when you refer to them with DFSMSHsm commands.

### **DFSMSHsm functions overview**

Your objective in using DFSMSHsm is to make the most efficient use of all your DASD storage, primarily by making better use of the level 0 volumes. Better use of the level 0 volumes results when:

- ▶ You specify thresholds for DFSMSHsm-managed volumes, and you can assure the users that space will be available for extending old data sets or allocating new data sets (space management).
- ▶ You can assure the users that a copy of their data sets will be available if the data sets should accidentally be lost from the level 0 volumes (availability management). Such assurance that you can provide backup copies may encourage users not to unnecessarily maintain their own backup copies on level 0 volumes.

DFSMSHsm allows you to perform both availability management and space management. This can be done automatically on a periodic basis and/or by issuing specific commands when manual operations are necessary or desirable. The dump and backup copies are made to tape (for dump) or to DASD or tape (for backup).

### ***Use of storage groups and management classes***

By managing storage with SMS, you establish the following:

- ▶ Storage classes to define levels of service provided to data sets
- ▶ Storage groups to govern the volumes to which data sets are allocated and subject to management by DFSMSHsm

- ▶ Storage groups to define which volumes are used as target volumes of the fast replication backup versions
- ▶ Management classes to define how DFSMSHsm manages the data sets
- ▶ Data classes to govern values for initially allocating the data sets
- ▶ Automatic class selection (ACS) routines that automatically select storage classes, management classes, data classes, and storage groups to govern the data sets

DFSMSHsm is concerned mainly with DASD storage groups and with management classes. DASD storage groups allow you to pool volumes for the purpose of defining how and whether:

- ▶ Automatic dumps are performed on the volumes and to what dump classes
- ▶ Automatic backups are performed on the volumes
- ▶ Automatic volume space management is performed on the volumes and at what level of occupancy

Management classes allow you to group data sets logically and specify how they are managed by space management and availability management.

As users allocate data sets, the automatic class selection (ACS) routine associates them with the appropriate storage class, storage group, management class, and data class (data and management classes are optional). If a data set is not SMS managed, SMS does not associate any class or storage group with the data set.

The storage group and the management class are related only by the needs of particular data sets. That is, SMS can allocate data sets that are associated with different management classes to the volumes assigned to a single storage group. In using DFSMSHsm to manage SMS-managed storage, you must think in terms of managing data sets rather than in terms of managing volumes. An exception to the preceding statement is the automatic dump function, which operates on a volume basis.

The four discussed backup methodologies are:

- ▶ Dump and restore functions
- ▶ Backup and recovery functions
- ▶ Aggregate backup and recovery functions
- ▶ Fast replication function

### ***Dump and restore functions***

Because the automatic dump is a volume function, the attributes that govern it are taken from the storage group. DFSMSHsm copies the data sets from level 0

volumes to dump volumes. Dump can be performed only on completed volumes. Restore, the reverse of dump, moves dumped data sets from dump volumes to level 0 volumes. There are conditions under which a data set, in contrast to a volume, can be restored.

### ***Backup and recovery functions***

The backup function is a data-set-level function when DFSMSHsm is processing SMS-managed volumes. That is, the management class attributes define how the data set is treated for creation and retention of backup versions.

On the same volume, you can have data sets with different:

- ▶ Permissions for automatic backup
- ▶ Conditions for command backup
- ▶ Minimum frequencies of backup
- ▶ Numbers of backup versions
- ▶ Retention periods for backup versions

For backup, DFSMSHsm makes copies (versions) of changed data sets residing on level 0 volumes onto backup volumes. DFSMSHsm uses the management class attributes and the guaranteed backup frequency attribute of the storage group for each data set to determine whether to copy the data set. After the data sets have been backed up, DFSMSHsm determines from the management class attributes for each data set how many backup versions to keep and how long to keep them.

Backup can occur to either DASD or tape, whichever you specify. If backup is performed to DASD, your DASD volumes can become filled with the current backup versions and earlier backup versions that have not been discarded. When DASD backup volumes become full, DFSMSHsm transfers the old (non-current but valid) backup versions to spill backup volumes. The spill backup volumes can be either tape or DASD volumes.

If the backup or the spill processes are performed to tape, the tapes eventually contain many invalid versions that have been superseded by the more current versions. When tape volumes have many invalid versions, they are selected by the recycle process, which moves valid versions to spill backup volumes.

Recovery, the reverse of backup, returns data sets from the daily backup volumes or the spill backup volumes to the level 0 volumes. Recovery can be performed only by command, and can be performed for individual data sets or for complete volumes.

For additional information about the backup and recovery function, refer to *DFSMSHsm Storage Administration Guide*, SC35-0421.

### ***Aggregate backup and recovery functions***

The aggregate backup and aggregate recovery functions provide you with the capability to back up and recover a user-defined group of data sets. The user-defined group of data sets may be those belonging to an OnDemand data set, or any combination of data sets that you want treated as a separate entity.

With the aggregate backup and aggregate recovery functions, you can:

- ▶ Define the components of an aggregate.
- ▶ Back up data sets by aggregate.
- ▶ Recover data sets by aggregate.
- ▶ Duplicate your aggregates at a remote site.
- ▶ Resume business at a remote location if necessary.

For additional information about aggregate backup and recovery support (ABARS), refer to Chapter 8, "Aggregate Backup and Recovery Support (ABARS)", in *DFSMSHsm Storage Administration Guide*, SC35-0421.

### ***Fast replication function***

DFSMS provides a volume-level fast replication function that allows you to create a backup that is managed by DFSMSHsm with minimal application outage. The fast replication function supports the FlashCopy and SnapShot functions.

By using the copy pool SMS construct, you can define a set of storage groups that DFSMSHsm processes collectively for fast replication functions. DFSMSHsm recovers a fast replication backup version from the volume or copy pool level, but not at the data set level.

For additional information about the fast replication function, refer to *DFSMSHsm Storage Administration Guide*, SC35-0421.

## **12.6 Tivoli Storage Manager (TSM)**

IBM Tivoli Storage Manager provides automated, policy-based, distributed data and storage management in an enterprise network environment.

### **12.6.1 TSM Overview**

Internally, TSM is similar to OAM in that it is composed of a database and storage pools. It uses policies to manage the movement of data between clients, servers, and storage devices. The main difference between TSM and OAM is that TSM is implemented using a client/server architecture.

At a high level the TSM client/server is composed of:

- ▶ The TSM database: a relational database designed to manage the metadata associated with stored data objects. The metadata "points" to the locations of the client files in the storage pools. Database transactions are written to an external log file called the recovery log. The recovery log can be used to restore the database if necessary. The TSM server activities include *expiration processing*, which is the process of deleting data that is eligible to be removed from the system, and *reclamation processing*, which is the process of reclaiming the space taken by expired data. Storage volumes that have been reclaimed can be reused. The storage management policies determine where data is stored and how long TSM maintains the data.
- ▶ TSM storage pools: This is where OnDemand stores the archived data such as reports and documents.
- ▶ Graphical and command line interfaces: These are used by TSM to configure, control and monitor server operations. The same task may be accomplished using one or more of the interfaces. SQL SELECT statements and ODBC data transfer are supported for advanced database reporting.  
  
An administrative client program that you can use to control and monitor the server program activities and define storage management policies.
- ▶ The TSM API: This is an API that OnDemand uses to communicate with TSM. It is required on the Library Server and all Object Servers that use TSM.
- ▶ The device support modules: These provide support for storage devices and storage libraries. A device driver is provided to run a wide variety of disk, optical, tape, and robotic storage devices. Many native device drivers can also be used with Tivoli Storage Manager. Refer to the Tivoli Software Support Web site [atwww.ibm.com/software/sysmgmt/products/support/IBMTivoliStorageManager.html](http://www.ibm.com/software/sysmgmt/products/support/IBMTivoliStorageManager.html) for a complete list of supported storage devices.
- ▶ The TSM client: The client sends data to, and retrieves data from the TSM server. The TSM client must be installed on every machine that will transfer data to server-managed storage. The TSM server uses a unique node name to identify each TSM client instance. A password can be used to authenticate communications between the TSM client and server. Data can be recovered to the same client machine that initially transferred it, or to another client with a compatible file system format.

## 12.6.2 TSM as the OnDemand z/OS archive manager

The ability to archive OnDemand reports in Tivoli Storage Manager (TSM) is the latest archive manager alternative that has been added to the OnDemand for z/OS (delivered by APAR PQ92029).



The diagram illustrates the TSM architecture components and their interactions:

- OnDemand Client:** Represented by a computer icon. It connects to the OnDemand Server.
- OnDemand Server:** A box containing:
  - Object server** and **Library server**: Both connect to the **TSM Client API / TSM Client**.
  - TSM Client API / TSM Client**: The central interface for the OnDemand Server.
- TSM Administrative Client:** Represented by a computer icon. It connects to the TSM Server.
- TSM Server:** A box containing:
  - TSM Server program** and **Device support modules**: The core software components.
  - Storage Pools** and **TSM Database**: Data management components.
  - Optical**, **Tape**, and **DASD**: Storage media types, each with a corresponding icon (Optical disk, Tape reel, and DASD cylinder).

Interactions are shown by arrows: a single arrow from OnDemand Client to OnDemand Server, a double-headed arrow between OnDemand Server and TSM Server, and a single arrow from TSM Administrative Client to TSM Server.

If TSM is to be used as a storage manager, the IBM Tivoli Storage Manager OS/390 Unix System Services Client API Version 5 Release 2 Level 2 or greater must be installed where the OnDemand servers (either library or object server) are installed. In addition, a TSM server must be configured and active that supports that level of the TSM API. The TSM server may reside on a different host or operating system than the OnDemand server.

The most common backup methodologies are:

- ▶ Full backup
- ▶ Full and incremental backup
- ▶ Full and differential backup
- ▶ Progressive backup

Using *full backup* method, every file on a computer or file system is copied whether or not it has changed since the last backup. The advantage is that while all files are backed up at a single point in time. This provides a consistent set of data. The drawback of is that large amounts of data are regularly moved.

Using *full and incremental backup* method, full backups are done on a regular basis (for example weekly), In between full backups, regular incremental backups copy only files that have changes since the last backup. The advantage is that less data is regularly moved than in a full backup. The drawback is to restore the data, the full backup must be restored first followed by all the incremental backups in the correct sequence.

Using *full & differential backup* method, full backups are done on a regular basis (for example weekly). In between full backups, differential backups copy only files that have changes since the last full backup. The advantage is that during a restore operation only the full backup is restored followed by the last differential backup. The disadvantage is that a differential backup will back up more data because it ignores differentials that were taken between the previous full and the current differential backup.

The drawback of these common backup methodologies is that whenever a full backup is made (for example weekly), all the data is copied to another storage device, whether it has changed or not. TSM provides a fourth alternative known as progressive backup.

Using *progressive backup* method, a full backup is first made, but then only incremental backups are made from that point on. Another full backup may never be required. The technique is as follows:

1. A full backup is performed once.
2. After the full backup, incremental backups copy only files that have changed since the last backup.
3. Metadata associated with backup copies is inventoried in the TSM database. The number of backup copies stored and the length of time they are retained is specified by the administrator.

The benefits of this method are:

- ▶ Essentially eliminates redundant data backups.
- ▶ TSM automatically releases expires file space to be overwritten; this reduces operator intervention and the chance of accidental overwrites of current data.

- ▶ Over time, less data is moved than in full and incremental or full and differential backups and data restoration is mediated by the database.

Progressive backup can be thought of as combining the backup benefits of the incremental approach with the restore benefits of the differential approach.

#### 12.6.4 TSM supplied functionality

The functionality provided by TSM includes:

- ▶ Backup and Restore
- ▶ Archive and Retrieval
- ▶ Instant Archive and Rapid Recovery
- ▶ Space Manager Client

With the *Backup and Restore* TSM function, the backup process creates a copy of the file or application data that can be recovered if the original data is lost or destroyed. Unlike other backup applications, TSM implements a progressive backup methodology to move data quickly and reliably.

With the *Archive and Retrieval* TSM function, the archive process creates a copy of a file or a set of files and stores it as a unique object for a specified period of time. This function is useful for maintaining copies of vital records for historical purposes.

With the *Instant Archive and Rapid Recovery* TSM function, TSM allows for the creation of a complete set of client files, called a backup set, on the TSM server system using the most recent backup versions stored by the server.

In a process called Instant Archive, a backup set is used to retain a snapshot of a client file system for a designated period of time. The Rapid Recovery process allows you to copy backup sets onto portable media for LAN-free recovery of a client system.

With the *Space Manager Client* TSM function, a separately licensed optional feature, it provides for the automatic and transparent movement of operational data from a client system to server-managed storage. This process, called Hierarchical Space Management (HSM), is implemented as a client installation and is controlled by policy defined to the TSM server. HSM frees up space on a client machine by using distributed storage media as a virtual hard drive for that machine. Files are automatically moved and stored according to size, age, and usage. When a user accesses this data, it is dynamically and transparently restored to the client machine.

For additional information, refer to:

- ▶ *IBM Tivoli Storage Manager for UNIX Backup-Archive Clients Installation and User's Guide*, GC32-0789.
- ▶ *DB2 Content Manager OnDemand for z/OS and OS/390 Configuration Guide Version 7.1*, GC27-1373.

# High availability for OnDemand z/OS in a SYSPLEX environment

In this chapter, we cover high availability for OnDemand z/OS in a SYSPLEX environment.

The following topics are covered:

- ▶ High availability on z/OS overview
- ▶ High availability strategy for a production OnDemand z/OS application
- ▶ SYSPLEX terminology
- ▶ TCP/IP port sharing
- ▶ The “shared” OnDemand server

## 13.1 High availability on z/OS overview

In this section, we provide an overview of high availability on z/OS operating system and the high availability concept for OnDemand z/OS.

### 13.1.1 z/OS the nucleus high availability component

The z/OS operating system has roots that go back to the early days of MVS, which was designed nearly 30 years ago with high availability in mind. The operating system takes advantage of the self-healing attributes of the hardware, and extends them by adding functions such as recovery services for all operating system code, address space isolation, and storage key protection. Functions such as Workload Manager (WLM), Resource Recovery Services (RRS), and Automatic Restart Manager (ARM) assure the availability of applications.

z/OS operating systems can be configured into a *Parallel Sysplex*, which is the clustering technology for the mainframes. The main objective of a Parallel Sysplex is continuous availability without compromising perceived client performance. High availability requires at least two OnDemand servers providing the same service to all the OnDemand clients, so that these servers allow for the recovery of service when failures occur. That is, the OnDemand servers perform the backup function for each other within the cluster. High availability minimizes unplanned outages.

Continuous availability is provided with the zSeries and z/OS Parallel Sysplex clustering technology. This technology implements a data-sharing design that allows a database to be concurrently read and updated by application clones running on multiple z/OS images on one or more physical servers. Continuous availability avoids or minimizes unplanned outages (high availability) and reduces or eliminates planned outages.

*Workload Manager* balances application workload across the systems in the Parallel Sysplex. If there is a failure or a planned outage on one system, other systems within the Parallel Sysplex take over the full workload. By implementing Geographically Dispersed Parallel Sysplex™ (GDPS/PPRC) in a multi-site base or Parallel Sysplex environment, the sites can be separated by up to 40-100 KM of fiber (depending on the type of external CF links are Inter System Coupling (ISC) links) from each other.

The Parallel Sysplex clustering architecture allows for continuous availability and the avoidance of an entire site-wide disaster. Other remote copy facilities exist that enable backing up data to remote sites at even further distances.

*Metro Mirror* (previously known as synchronous Peer-to-Peer Remote Copy, or PPRC) provides real-time mirroring of logical volumes between two DS8000s

systems that can be located up to 300 KM from each other. It is a synchronous copy solution where write operations are completed on both copies (local and remote site) before they are considered to be complete. It is typically used for applications that cannot suffer any data loss in the event of a failure.

Global Copy for DS8000 allows for a non-synchronous long distance copy option suitable for data migration, transmission of database logs, and periodic off-site backup. With Global Copy, updates are continually transferred to the secondary DS8000 at the remote site in a non synchronous mode. This means updates to the DS8000 primary at the local site are considered complete (that is, ending status is returned to the application) before they are transmitted to the secondary DS8000 at the remote site. With a non-synchronous operation, the distance between the primary and secondary DS8000 will have only a minimal effect on the application response time. Therefore, Global Copy can operate at very long distances. Distances well beyond the 300 KM supported with Metro Mirror (Synchronous PPRC) for DS8000 - with the distance limited only by the capabilities of the network and channel extension technologies.

### **13.1.2 High availability concept for OnDemand z/OS**

IBM DB2 Content Manager OnDemand z/OS is an enterprise-class application. OnDemand z/OS is built on the strength of IBM DB2. One of the key capabilities that you can take advantage of is the ability to set up your production system in a high availability environment.

High availability configurations for OnDemand are designed to eliminate single point of failure in the OnDemand z/OS stack by leveraging the high availability capabilities of a Parallel Sysplex.

The OnDemand z/OS components that can be configured for high availability include the Library Server database and the Object Server database. Setup of a production environment for high availability requires a deep understanding of the various high availability, storage and networking technologies involved. Most often, malfunctions of the high availability environment are attributed to insufficient testing of the various resources that might become unavailable at some point in time. A failure of a high availability environment does not necessarily indicate there is a problem with OnDemand z/OS. Because there are many possible different configuration variants and setting up such an environment correctly is complex, we highly recommend using consulting services, such as IBM Content Management Lab Services or IBM Global Services, for OnDemand z/OS high availability planning, testing, implementation, and support.

It is beyond the scope of this redbook to discuss the actual procedures and steps of setting up a Sysplex environment. Our intention here is to discuss the

strategies that are available to you. We encourage you to explore these strategies and options further with an consulting service.

### **Factors affecting availability**

From the end users' point of view, the Parallel Sysplex can provide continuous availability. This availability can only be provided if the phone company and data lines are available (referred to in this text as the external network).

At the data center level, Parallel sysplex uses coupling technology to allow multiple CECs to work in a single system image. These CECs may be in one or more physical locations and data centers. At the highest level of availability, a Parallel Sysplex configuration has full redundancy. There is no single point of failure. Failure of any redundant component of the Parallel Sysplex is thus to the system users.

Within the Parallel Sysplex, the WLM provides dynamic workload balancing across all systems, so work will be routed to the least utilized system and routed away from a failed system. This allows for a complete CEC to fail, without the users being affected.

If WLM is not used, essentially with no sysplex, users will see an outage when a failure occurs. However, if the application uses data sharing and TCP/IP port sharing, it can be restarted immediately on another LPAR (in the same CEC) without waiting for the failing LPAR to be recovered.

*Automatic Restart Manager (ARM)* can be used to restart a failing subsystem automatically, either on the same system or on another system.

In general, there are scheduled (maintenance/upgrades) outage and unscheduled (due to failures) outage. A short list of different outage causes follows:

### **Scheduled outages (planned for upgrades)**

Scheduled outages include:

- ▶ Hardware
  - Processor upgrades, adding new features, upgrading microcode
  - DASD Upgrades, for example, moving system volumes to a new string of DASD
- ▶ Software

New software releases or upgrades to the operating system, subsystems or applications.



- ▶ Local environment  
Repair or upgrade of the local data center. For example, power supply, air conditioning and chilled water supply.
- ▶ Local network  
Hardware, software, or configuration upgrades or changes.

### **Unscheduled outage (due to failures)**

Unscheduled outage include:

- ▶ Hardware  
Processor, I/O device, communications device, DASD or printer. This is an infrequent event with current technology levels.
- ▶ Software:
  - Operating system or subsystem failures. Such as MVS, JES, VTAM®, TCP/IP and DB2.
  - Application errors may occur under abnormal conditions, or when some other software is upgraded.
- ▶ Operational error  
For example, a command wrongly submitted by a system or subsystem operator, or a job submitted incorrectly by an operator or by an automated job scheduling package.
- ▶ Local environment  
For example, data center such as power supply, air conditioning and chilled water.
- ▶ Local network  
Failure of hardware, switches/router, and cables.
- ▶ External environment  
For example, terrorism or natural disaster
- ▶ External network  
For example, phone company equipment or data line failures.

When an outage occurs in a Sysplex environment, the system will continue to function, possibly at a degraded (lower performance) level, depending on the availability of spare capacity in the remaining systems. This will continue for a period of time until the backup and recovery efforts are complete.

System capacity is often measured in number of online transactions per second, or number of completed batch jobs per hour that a system can handle at

maximum throughput. If no performance degradation is to occur, then the Sysplex configuration needs to be able to handle the full load even if one of the CECs in the Sysplex is down. On the other hand, it is entirely possible that an organization can tolerate a degraded response time for the period that it takes to recover from the failure.

The two operational times that need to be evaluated are:

- ▶ Online response: The response to users' requests will be slower.
- ▶ Batch jobs: Batch jobs will have longer elapsed times. It could be possible to reschedule some of the batch jobs until the system has recovered.

## 13.2 HA strategies for an OnDemand z/OS application

In this section, we discuss the high availability strategies for a production OnDemand for z/OS application. We represent the OnDemand system by a tiered model. For the system to be operational, at least one redundant path in each of the tiers must be operational. This tiered model is summarized in the following section.

### 13.2.1 A 4-tier logical model

OnDemand for z/OS can be represented by a 4-tier logical model. Each of the tiers consists of several components, all of which need to be taken into consideration when designing an end-to-end high availability system. This model is used for illustrative purposes only. Specific details vary on a per implementation basis.

Figure 13-1 on page 287 shows the various tiers of an OnDemand for z/OS system and their respective components. The diagram illustrates both a typical 2-tier model in which client machine (Tier 1) connect directly to the server (Tier 4) and a typical 4-tier model in which Web browsers connect through an HTTP server to a Web server which then connects to the backend server.

# OnDemand for z/OS End-to-end Tier Components

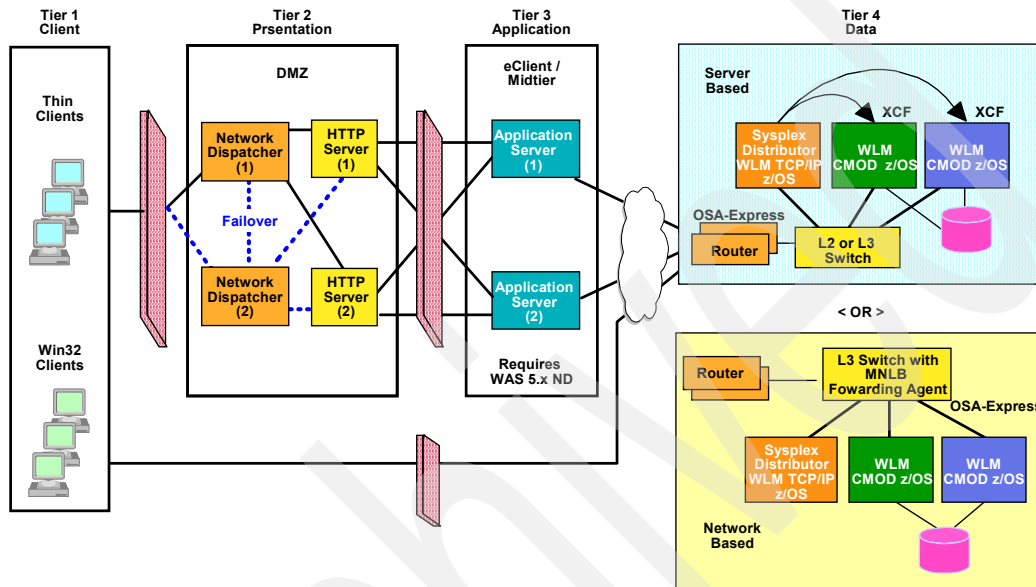


Figure 13-1 End-to-end tier components diagram

In this chapter, we concentrate on the Tier 4, the OnDemand Library Server and Object Server, and briefly discuss the other tiers.

## Tier 1

Tier 1 of an OnDemand for z/OS system consists of the following components:

- ▶ Client workstations
- ▶ Local area network
- ▶ Router or switch
- ▶ network

## Tier 2

Tier 2 of an OnDemand for z/OS system consists of the following components:

- ▶ Firewall process
- ▶ DNS server
- ▶ IP sprayer / load balancer
- ▶ HTTP server

### **Tier 3**

Tier 3 of an OnDemand for z/OS system consists of the following components:

- ▶ Firewall process
- ▶ Web Application Server (WAS)
- ▶ Mid-tier applications (ODWEK)

### **Tier 4**

Tier 4 of an OnDemand for z/OS system consists of the following components:

- ▶ Intelligent routing of inbound traffic
- ▶ Virtual IP Address (VIPA)
- ▶ Sysplex Distributor (SD)
- ▶ Workload Manager (WLM)
- ▶ Cross Coupling Facility (XCF)
- ▶ Library Server application and database
- ▶ Object Server application and database
- ▶ DB2 Data Sharing
- ▶ OAMplex and database
- ▶ Disk subsystem
- ▶ Tape and optical libraries

Tier 4, the data tier, in a Parallel Sysplex environment, is commonly referred to as the backend tier. Examples include the database manager systems, OnDemand system, mainframe transaction processing or other existing systems.

Although it is possible (and likely) for tiers 2 and 3 to be physically operating on a z/OS system. In this redbook, we discuss only the placement of tier 4 on the z/OS system, although the same principals apply to both tiers 2 and 3.

## **13.2.2 Breaking out the data tier (Tier 4)**

The OnDemand z/OS Library Server and Object Server(s) are commonly referred to as the backend data tier. With OnDemand z/OS being a DB2 application, you can take full advantage of the operating system in a SYSPLEX environment using Work Load Manager (WLM) as the traffic cop to intercept failovers and distribute requests (see Figure 13-2 on page 289).

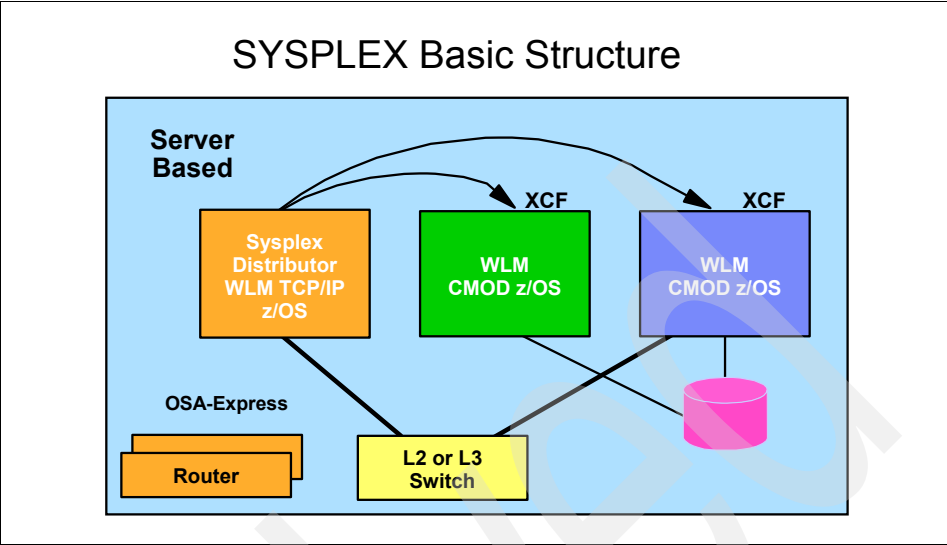


Figure 13-2 SYSplex basic structure

Figure 13-3 breaks out the data tier into 4 functional components. We examine each of the components and their failure scenarios in the following sections.

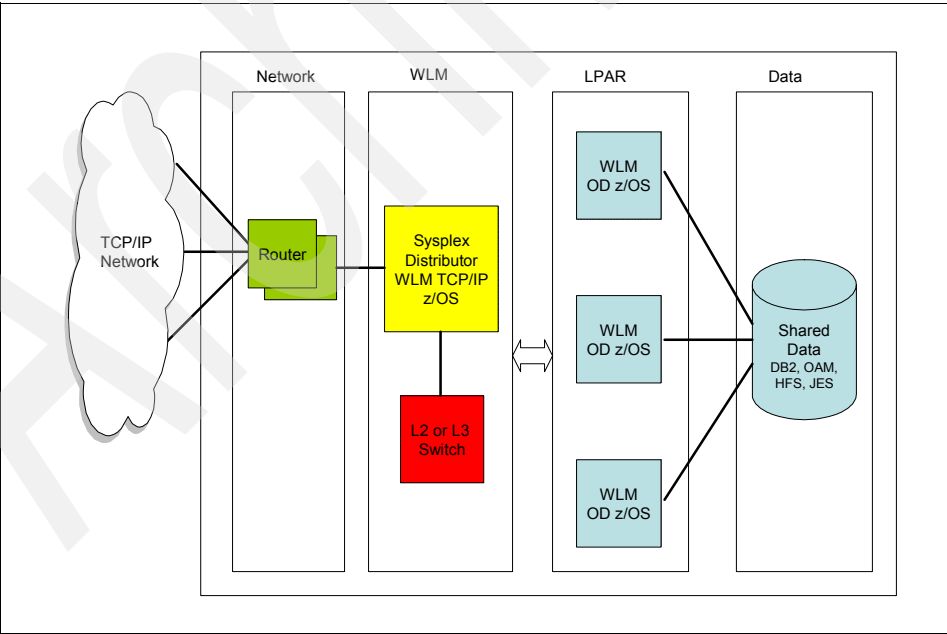


Figure 13-3 breakout of SYSplex basic structure

The four component layers of the SYSPLEX basic structure are:

- ▶ Network
- ▶ WLM
- ▶ LPAR
- ▶ Data

The *Network* layer is composed of redundant routers, the purpose of which are to connect the network to the appropriate WLM layer. If one of the routers fails, then the remaining router(s) should have sufficient network throughput to allow the usage of the system without degradation.

The *WLM* layer contains both the Sysplex distributor and a switch (both of which should be redundant for high availability). The purpose of this layer is to control the workload distribution and the sharing of data locks between the LPARs. The Sysplex distributor and the switch should both contain sufficient processing power and the LPAR connection throughput such that they do not slow down the multiple LARs that they are connected too.

The *LPAR* layer is composed of two or more LPARs residing on one or more CECs. If the LPARs are on the same CEC, then the system will be able to function with LPAR failures but not CEC failures. If the LPARs are on different CECs, then the system would be able to function with CEC failures. This is the layer in which multiple copies of the OnDemand Library Server and Object Server reside.

The *Data* layer is the layer that provides the shared data facilities, such as DB2, OAM, HFS, JES. All the OnDemand report and index data is stored in this layer. The method(s) used to backup this layer (Tape, Disk, Flash Copy) determine how fast (seconds to days) the system recovers from outages.

### 13.2.3 Intelligent routing of inbound traffic options

There are two options for intelligent routing of inbound traffic as shown in Figure 13-4 on page 291:

- ▶ Server based: TCP/IP Parallel Sysplex Distributor uses WLM and Policy Agent information and forwards all inbound traffic.
- ▶ Network based: CISCO router workload balances Multi-Node Load Balancer (MNLB) and queries WLM.

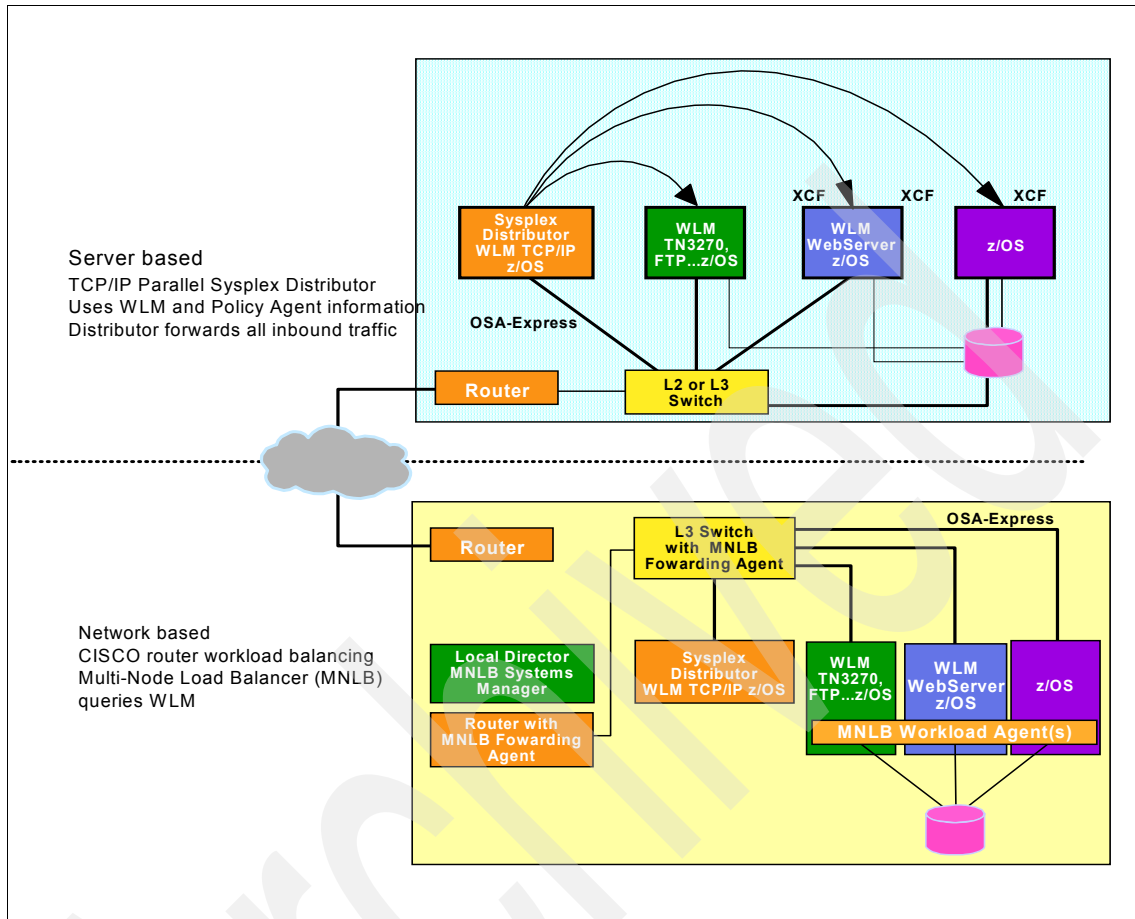


Figure 13-4 Intelligent routing of inbound traffic options

### Server based availability

Sysplex Distributor is the strategic solution for connection workload balancing and availability in the OS/390 or zSeries Sysplex. The concept is that a cluster of server instances is represented by a single IP address, which in general can be called a Cluster IP address. Since the Sysplex Distributor is built on Dynamic VIPAs, the Sysplex Distributor term is Distributed DVIPA. A DVIPA is defined on a primary TCP/IP stack via VIPADEFINE as described above, and on backup stacks as appropriate. In addition, the primary stack contains a configuration statement which identifies the DVIPA as Distributed, identifies the ports the application will use (between one and four ports per Distributed DVIPA), and the target application hosting TCP/IP stacks in the sysplex.

### ***Network based availability***

Network Dispatcher is also an approach to workload distribution across a set of application instances in a cluster, but it works with a cluster IP address, rather than with name resolution. Network Dispatcher (and the similar function in the CISCO Multi-Node Load Balancer, or MNLB) is an external entity adjacent to the Parallel Sysplex cluster. An agent in the sysplex communicates workload capacities of the nodes hosting the application to the Network Dispatcher node. The Network Dispatcher advertises ownership of the cluster IP address (application IP address) to the routing network.

OS/390 TCP/IP stacks hosting the application define the same address as a loop-back address, which is not advertised to the routing network. The Network Dispatcher must have a direct link (single IP hop) to each TCP/IP stack hosting the IP application. When a new TCP connection request arrives, Network Dispatcher consults the most recent capacity information received from the WLM agents, selects the appropriate TCP/IP for this request, and forwards the request over the direct link to the selected stack. For selected applications such as Web serving, an application advisor function in the Network Dispatcher periodically queries the application to be sure it is available and responding, and ensure that requests are routed only to functioning server applications.

Subsequent traffic from the client to the server is routed through Network Dispatcher to the same TCP/IP, though return traffic from the server application to the client need not flow through the Network Dispatcher. Clients thus see a network presence represented by a single IP address, and the servers that make up the processing that backs up the cluster address are hidden from the clients.

## **13.2.4 Achieving HA for OnDemand system**

Figure 13-5 on page 293 depicts the high availability operating environment of an OnDemand system for z/OS.



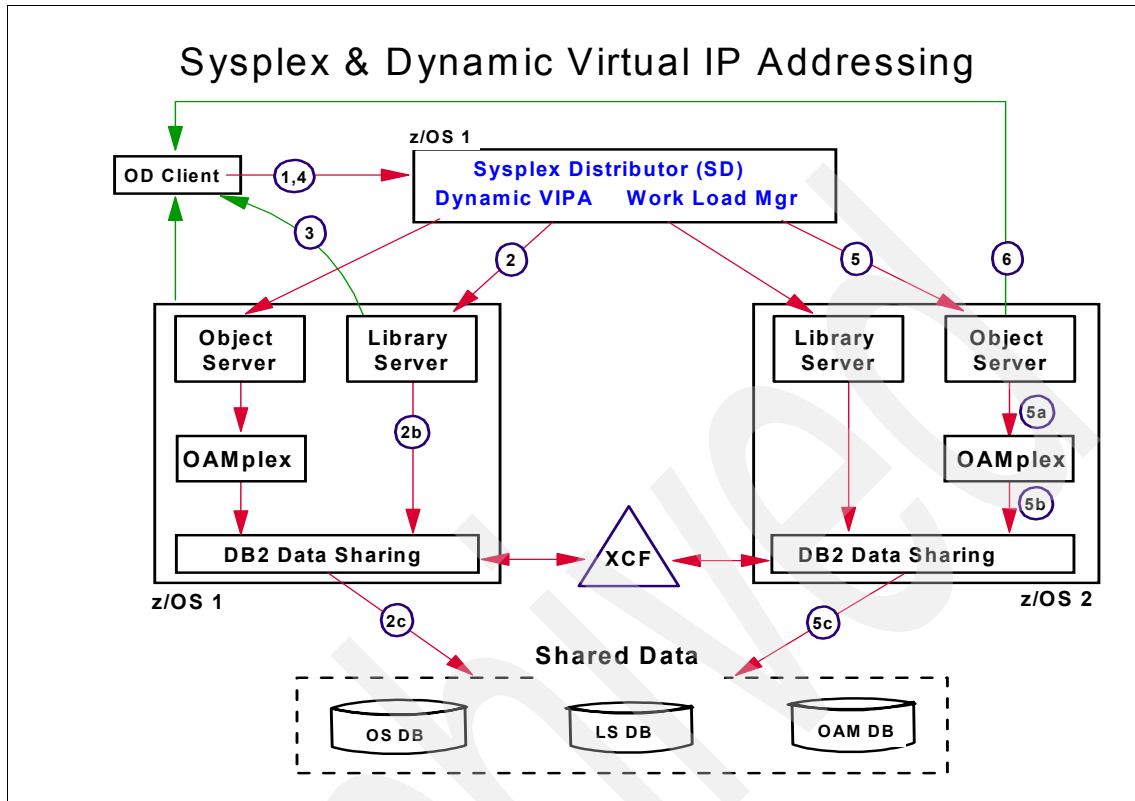


Figure 13-5 Sysplex and dynamic virtual IP addressing (VIPA)

### Library Server

Within an OnDemand for z/OS system, the Library Server (LS) provides the functions of securing access to the data, storing the metadata, searching of the metadata, and production of a query results list that are transmitted to the OnDemand client via TCP/IP.

The Library Server is implemented as DB2 control databases. The exploitation of Sysplex Distributor, Dynamic Virtual IP Addressing (VIPA) and WLM enables the creation of a high availability environment.

In Figure 13-5, an OnDemand client connects to an OnDemand server using a static IP address or host name (1). The TCP/IP connection request is intercepted by the Sysplex Distributor (1). The Sysplex Distributor/Workload Manager determine which Library Server at that time has the least load and routes the request to that Library Server (2). In this case that would be on the z/OS 1 system in the sysplex. The Library Server then accesses and queries the shared

database through the DB2 data sharing facility (2b and 2c). One or more queries may be issued to satisfy each request. The Library Server then compiles the results of the query (for example a list of documents that meet some specified criteria, and the location and node at which the documents can be found) and sends the response directly back to the OnDemand Client (3). For the duration of the transaction the Library Server communicates directly to the OnDemand client. It is the WLM that balances the connection algorithm and not the Sysplex Distributor.

## Object Server

Within an OnDemand for z/OS system, the Object Server (OS) provides the functions of storing and retrieving the OnDemand reports and documents. The Object Server may use one or more of several archiving mechanisms, namely, OAM, VSAM, TSM or cache. In Figure 13-5 on page 293, OAM is used as an example.

Following the description in Library Server section earlier, after the Library Server returns the hit list to the OnDemand client, the hit lists is displayed to the user. User may select one or more documents for viewing. The client then issues one or more requests to the Object Server to retrieve the document(s) (4). The TCP/IP request is once again routed through the Sysplex Distributor. The Sysplex Distributor/Work Load Manager determine which system in the Sysplex is available and routes the request to that system (5), in this case, z/OS 2. The Parallel Sysplex installation creates an environment in which there appears a single virtual Object Server, regardless of which Object Server that is actually accessed and which system that it is running on it will access the same OAM data/database. That is, an object is not tied to a specific WLM-created instance of an Object Server. Objects are stored through OAM and WLM can create several port addresses to transparent user requests for optimum availability and performance. The Object Server then uses the functionality provided by the OAMplex (5a), DB2 Data Sharing (5b) to retrieve the documents from the shared data pool (5c). The retrieved documents are then directly transmitted back to the OnDemand client (6).

The Object Server instances may be running in various systems within the Parallel Sysplex. Each instance is based on the number of Task Control Blocks (TCB) that is selected during Object Server customization. There may be various DB2 subsystems associated with these systems; however, DB2 also employs the coupling technology of the Parallel Sysplex to allow data sharing among various z/OS systems and DB2 subsystems. As long as the Object Server's DB2 plan is bound with the Sysplex identifier tying these multiple DB2 subsystems together, it does not matter which system in the Sysplex stores the object or later retrieves the object. The object is accessible to all systems. DB2 data sharing, in combination with the Parallel Sysplex coupling hardware and software, ensures the integrity of the data.

### 13.2.5 Availability strategy failure scenarios

While the OnDemand for z/OS system is in production, any one of its components may fail. We examine the availability strategy in place to recover from the potential failures as listed below:

- ▶ Library Server failure
- ▶ Object Server failure
- ▶ Sysplex Distributor (SD) failure
- ▶ z/OS failure
- ▶ DASD failure affecting DB2 tables
- ▶ OAM failure

#### Availability strategy - Library Server failure

When a Library Server fails (see Figure 13-6 on page 296), the following steps describe how the system handles the failure with the availability strategy incorporated in the system setup:

1. At step (1) of Figure 13-6 on page 296, a search request for the Library Server comes from an OnDemand (OD) client, that goes into Sysplex Distributor (SD).
2. At step (2) of the diagram, SD/WLM determines that Library Server on z/OS 1 is available.
3. At step (3) of the diagram, the Library Server passes request back to the OnDemand client.  
  
<<< A failure occurs: The Library Server on z/OS 1 goes down. >>>
4. At step (4), search request for the Library Server comes from the OnDemand client which goes into Sysplex Distributor.
5. At step (5), SD/WLM determines that the Library Server is down on z/OS 1 and the Library Server on z/OS 2 is up.
6. At step (6), the Library Server on z/OS 2 acquires the metadata and sends it back to the OnDemand client.

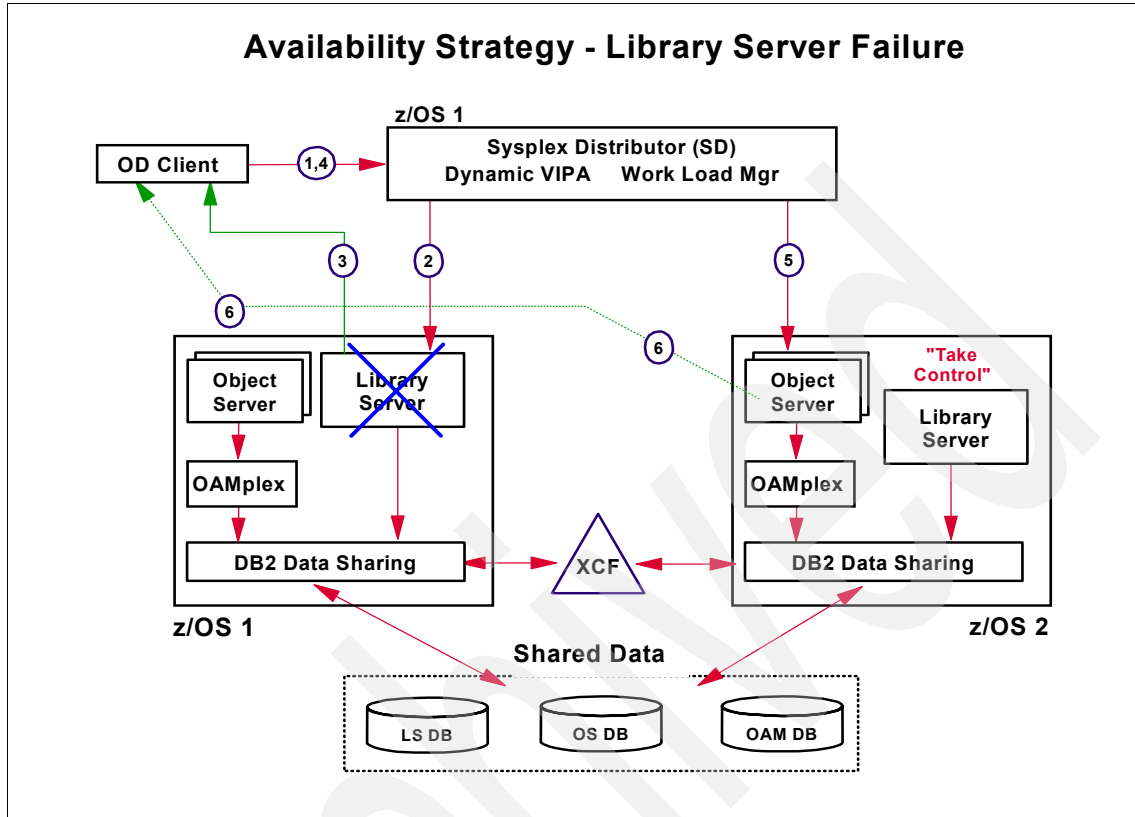


Figure 13-6 Availability strategy for Library Server failure

### Availability strategy - Object Server failure

When an Object Server fails (see Figure 13-7 on page 297), the following steps describe how the system handles the failure with the availability strategy incorporated in the system setup:

1. As shown in Figure 13-7 on page 297, step (1) indicates an OnDemand (OD) client initiates a request for the Library Server, which is actually directed into Sysplex Distributor (SD).
2. At step (2) of the diagram, the SD/WLM determines that the Library Server on z/OS 1 is available.
3. At step (3) of the diagram, the Library Server passes the request back to OnDemand client.

<<< A failure occurs: The Object Server on z/OS 1 goes down. >>>

4. At step (4) of the diagram, the OnDemand client's request to connect to Object Server goes into Sysplex Distributor.
5. At step (5) of the diagram, SD/WLM determines that Object Server is down on z/OS1 and the Object Server on z/OS 2 is up.
6. At step (6), the Object Server on z/OS 2 acquires the object from OAM and sends it to the OnDemand client.

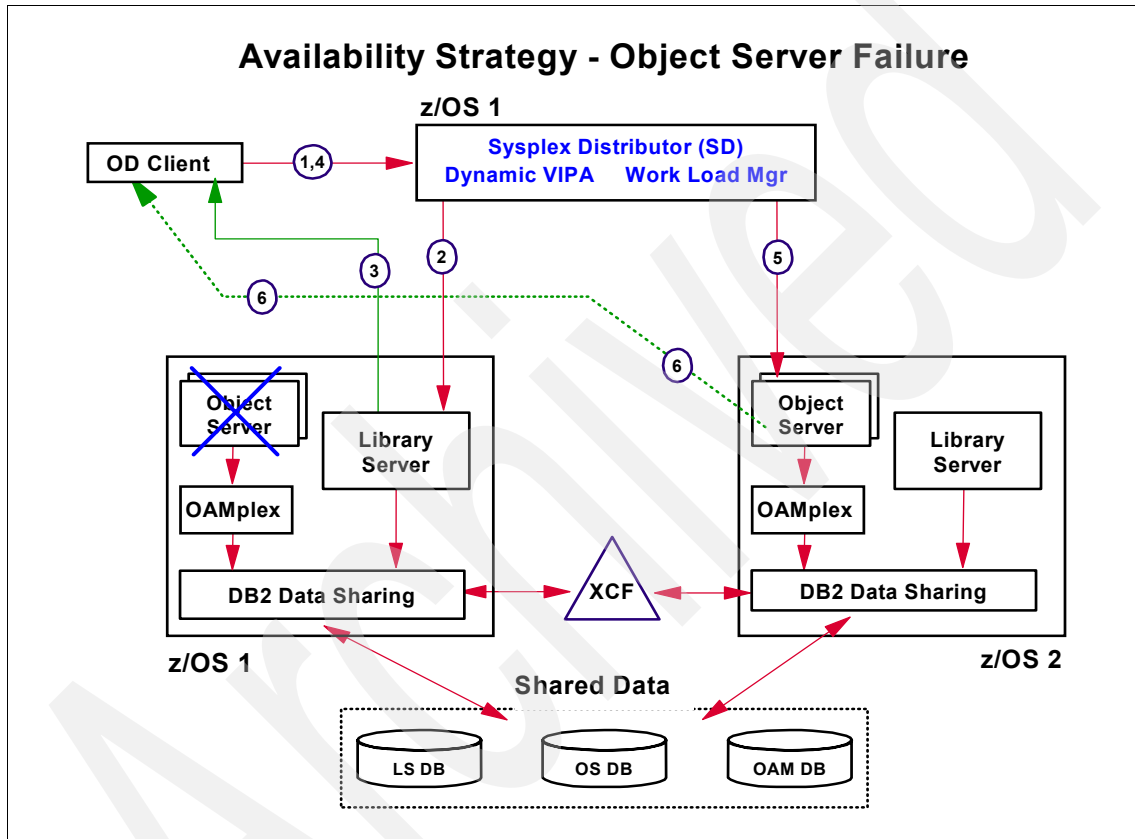


Figure 13-7 Availability strategy - Object Server failure

### Availability strategy - Sysplex Distributor failure

When the Sysplex Distributor fails (see Figure 13-8 on page 298), the following steps describe how the system handles the failure with the availability strategy incorporated in the system setup:

1. At step (1) of Figure 13-8 on page 298, a request for the Library Server from an OnDemand (OD) Client goes into the Sysplex Distributor.
2. At step (2), SD/WLM determines that Library Server on z/OS 1 is available.

- At step (3) of the diagram, the Library Server passes the request back to the OnDemand client.

<<< A failure occurs: The Sysplex Distributor on z/OS 1 goes down. >>>

- At step (4) of the diagram, a request for the Object Server from the OnDemand client comes in and gets routed to the SD Backup.
- At step (5) of the diagram, SD/WLM on z/OS 2 determines that the Object Server is busy on z/OS 1 and sends a request to the Object Server on z/OS 2.
- At step (6) of the diagram, the Object Server on z/OS 2 acquires the object from OAM and sends it to the OnDemand Client.

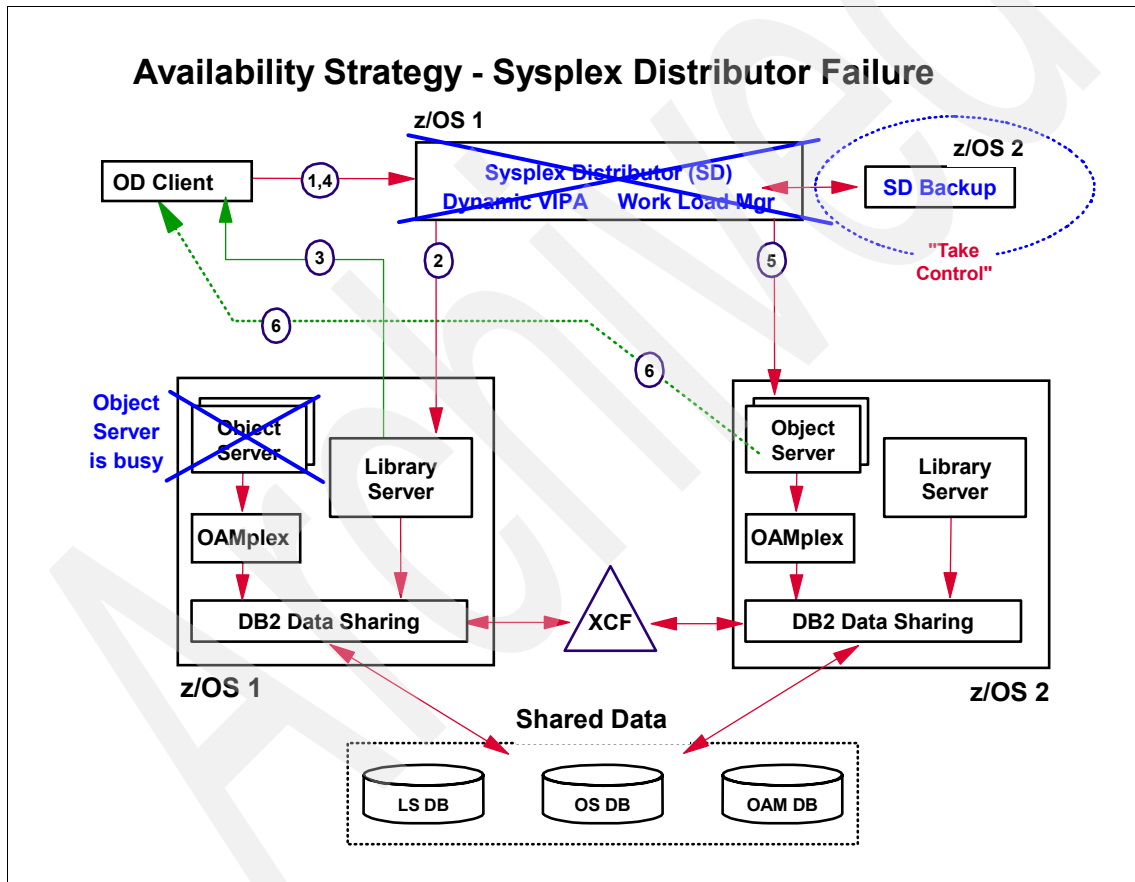


Figure 13-8 Availability strategy - Sysplex Distributor failure

## Availability strategy - z/OS failure

When the z/OS operating system fails (see Figure 13-9), the following steps describe how the system handles the failure with the availability strategy incorporated in the system setup:

1. At step (1) of Figure 13-9, a request for the Library Server comes from an OnDemand client and goes into the Sysplex Distributor.
2. At step (2), SD/WLM determines that Library Server on z/OS 1 is available.
3. At step (3), the Library Server passes the request back to the OnDemand client.

<<< A failure occurs: The z/OS 1 goes down, including SD. >>>

4. At step (4) of the diagram, a request for the Object Server comes from the OnDemand client and gets routed into SD Backup on z/OS 2.
5. At step (5), SD/WLM determines z/OS 1 is down and sends the request to the Object Server on z/OS 2.
6. At step (6), the Object Server on z/OS 2 acquires the object from OAM and sends it to the OnDemand client.

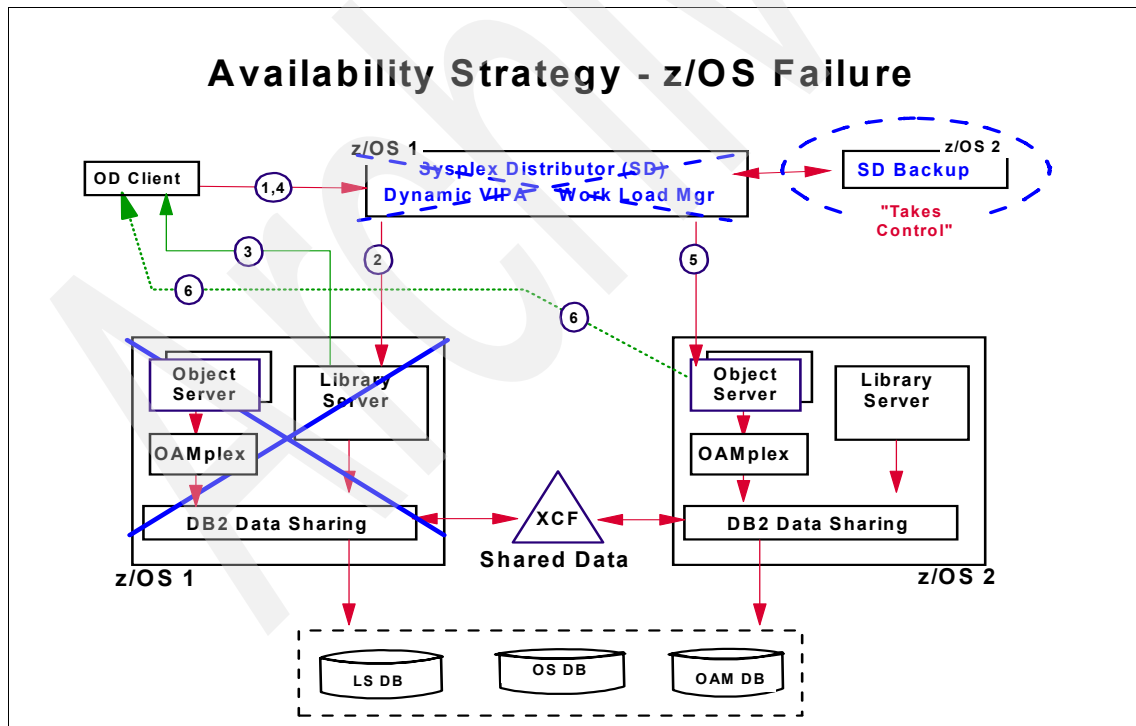


Figure 13-9 Availability strategy - z/OS failure

## **Availability strategy - DASD failure affecting DB2 tables**

There is nothing unique for OnDemand for z/OS when DASD failed that affect DB2 tables because OnDemand is a DB2 application. OnDemand for z/OS must be in a data sharing Sysplex environment to remain in a high availability state.

Below are a few recovery scenarios that may be considered:

- ▶ Standard DB2 Recovery.  
Restore tables using the last DB2 image copy and applying DB2 logs.
- ▶ Recovery 1+ hour.  
FlashCopy with ESS "Shark" DASD.  
Restore volume using the last IBM's FlashCopy and applying DB2 logs.  
The recovery period may take over an hour, depending on the size of the volume to be restored, the number of DB2 logs, and the speed of the communication channels.
- ▶ Recovery 1+ hour.  
Have GDPS/PPRC V2.8 with Hyperswap in place.  
Geographically Dispersed Parallel Sysplex (GDPS®).  
Peer-to-Peer Remote Copy (PPRC).  
Point to "mirrored" volume.  
The recovery period may take over an hour, depending on the size of the volume to be restored, the number of DB2 logs, and the speed of the communication channels.
- ▶ Recovery:  
Non-disruptive - applications keep using same device addresses  
Will affect normal performance due to dual serial writes.

## **Availability strategy - OAM failure**

In case of OAM failure, you should have availability strategies based on the storage media, DASD, optical, or tape.

### ***Availability strategy - OAM - DASD***

OAM has extra recovery procedures beyond a DASD failure.

OAM Automatic Access Backup for DASD objects

APAR: OA02967 PTF: UA04375 available 03/07/29



OAM now support automatic access to backup function for objects that reside in DB2.

Backup and recovery is a long process. If the objects could be retrieved from the backup while the 4k or 32k object table is being recovered, it should reduce customer data unavailability.

### ***Availability strategy - OAM - Optical***

As long as all instances of OAM involved with the transaction belong to the same OAMplex, any OAM object can be retrieved from any z/OS system in a Sysplex, regardless of which OAM in the sysplex stored the object or on which medium (3995 optical, tape, or DASD) the object resides.

### ***Availability strategy - OAM - Tape***

Basic concept of transaction shipping (sending transaction requests between OAMs within the same OAMplex for processing) still pertains.

However, MVS dynamic allocation is used to handle the required tape resource allocation, because OAM does not control tape resources.

Tape resources are allocated as needed and only for the time required for their use. Tape drives must be available to any OAM in an OAMplex where a tape request needs to be processed.

## **13.3 Sysplex terminology**

In this section, we cover some of the Sysplex terminology for your review and the additional resources you may want to read to gain additional information about this subject matter.

The following terminologies are covered:

- ▶ Sysplex Distributor (SD)
- ▶ Workload Manager (WLM)
- ▶ Virtual IP Address (VIPA)
- ▶ DB2 data sharing (group)
- ▶ OAMplex
- ▶ Cross Coupling Facility (XCF)

### ***Sysplex Distributor (SD)***

Sysplex Distributor in OS/390 V2R10 and in z/OS provides additional customer network configuration flexibility, particularly as related to network attachment costs and sharing of Open System Adapters (OSAs) among LPARs.

When a TCP connection request arrives, the Sysplex Distributor routing stack does not complete the connection itself as it usually would. Instead, WLM is consulted to find the relative available capacities on the nodes (OS/390 or z/OS) hosting the stack and application. The routing node also consults Service Policy Agent for network performance and defined policies that might affect the distribution decision. With all available relevant information, including precisely which target stacks have server applications ready for work, the routing stack selects a target stack and forwards the request to that target stack for processing.

### ***Workload Manager (WLM)***

The Workload Manager (WLM) can dynamically manage jobs that are goal-oriented by distributing them to multiple systems in a Sysplex environment, thus reducing operational demands and improving their total response time.

Performance goals are defined in business terms. Each goal is assigned a business importance. The Workload Manager decides how much resource, such as CPU and storage, should be given to the work to meet its goal. The system matches resources to the work to meet those goals, constantly monitoring and adapting processing to meet these goals.

### ***Virtual IP Address (VIPA)***

Virtual IP Address (VIPA) is an IP address that is independent of any particular network interface.

In OS/390 V2R8, the concept of a Dynamic VIPA was introduced.

Simplified configuration definitions allow Dynamic VIPAs to be activated either continuously or on demand from an application.

Other TCP/IPs in the Sysplex may also be configured as backup for a continuously active Dynamic VIPA, such that the Dynamic VIPA is automatically activated on a backup stack whenever the normally owning TCP/IP suffers an outage. This automatic Dynamic VIPA backup is called VIPA Takeover.

### ***DB2 data sharing (group)***

DB2 data sharing (group) is a collection of one or more DB2 subsystems that access shared DB2 data.

DB2 data sharing enables applications that run on more than one DB2 subsystem to read from and write to the same set of data concurrently. DB2 subsystems that share data must belong to a DB2 data sharing group, which runs on a Parallel Sysplex.

Each DB2 subsystem that belongs to a particular data sharing group is a member of that group. All members of a data sharing group use the same shared DB2 catalog and directory. Currently, the maximum number of members in a data sharing group is 32.

### ***OAMplex***

OAMplex consists of one or more instances of OAM running on systems that are part of a Parallel Sysplex. It has a one-to-one correlation to an Cross Coupling Facility (XCF) group in a Parallel Sysplex. The XCF group associated with an OAMplex is the XCF group joined by instances of OAM address spaces, running on separate systems in a Parallel Sysplex, sharing a common OAM database in a DB2 sharing group. Each instance of OAM is a member of the same XCF group. Also, the DB2 subsystems connected to these instances of OAM belong to the same DB2 data sharing group. The instances of OAM belonging to the same XCF group are the instances of OAM that are able to communicate with each other through the services of the XCF component.

### ***Cross Coupling Facility (XCF)***

Cross Coupling Facility (XCF) provides a robust set of clustering services. Coupling Facility contains the group buffer pool that represents a true shared cache. Once a member has updated a page and committed the change, the page is written to the group buffer pool.

If necessary, the CF hardware sends buffer invalidate signals to the other members that have registered interest in the page. XCF provides the services needed for this communication at a high speed. Because hardware and operating system components are the foundation of the Parallel Sysplex architecture, the performance and small amount of overhead of this type of architecture is impressive.

## **13.4 TCP/IP port sharing**

TCP/IP port sharing is a simple way of spreading a workload over multiple cloned OnDemand servers (LPARs) in one CEC by allowing all the OnDemand servers to listen on the same TCP/IP port number (see Figure 13-10 on page 304).

The SHAREPORT parameter of the PORT TCP/IP configuration statement is used to define the names of all of the OnDemand servers which may listen on a particular port. The same port number is specified in the ars.ini configuration file for each OnDemand instance. When TCP/IP receives an incoming client connection request, it selects the OnDemand server with the fewest current connections (both active and in the backlog) and routes the request to that

OnDemand server. All further requests from the same client are routed to the same OnDemand server until the connection is closed.

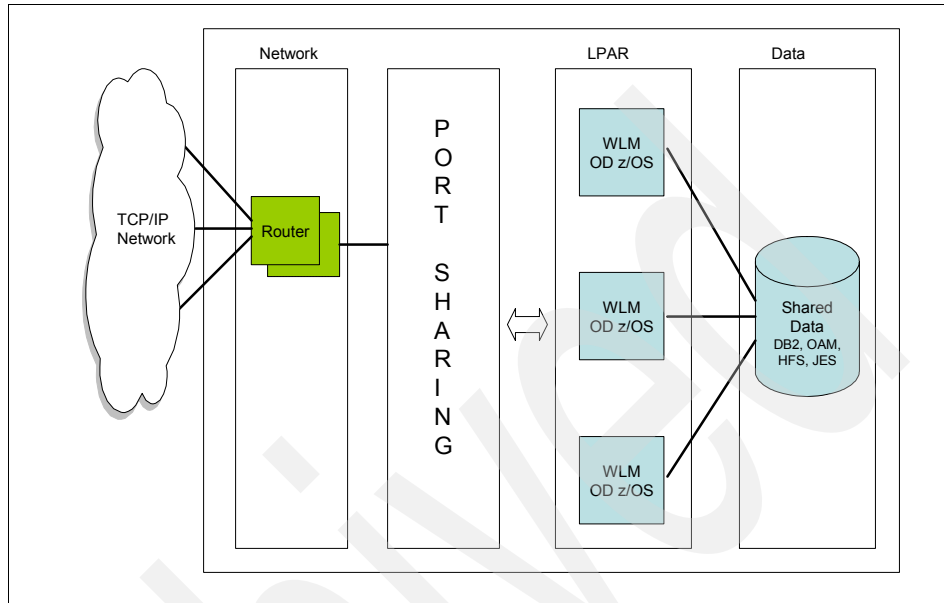


Figure 13-10 Port sharing

Port sharing is a useful technique when all the OnDemand servers are located on a single CEC. It can be used in combination with other techniques to provide a high availability environment. A Sysplex distributor environment will provide easier (more centralized) control of system resources and more sophisticated algorithms can be used in the dynamic routing program to decide the destination of a routing request; this leads to a higher degree of high availability.

## 13.5 The shared OnDemand server

Both the Sysplex distributor facilities and the port sharing facilities allow multiple instances of OnDemand to access the same database and the same report archives. This enables one to advantage of the combined OnDemand Library and Object Server functionality. This combined functionality allows for the installation of multiple OnDemand Library/Object Servers all of which access shared data (for example, DB2, OAM, JES and HFS), thus creating an OnDemand environment from which it is possible to load data from any LPAR and to retrieve data from any LPAR that may reside in one or more CECs that are defined as being part of that environment.

## Additional resources

For additional information, refer to the following manuals, Redbooks, white papers, and other publications:

- ▶ *IBM Content Manager OnDemand v7.1 for z/OS Sysplex Cookbook:*  
<http://www.ibm.com/support/search.wss?q=sysplex&tc=SSTP4L&rs=821>
- ▶ *GDPS: The Ultimate e-business Availability Solution:*  
<http://www.ibm.com/servers/eserver/zseries/library/whitepapers/gf225114>
- ▶ *Leveraging z/OS TCP/IP Dynamic VIPAs and Sysplex Distributor for higher availability DB2 Data Sharing* (white paper):  
<http://www.ibm.com/servers/eserver/zseries/library/techpapers/pdf/gm130165.pdf>
- ▶ *OS/390 and z/OS TCP/IP in the Parallel Sysplex Environment: Blurring the Boundaries* (white paper):  
<http://www.ibm.com/servers/eserver/zseries/library/techpapers/pdf/gm130026.pdf>
- ▶ *Parallel Sysplex Operational Scenarios*, SG24-2079 (redbook)
- ▶ *WebSphere for z/OS High Availability Book Update V6*, SG24-6850 (redbook)
- ▶ *Universal Database for OS/390 and z/OS Data Sharing: Planning and Administration V7*, SC26-9935
- ▶ *z/OS DFSMS Object Access Method Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426
- ▶ *z/OS Health Checker*, at:  
<http://synergy.austin.ibm.com/pdc/S390/index.html>

Archived

## Case study for OnDemand z/OS

In this chapter, we apply some of the technologies, strategies, and options, discussed earlier in this redbook, to three case studies. For each case study, we describe its background information, the backup procedures, the high availability configuration, and disaster recovery plan implemented.

The three case studies covered in this chapter include:

- ▶ International financial services company
- ▶ Communications services company
- ▶ Manufacturing company

## 14.1 International financial services company

Our first case is a study of an international financial services company.

### 14.1.1 Background

This company is a world leader in financial services with over trillions of dollars in assets under its custody. Its customers include investment managers, pension plan sponsors, collective investment fund managers, banks, corporations and not-for-profit. The company employs over tens of thousands of employees worldwide. It maintains offices in more than 25 countries covering all major investment centers, and its network spans more than 100 financial markets, facilitating its clients' investment strategies anywhere in the world.

The International financial services company has a requirement to make stock market valuation information available to their customers online as soon as the data becomes available. They also want to provide their customer service representatives with the most recent statements received by their customers, an archive repository of statements, and reports from years past. This business requirement was translated into a need to index and store an average of 500 reports per minute on a 24x7x365 basis.

They implemented OnDemand for z/OS for their company need.

### 14.1.2 Backup, recovery, and high availability approach

The international financial services company has the following backup procedure, high availability configuration, and disaster recovery plan.

#### **Backup procedure**

Remote copy to off-site DASD is done on a continuous basis. A full OnDemand database backup is performed every weekend on the Library Server and is stored in OAM. Because the full offline backup takes a snapshot of the database at a point in time, archive logs are not required to restore the database. The OnDemand application is never offline.

Because application groups are configured to store data to OAM defined nodes at load time, the cache file system is not used.

#### **High availability configuration**

This international financial services company is configured to run OnDemand by running two parallel CECs with multiple LPARs in each CEC. Dynamic VIPA, a coupling facility and a parallel sysplex WLM make all LPARs in both CECs



appear as a single system to its users. The OnDemand “shared” Library/object server runs on each of the LPARs in both of the CECs. By implementing a DB2 shared environment, an OAMplex, a shared HFS and shared JES, reports can be loaded into the OnDemand database from any LPAR and retrieved from any LPAR.

High availability is achieved at three levels. At the first level, the single CEC level, if a single LPAR were to fail, then the other LPARs would continue to load data from the shared JES and continue to store the OnDemand index data in the shared DB2 and the OnDemand report data in the OAMplex. At the second level, if a complete CEC were to fail, then the second CEC would continue to load and store the OnDemand data to the shared environment. At the third level, if any component of the data sharing environment were interrupted, then the backup environment would immediately be accessed through the remote copy facility. In all cases the stored reports are accessed by the customers on a 24x7x365 basis via a single IP address that represents the "OnDemand system". Customers are totally unaware of any operational problem with any hardware components in the system.

This company's high availability plan is visualized in Figure 14-1 on page 310. When all the LPARs on the system are healthy, the system performs at its optimum level, data is indexed and stored at the rate of 500 reports per minute. When one or more LPARs fail, the system continues to ingest data at a slower rate depending on the severity of the failure. Report loading delays may be backed up for seconds and in the worst case minutes. Regardless of the extent of the failure, client report retrieval and display service levels are achieved.

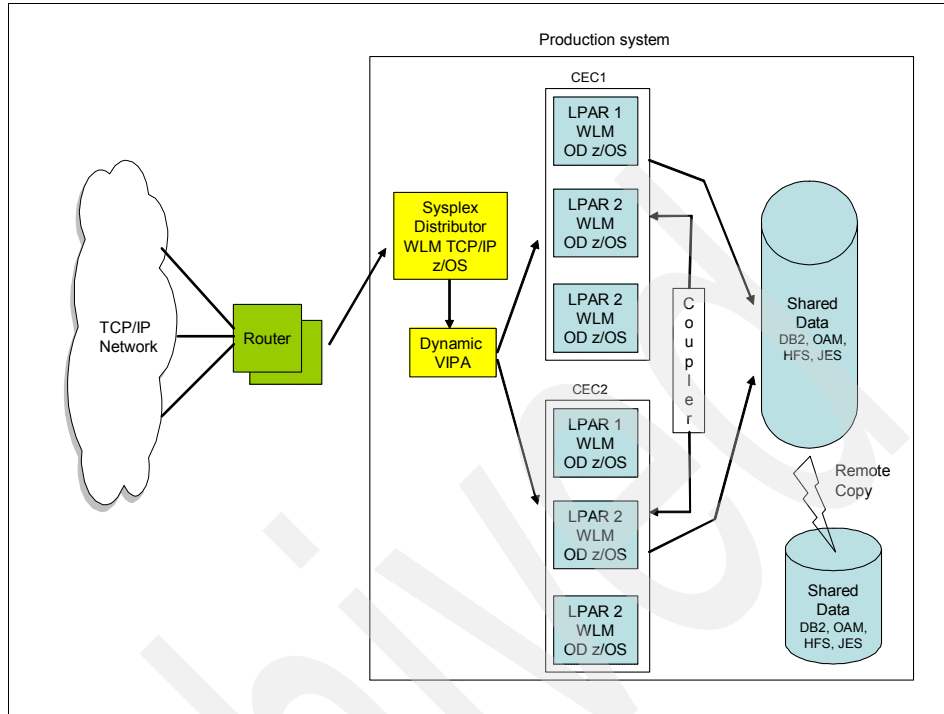


Figure 14-1 High availability configuration using a parallel sysplex

## Disaster recovery plan

The disaster recovery plan is to store backups at an off-site location. There is no mirrored site, but rather an expectation that if a catastrophic event were to occur at their data center, an identical hardware configuration could be recreated, and the backup and storage volumes could be restored.

## 14.2 Communications services company

Our second case study is that of a communications services company.

### 14.2.1 Background

The communication services company is one of the world's leading providers of communications services. It owns and operates a reliable wireless network. The company has tens of thousands of employees that serves tens of millions of voice and data customers. This company completes and maintains customer calls across U.S.

The communications services company has a requirement to make customer data available to their customers online as soon as the data becomes available. They also want to provide their customer service representatives with the most recent statements received by their customers, in addition to the archive repository of statements, and reports from previous years. This business requirement was translated into a need to index and store and retrieve reports on a continuous 24x7x365 basis.

This company had originally installed OnDemand V2 and had recently migrated to OnDemand V7. Thus they also have a requirement to retrieve both the previously archived V2 reports in addition to the newly archived V7 reports.

The company has its customer base and daily operations divided geographically into an east coast region and a west coast region. It must operate and maintain two data centers, each center serves approximately half of the total customer base.

### **14.2.2 Backup, recovery, and high availability approach**

The communications services company has the following backup procedure, high availability configuration, and disaster recovery plan.

#### **Backup procedure**

Each data center maintains a backup copy of its DASD at the other data center on DASD. This backup copy includes all the data and configuration files necessary for operating the system. There are no scheduled outages for the OnDemand application for backup purposes. A second “offline” backup is made from the backed up data and this is stored off-site.

#### **High availability configuration**

At the communications services company, each of the data centers is configured to run OnDemand by running a single CEC with multiple LPARs in each CEC. On a normal basis, each of the data centers operates independent of the other data center.

This company's high availability plan is visualized in Figure 14-2 on page 312. Within each data center, a production system uses port sharing and data sharing such that each of the multiple LPARs are accessed by the clients through a common IP address and each of the multiple LPARs has access to all of (the same) data stored by the other LPARs. The system acts as a single system. In the case of a single LPAR failure, the other LPARs will take over the workload. Because there is excess capacity in the system, the LPAR failure does not cause any degradation in performance and work progresses normally with no interruption.

If a complete CEC were to fail at one of the data centers, then work from that data center is routed to the other data center, and work would proceed normally with a degraded response time due to system capacity limitations. Each data center is capable of handling the full load but not at the same performance level that would be achieved when both data centers are operational.

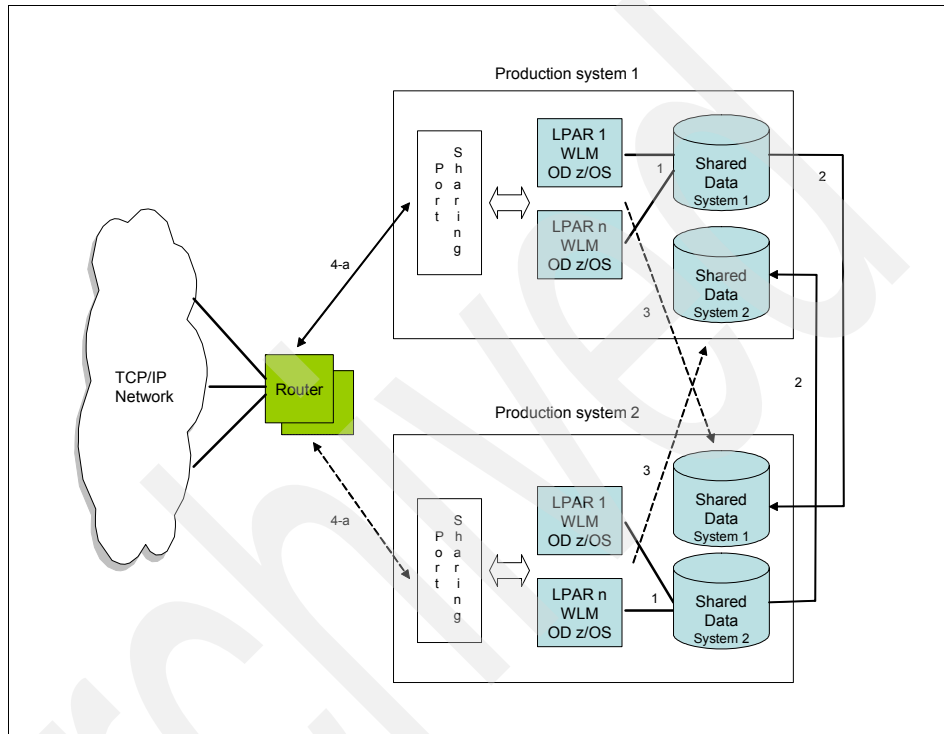


Figure 14-2 High availability configuration using port sharing

## Disaster recovery plan

The disaster recovery plan is based on the notion that a complete copy of each data centers' data is stored at the other data center. One data center is on the east coast and the other is on the west coast. The first level of disaster recovery is to use the facility of the other data center in the case of a catastrophic failure.

If both data centers were to permanently fail, then a new system would need to be built using the backup recovery tapes.

## 14.3 Manufacturing company

Our third case study is that of a Manufacturing Company.

### 14.3.1 Background

This manufacturing company is involved in the midst of an enormous growth cycle. It currently expects to double its production over the next five years.

This manufacturing company completes and maintains needs to keep track of all of its generated reports, including daily sales by location, supplier shipments, and management reports. It is also expected that the computer system will double in size in the near future. The reports stored in OnDemand need to be made available to both the manufacturers operation, sales and management personnel. In addition, a subset of the reports need to be made available to a loosely affiliated network of distributors and sales outlets. This business requirement was translated into a need to index and store and make reports available on a 24x7x365 basis.

They implemented OnDemand for z/OS for their companies need.

The company has a single data center and a single CEC that will be expanded in the future. The stored reports must be made accessible both within the company and nationwide to its suppliers and distributors. A sysplex distributor/data sharing environment was setup up so as to satisfy both the current need and the future needs. A second CEC will be installed in the future (potentially at a different physical location).

The system setup is illustrated in Figure 14-3 on page 314.

### 14.3.2 Backup, recovery, and high availability approach

The manufacturing company has the following backup procedure, high availability configuration, and disaster recovery plan.

#### **Backup procedure**

Data from the live system is replicated to a second set of DASD. The data from the second set of DASD is then copied to tape, which is stored off-site.

#### **High availability configuration**

At the manufacturing company, two LPARs are set up to share the indexing, loading and retrieval workload. The distribution of work is handled by the WLM. In the event of a single LPAR failure, the second LPAR would take on the system load. In the event of a data store failure, then the second data store would become the active data store and all data would be loaded and retrieved from it. When the first data store is brought back online, a synchronization process takes place and returns both data stores to a current state.

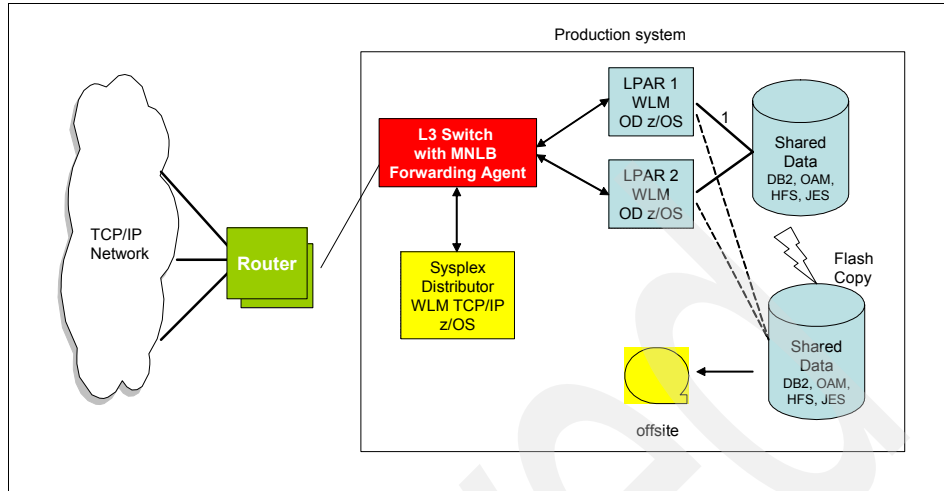


Figure 14-3 High availability configuration for a growing manufacturing company

## Disaster recovery plan

The disaster recovery plan is based on the notion that if the complete data center were destroyed, then the backup tape would be used to re-create the OnDemand (and other) systems at another location. This would involve an outage of a couple of days, but for the mean time, it is considered to be the most cost effective solution.



## Part 5

# Appendixes



Archived



## Sample scripts and programs for the high availability scenarios

This appendix provides the scripts we used during the implementation of two high availability test scenarios. We provide the source code and a detailed explanation for the following programs:

- ▶ OnDemand consolidated Library Server and Object Server startup and shutdown scripts for HACMP for a standby configuration
- ▶ OnDemand distributed Library Server and Object Server startup and shutdown scripts for HACMP for a mutual takeover configuration
- ▶ Cleanup scripts to reinitialize automated loading

**Note:** The scripts contained in this section are written for AIX only.

## A.1 HACMP standby configuration scripts

As specified in Figure 4-11, “Add an application server smit panel” on page 123, we need to provide HACMP with user-defined startup and shutdown scripts for the application we want to make highly available in order to include it in the HACMP resource group. These scripts include the automated steps specific to the application that HACMP needs to perform when moving the resource group from one node to the other.

**Note:** These scripts must be tailored for each particular installation. Your HACMP and OnDemand infrastructure specialists need to spend time working together on these scripts as a critical part of the OnDemand HACMP implementation. The scripts we used in our test environment are provided only as a *reference*.

### A.1.1 Standby configuration startup script

The Library Server startup script, as shown in Example A-1, is called by the HACMP daemons to start the main Library Server and Object Server daemon (ARSSOCKD) on a specific node after all the rest of the resources, such as IP address and disks, are already available.

In case of a failure on the primary node, the secondary node will take over the shared disks that were being used by the primary node and try to start the application using this startup script. In this case, it is most probable that the database will not be able to shut down gracefully. For this reason, it is important to make the script capable of starting a database that was previously shut down without control. It must be able to deal with leftovers such as open shared resources or temporary files. In fact, the most difficult part of the HACMP configuration for a particular component consists of identifying all the possible states of a failing application and creating an automatic recovery and startup script.

*Example: A-1 OnDemand HACMP standby startup script*

---

```
#!/usr/bin/ksh
LOG=/tmp/start_ondemand`date +%m%d%y_%I%M`~.log
DB2NODES=/home/archive/sql1lib/db2nodes.cfg
DB2NODETEMP=/tmp/db2nodes.cfg
INSTANCE_NAME=archive
INSTANCE_GRP=sysadm1

{
 echo "Removing sems./shared memory"
 su - ${INSTANCE_NAME} -c /home/${INSTANCE_NAME}/sql1lib/bin/ipclean
```

```

/usr/bin/rm /tmp/${INSTANCE_NAME}*
echo "Fixing hostname in DB2 instance configuration"
HOST=`/bin/hostname | cut -d'.' -f1`
CUR_HOST=`su - ${INSTANCE_NAME} -c cat $DB2NODES | cut -d' ' -f2`
if ["$HOST" != "$CUR_HOST"] ; then
 echo "Changing db2nodes.cfg"
 sed -e s_${CUR_HOST}_${HOST}_g $DB2NODES > $DB2NODETEMP
 cp $DB2NODETEMP $DB2NODES
 chown ${INSTANCE_NAME}:${INSTANCE_GRP} $DB2NODES
fi
echo "Starting database manager"
. /home/${INSTANCE_NAME}/sql1lib/db2profile
/usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -gkv

echo "starting OnDemand"
sleep 5
/usr/lpp/ars/bin/arssockd start ${INSTANCE_NAME}
sleep 5
/usr/lpp/ars/bin/arsload -I ${INSTANCE_NAME} -vf -t 240 -c /arsacif -d
/arsload &
} >$LOG 2>$1
exit 0

```

---

This first thing to note is that the script has almost no error checking. The error checking should be added in a real-life implementation, allowing the script to react to different scenarios. In our case, we keep it as simple as possible for clarity. Note that the script must exist with return code 0 for the failover to succeed.

The first part of the script is variable setup for easy maintenance. After that, we perform resource cleanup in case the OnDemand server shutdown was not graceful. In our environment, we only need to take care of the DB2 instance startup after a system crash. We decide that it would be enough if we run the DB2 resources cleanup program and erase any temporary file that starts with the name of the DB2 instance.

You should test your startup scripts in your environment and make sure you clean up everything in your environment at this stage.

Next, we need to adjust the host name configuration for DB2 to run in the node of the cluster that is trying to start the OnDemand server. To do this, we need to have the correct host name in the db2nodes.cfg file for the DB2 instance. In our environment, we are not changing the host name of the machine when the takeover occurs. Therefore, we need to correct this file to contain the host name of the new owner node.

Finally, we start the DB2 database manager for the instance that holds the Library Server database, the OnDemand server program ARSSOCKD, and the OnDemand ARSLOAD daemon.

We called this script start\_od.ksh and placed it in /usr/es/sbin/cluster/local in both the active and backup nodes of the cluster.

### A.1.2 Standby configuration shutdown script

The OnDemand server shutdown script, as shown in Example A-2, is called by HACMP when you bring down the OnDemand resource group for maintenance or for migration from one node to another in the cluster.

The goal of this script is to shut down, as gracefully as possible, all the applications that conform to this part of the resource group. In the case of OnDemand, this includes the DB2 database manager for the Library Server database instance and the main Library Server and Object Server daemon (ARSSOCKD).

*Example: A-2 OnDemand HACMP standby shutdown script*

---

```
#!/usr/bin/ksh

INSTANCE_NAME=archive
LOG=/tmp/stop_ondemand`date +%m%d%y_%I%M`~
{
 echo "Stopping OnDemand"
 /usr/lpp/ars/bin/arssockd stop ${INSTANCE_NAME}
 echo "Disconnecting database applications"
 su - ${INSTANCE_NAME} -c "db2 force applications all"
 sleep 10
 echo "Stopping database manager"
 /usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -hv
 RC=$?
 if ["$RC" -eq 0] ; then
 echo "database manager stopped"
 else
 echo "forcing database manager shutdown"
 su - ${INSTANCE_NAME} -c db2stop force
 fi
} >$LOG 2>$1
exit 0
```

---

Again, this script almost does not contain any error checking for simplicity and clarity.

In this script, we first shut down the OnDemand server. Then we disconnect any other applications that might be accessing the database. Finally, we shutdown the DB2 database manager.

Note that we need an exit status of 0 for successful resource group shutdown. We also must make sure that DB2 stops even if we need to force connections down or kill processes. This will leave the system in a better condition than when the system is halted at a lower level to allow resource group migration to another node.

We called this script `stop_od.ksh` and placed it in `/usr/es/sbin/cluster/local` in both the active and backup nodes of the cluster.

## A.2 HACMP mutual takeover configuration scripts

As specified in Figure 4-17, “Add an application server smit panel” on page 140, we need to provide HACMP with user-defined startup and shutdown scripts for the application we want to make highly available in order to include it in the HACMP resource group. These scripts include the automated steps specific to the application that HACMP needs to perform when moving the resource group from one node to the other. What is different about these scripts from the `start_od.ksh` and `stop_od.ksh` scripts listed in the previous section, is that these scripts need to determine which role the node is presently acting as in order to determine which role it will need to act as after the resource group is moved and the OnDemand comes back online.

**Note:** These scripts must be tailored for each particular installation. Your HACMP and OnDemand infrastructure specialists need to spend time working together on these scripts as a critical part of the OnDemand HACMP implementation. The scripts we used in our test environment are provided only as a *reference*.

### A.2.1 Mutual takeover configuration startup scripts

Start scripts are called by the HACMP daemons when a node acquires a new resource group. In our mutual takeover configuration, there are two resource groups which may be acquired, the `LibraryServerRG` and the `ObjectServerRG`. The applications servers which are part of these resource group specify the start scripts that run upon attainment.

#### Library Server start script

The Library Server start script, as shown in Example A-3 on page 322, is called by the HACMP daemons when a node acquires the `LibraryServerRG` (resource

group), after all the rest of the resources, such as IP address and disks are already available.

In case of a failure on the Library Server node, the Object Server node will take over the resource group being used by the former and try to start the application using this startup script. In this case, it is most probable that the database will not be able to shut down gracefully. For this reason, it is important to make the script capable of starting a database that was previously shut down without control. It must be able to deal with leftovers such as open shared resources or temporary files. The most difficult part of the HACMP configuration for a particular component consists of identifying all the possible states of a failing application and creating an automatic recovery and startup script.

*Example: A-3 OnDemand HACMP mutual takeover Library Server start script*

---

```
#!/usr/bin/ksh

stop the object server if it is running
OBJPID=`ps -Aef | grep "arsobjd:" | grep "accepting" | grep -v "grep" | head -1
| awk '{ print $2 }'`
kill $OBJPID > /dev/null 2>&1

if the object server was running, then it was running as a stand alone object
server
and will need to start as a library and object server (failover)
if not, start as a stand alone library server (fallback)

if [$OBJPID != ""] ; then
 cp /usr/lpp/ars/config/ars.ls_and_os.ini /usr/lpp/ars/config/ars.ini
else
 cp /usr/lpp/ars/config/ars.ls.ini /usr/lpp/ars/config/ars.ini
fi

start OD - same start procedure regardless of role in this case
LOG=/tmp/start_ondemand_libsrvr_`date +%m%d%y_%I%M`.log
INSTANCE_NAME=archive
INSTANCE_GRP=sysadm1
DB2NODES=/home/${INSTANCE_NAME}/sqllib/db2nodes.cfg
DB2NODETEMP=/tmp/db2nodes.cfg
{
 echo "Removing sems./shared memory"
 su - ${INSTANCE_NAME} -c /home/${INSTANCE_NAME}/sqllib/bin/ipclean
 echo "Fixing hostname in DB2 instance configuration"
 HOST=`/bin/hostname | cut -d'.' -f1`
 CUR_HOST=`su - ${INSTANCE_NAME} -c cat $DB2NODES | cut -d' ' -f2`
 if ["$HOST" != "$CUR_HOST"] ; then
 echo "Changing db2nodes.cfg"
 sed -e s_${CUR_HOST}_${HOST}_g $DB2NODES > $DB2NODETEMP
```

```

 cp $DB2NODETEMP $DB2NODES
 chown ${INSTANCE_NAME}:${INSTANCE_GRP} $DB2NODES
 fi
 echo "Starting database manager"
 . /home/${INSTANCE_NAME}/sql1lib/db2profile
 /usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -gkv

 echo "starting OnDemand"
 sleep 5
 /usr/lpp/ars/bin/arssockd start ${INSTANCE_NAME}
} >$LOG 2>$1
exit 0

```

---

Note, this script almost does not contain any error checking for simplicity and clarity.

The LibraryServerRG could be acquired by either of the two nodes depending on circumstances. In the case where *aristotle* is acquiring the resource group, there has been a failure or planned outage on *euclid* (failover). In the case where *euclid* is acquiring the resource group, *euclid* is returning to the cluster to assume its role as the dedicated Library Server after a failover (fallback).

In this script, we first try to determine which scenario applies for the situation. We stop the Object Server if it is running. If this node was running as a dedicated Object Server, this is a failover; otherwise, it is a fallback. The desired *ars.ini* file is put into place according to the role the node will now play. See Step 2 “Configure the OnDemand instance.” on page 129 for example *ars.ini* files.

Next, we need to adjust the host name configuration for DB2 to run in the node of the cluster that is trying to start the OnDemand Library Server. To do this, we need to have the correct host name in the *db2nodes.cfg* file for the DB2 instance. In our environment, we are not changing the host name of the machine when the takeover occurs. We need to correct this file to contain the host name of the new owner node.

Finally, we start the DB2 database manager for the instance that holds the Library Server database and the OnDemand server program ARSSOCKD. Whether acting in the role of stand alone Library Server or consolidated Library Server and Object Server, the command is the same. Optionally, configure this script to start the OnDemand ARSLOAD daemon.

We called this script *start\_ls.ksh* and placed it in */usr/es/sbin/cluster/local* in both the Library Server and Object Server nodes of the cluster.

## Object Server start script

The Object Server startup script, as shown in Example A-4 on page 324, is called by the HACMP daemons when a node acquires the ObjectServerRG (resource group). after all the rest of the resources, such as IP address and disks are already available.

In case of a failure on the Object Server node, the Library Server node will take over the shared disks that were being used by the former and try to start the application using this startup script.

*Example: A-4 OnDemand HACMP mutual takeover Object Server start script*

---

```
#!/usr/bin/ksh

LOG=/tmp/start_ondemand_objsrvr_`date +%m%d%y_%I%M`~.log
INSTANCE_NAME=archive
INSTANCE_GRP=sysadm1
DB2NODES=/home/${INSTANCE_NAME}/sql1lib/db2nodes.cfg
DB2NODETEMP=/tmp/db2nodes.cfg
LIB_SRVR_RUNNING="no"

check if the library server is already running
if yes, it is running as a stand alone library server
and will need to start as a library and object server (failover)
if no, start as a stand alone object server (fallback)

if [$(ps -ef | grep "arssockd:" | grep "accepting" | grep -v grep | wc -l) = 1
] ; then
 LIB_SRVR_RUNNING='yes'
 cp /usr/lpp/ars/config/ars.ls_and_os.ini /usr/lpp/ars/config/ars.ini
else
 cp /usr/lpp/ars/config/ars.os_only.ini /usr/lpp/ars/config/ars.ini
fi

stop OnDemand
{
 echo "stopping OnDemand"
 /usr/lpp/ars/bin/arssockd stop ${INSTANCE_NAME}
 echo "Disconnecting database applications"
 su - ${INSTANCE_NAME} -c "db2 force applications all"
 sleep 10
 echo "Stopping database manager"
 /usr/lpp/ars/bin/arsdb ${INSTANCE_NAME} -hv
 RC=$?
 if ["$RC" -eq 0] ;
 then
 echo "database manager stopped"
 else

```



```

 echo "forcing database manager shutdown"
 su - ${INSTANCE_NAME} -c db2stop force
 fi

 if [$LIB_SRVR_RUNNING='yes'] ; then
 # start ondemand as a lib/obj server

 echo "Starting database manager"
 . /home/${INSTANCE_NAME}/sqllib/db2profile
 /usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -gkv

 echo "starting OnDemand as library and object server"
 sleep 5
 /usr/lpp/ars/bin/arssockd start ${INSTANCE_NAME}
 else
 echo "Starting OnDemand object server only"
 /usr/lpp/ars/bin/arsobjd ${INSTANCE_NAME}
 fi
} >$LOG 2>$1
exit 0

```

---

Note, this script contains minimal error checking for simplicity and clarity.

The ObjectServerRG could be acquired by either of the two nodes depending on circumstances. In the case where *euclid* is acquiring the resource group, there has been a failure or planned outage on *aristotle* (failover). In the case where *aristotle* is acquiring the resource group, *aristotle* is returning to the cluster to assume its role as the dedicated object server after a failover (fallback).

In this script, we first try to determine which scenario applies for the situation. We check to see if the stand alone Library Server is running. If this node was running as a dedicated Library Server, this is a failover; otherwise, it is a fallback. The desired *ars.ini* file is put into place according to the role the node will now play. See Step 2 “Configure the OnDemand instance.” on page 129 for example *ars.ini* files.

Finally, we start the DB2 database manager for the instance that holds the Library Server database and the OnDemand server program ARSSOCKD. Or, if the stand alone Library Server daemon was not running, this must be a fallback, so start the Object Server daemon only. Optionally, configure this script to start the OnDemand ARSLOAD daemon.

We called this script *start\_os.ksh* and placed it in */usr/es/sbin/cluster/local* in both the Library Server and Object Server nodes of the cluster.

## A.2.2 Mutual takeover configuration shutdown scripts

Stop scripts are called by the HACMP daemons when the cluster service is stopped, for example, when a fallback occurs and a resource group is surrendered by a node. In our mutual takeover configuration, there are two resource groups which may be surrendered, the LibraryServerRG and the ObjectServerRG. The applications servers which are part of these resource group specify the stop scripts that run when the resource group is relinquished.

### Library Server stop script

The Library Server stop script, as shown in Example A-5, is called by the HACMP daemons when cluster services are stopped and the LibraryServerRG (resource group) is released.

The goal of this script is to shut down as gracefully as possible all the applications that conform to this part of the resource group, and, when necessary, restart OnDemand as a dedicated object server.

*Example: A-5 OnDemand HACMP mutual takeover Library Server stop script*

---

```
#!/usr/bin/ksh

LOG=/tmp/stop_ondemand_libsrvr_`date +%m%d%y_%I%M`~.log
INSTANCE_NAME=archive

{
 echo "Stopping OnDemand"
 /usr/lpp/ars/bin/arssockd stop ${INSTANCE_NAME}
 echo "Disconnecting database applications"
 su - ${INSTANCE_NAME} -c "db2 force applications all"
 sleep 10
 echo "Stopping database manager"
 /usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -hv
 RC=$?
 if ["$RC" -eq 0] ; then
 echo "database manager stopped"
 else
 echo "forcing database manager shutdown"
 su - ${INSTANCE_NAME} -c db2stop force
 fi

 # if this node
 if [[$(grep ARS_SRVR= /usr/lpp/ars/config/ars.cfg | grep -v \# | cut -d'=' -f2) != '']] ; then
 echo "Starting OnDemand object server only"
 cp /usr/lpp/ars/config/ars.os.ini /usr/lpp/ars/config/ars.ini
 fi
}
```

```

 /usr/lpp/ars/bin/arsobjd ${INSTANCE_NAME}
 else
 echo "OnDemand library server shutdown complete. No other
processes started"
 fi
} >$LOG 2>$1
exit 0

```

---

Note, this script contains minimum error checking for simplicity and clarity.

The LibraryServerRG could be released by either of the two nodes depending on circumstances. In the case where *euclid* is surrendering the resource group, there is likely a planned outage on *euclid* (failover) and the resource group is being shifted to the remaining node. In the more likely scenario, *aristotle* is surrendering the resource group. *aristotle* is returning the resource group to the cluster after *euclid* is ready to reenter the cluster and then will assume its role as the dedicated object server (fallback).

In this script, we first stop the Library Server daemon and the DB2 database instance. We then determine the role of the server based on the *ars.cfg* file. If the current *ars.ini* file points to an *ars.cfg* file that indicates this server was acting as a Library Server and Object Server (fallback), the desired *ars.ini* file is put into place and the stand alone Object Server is started. See Step 2 “Configure the OnDemand instance.” on page 129 for example *ars.ini* files. Otherwise, a planned outage is occurring and the resource group is being shifted to the remaining node (failover), so no OnDemand processes are started.

Optionally, configure this script to start the OnDemand ARSLOAD daemon.

We called this script *stop\_ls.ksh* and placed it in */usr/es/sbin/cluster/local* in both the Library Server and Object Server nodes of the cluster.

### Object Server stop script

The object server stop script, as shown in Example A-6, is called by the HACMP daemons when cluster services are stopped and the ObjectServerRG (resource group) is released.

The goal of this script is to shut down as gracefully as possible all the applications that conform to this part of the resource group, and, when necessary, restart OnDemand as a dedicated Library Server.

*Example: A-6 OnDemand HACMP mutual takeover Object Server stop script*

---

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

```
LOG=/tmp/stop_ondemand_objsrvr_`date +%m%d%y_%I%M`~.log
```

```

INSTANCE_NAME=archive
INSTANCE_GRP=sysadm1
DB2NODES=/home/${INSTANCE_NAME}/sql1lib/db2nodes.cfg
DB2NODETEMP=/tmp/db2nodes.cfg
LIB_SRVR_RUNNING="no"

if [$(ps -ef | grep "arssockd: (accepting)" | grep -v grep | wc -l) = 1]
then
 LIB_SRVR_RUNNING="yes"
fi
{
 if [["$LIB_SRVR_RUNNING" = "yes"]]
 then
 echo "Stopping OnDemand"
 /usr/lpp/ars/bin/arssockd stop ${INSTANCE_NAME}
 echo "Disconnecting database applications"
 su - ${INSTANCE_NAME} -c "db2 force applications all"
 sleep 10
 echo "Stopping database manager"
 /usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -hv
 RC=$?
 if ["$RC" -eq 0] ; then
 echo "database manager stopped"
 else
 echo "forcing database manager shutdown"
 su - ${INSTANCE_NAME} -c db2stop force
 fi

 # restart ondemand as a lib server only
 cp /usr/lpp/ars/config/ars.ls.ini /usr/lpp/ars/config/ars.ini
 echo "Starting database manager"
 . /home/${INSTANCE_NAME}/sql1lib/db2profile
 /usr/lpp/ars/bin/arsdb -I ${INSTANCE_NAME} -gkv

 echo "starting OnDemand as library server only"
 sleep 5
 /usr/lpp/ars/bin/arssockd start ${INSTANCE_NAME}
 else
 echo "Shutting down OnDemand object server"

 OBJPID=`ps -Aef | grep "arsobjd: (accepting)" | grep -v "grep" |
head -1 | awk '{ print $2 }'`
 kill $OBJPID > /dev/null 2>&1

 echo "OnDemand object server shutdown complete No other processes
started"
 fi
} >$LOG 2>$1

```

Note, this script contains minimum error checking for simplicity and clarity.

The ObjectServerRG could be released by either of the two nodes depending on circumstances. In the case where *aristotle* is surrendering the resource group, there is likely a planned outage on *aristotle* (failover) and the resource group is being shifted to the remaining node. In the more likely scenario, *euclid* is surrendering the resource group. *euclid* is returning the resource group to the node after *aristotle* is ready to reenter the cluster and then will assume its role as the dedicated Library Server (fallback).

In this script, we first determine if the surrendering node is acting as a consolidated Library Server and Object Server. If it is, the situation is a fallback, the ObjectServerRG is falling back to *aristotle*. We stop the OnDemand Library Server and Object Server on the node, stop the DB2 database instance, copy the desired *ars.ini* file into place start the dedicated Library Server daemon. See Step 2 “Configure the OnDemand instance.” on page 129 for example *ars.ini* files. Otherwise, a planned outage is occurring and the resource group is being shifted to the remaining node (failover). We stop the Object Server daemon and do not start any OnDemand processes.

Optionally, configure this script to start the OnDemand ARSLOAD daemon.

We called this script *stop\_os.ksh* and placed it in */usr/es/sbin/cluster/local* in both the Library Server and Object Server nodes of the cluster.

## A.3 OnDemand load daemon cleanup script

The OnDemand load facility can be run as a daemon process. The ARSLOAD daemon monitors a directory at an interval, checking for files with a particular naming convention. If a file is found that matches the convention in the monitored directory, ARSLOAD attempts to load the file into the OnDemand system. The load daemon typically runs on the Object Server. In our mutual takeover example, the file systems associated with the load daemon are part of the ObjectServerRG. They are */arsload* and */arsacif*. */arsload* is typically the file system being monitored by the daemon. */arsacif* is typically the file system that holds the temporary files that are created by the subprocesses of ARSLOAD.

If a failure occurs within the cluster and a failover occurs, it is possible that the load process could be interrupted. The possible results could be that some temporary files are left around that need to be “cleaned up”, or that a partial load has occurred and needs to be “cleaned up”. In either case, the system should be

set back to the state that it was in prior to the crash, and the ARSLOAD process needs to be restarted.

**Note:** We recommended that these steps be manually performed after a failover. However, if you do decide to add these steps as part of your start scripts, make sure you have thoroughly tested all scenarios specific to your environment. These scripts must be tailored for each installation. Your HACMP and OnDemand infrastructure specialists need to spend time working together on these scripts as a critical part of the OnDemand HACMP implementation. The scripts we used in our test environment are provided only as a *reference*.

Generally speaking, the ARSLOAD process can be divided into two phases, indexing and loading. The indexing phase takes a defined set of parameters and parses the file, extracting pertinent information to store in the database, gathering document resources (for Advanced Function Presentation - AFP -data), and preparing the data for storage. The indexing phase creates some temporary files used as input for the next step, the load phase. The load phase inserts rows into the database application group tables, and packages and compresses (when requested) data objects for storage to OnDemand managed cache and/or TSM.

If the ARSLOAD process is interrupted during the indexing phase, there are temporary files in the indexing directory (in our example /arsacif) named:

```
<filename>.ind
<filename>.out
<filename>.res
<filename>.parm.tmp
<filename>.tmp
```

Depending on when the process is interrupted; the <filename>.ind, <filename>.out, and <filename>.res (which are used as input to the load phase) are most likely truncated temporary files. If OnDemand tried to use the truncated files to continue to the load phase, an error would occur. All temporary indexing files should be deleted and the ARSLOAD process for this loadfile restarted.

**Note:** The exception would be if your implementation receives pre-indexed files (temporary files created elsewhere) to the monitored directories. These files would not be truncated and therefore, they would be usable to the OnDemand system. Your scripts should be modified to handle this case.

If the ARSLOAD process is interrupted during the load phase, there are temporary files in the indexing directory (/arsacif) named:

```
<filename>.ind
```

```
<filename>.out
<filename>.res
<filename>.<1FAA1>
<filename>.<1FAA1>.utf8 (possibly)
```

There may also be temporary files in the directory specified by the ARS\_TMP parameter in the ars.cfg file (in our example /arstmp) named:

```
<load_process_id>.AOD
<load_process_id>.load_id
```

In this case, a partial load has likely started and has been interrupted. If that is true, some database rows have been inserted in application group tables and some storage objects have been stored to OnDemand managed cache storage and/or TSM. To ensure a clean load with no duplicate rows or objects, the database rows should be unloaded, the storage objects removed, and the load process for this loadfile restarted. The challenge is that the load\_id necessary for use in **arsadmin unload** has probably not yet been output to the system log, and is therefore it is not available for lookup. To get this information, we need to look in the temporary files themselves. In this case, we can use an **arsadmin unload** command option -Q to force the unload of the database rows, despite the absence of a match in the system log.

To determine where we are in the ARSLOAD process, we use the presence of particular temporary files in the indexing directory. If we determine the failure occurred at a time when a partial load has likely occurred, we use the information in the temporary files in the temp directory to unload the data.

**Important:** Unloading data from OnDemand using the -Q option should be used with extreme care, as you may unload data that was not intended to be unloaded.

There is a possibility that there is more than one file involved in the ARSLOAD process at a time. Add a loop to your script to handle this condition. Example A-7 shows the load cleanup script.

*Example: A-7 OnDemand HACMP load cleanup script*

```
#!/usr/bin/ksh

LOG=/tmp/start_ondemand`date +%m%d%y_%I%M`.log
INST_NAME=archive
INDEXING_DIR=/ondemand/ha/arsacif
DWNLD_DIR=/ondemand/ha/arsload
TEMP_DIR=/ondemand/ha/arstmp
HOST=`bin/hostname | cut -d'.' -f1`
USERID=admin
```

```

PASSWD=ondemand

{
 # Find out if there is anything to cleanup.
 # There will be a load file in the download directory
 # if a load has been initiated
 if [-a ${DWNLD_DIR}/*.ARD] ; then

 ls ${DWNLD_DIR}/*.ARD | while read LOADFILE
 do
 print ${LOADFILE} | awk -F[.] '{print $1, $2, $3, $4,
$5, $6, $7}' | read MVS JOBNAME DATASET FORMS YYDDD HHMMM EXT
LOADFILE=${MVS}.${JOBNAME}.${DATASET}.${FORMS}.${YYDDD}.${HHMMM}.${EXT}
LOADFILENAME=${LOADFILE##*/}
 done

 # before we start the load daemon again
 # check indexing directory to see if indexing was interrupted
 # by looking for particular temp files
 # if indexing was interrupted, only need to delete temp
 # files and the next time the load daemon
 # checks the download directory, the load process will
 # start again on the original file

 if [-a ${INDEXING_DIR}/${LOADFILENAME}.parm.tmp] ; then

 rm ${INDEXING_DIR}/${LOADFILENAME}.ind
 rm ${INDEXING_DIR}/${LOADFILENAME}.out
 rm ${INDEXING_DIR}/${LOADFILENAME}.res
 rm ${INDEXING_DIR}/${LOADFILENAME}.parm*

 # check OnDemand temp directory to see if loading was
interrupted
 # by looking for particular temp files
 # if load phase was interrupted, we need to get all of the info
 # necessary to unload a partial load and then delete temp
files.
 # After the partial load is unloaded, the original input file
 # will be picked up by the arslload daemon

 elif [-a ${TEMP_DIR}/*.load_id] ; then

 LOADIDFILE=$(ls ${TEMP_DIR}/*.load_id)
 LOADIDFILENAME=${LOADIDFILE##*/}
 LOADPROCID=`print ${LOADIDFILENAME} | cut -d'.' -f1`
 sed "s/\>//g;s/\<//g" ${TEMP_DIR}/${LOADIDFILENAME} >
${TEMP_DIR}/temp.loadid

```



```

cat ${TEMP_DIR}/temp.loadid | grep "OnDemand Load Id" |
read A B C D E

LOADID=$E
rm ${TEMP_DIR}/temp.loadid

AGID=`print $LOADID | cut -d'-' -f1`
db2 "connect to ${INST_NAME}"
APPGRP_NAME=$(db2 "select name from arsag where agid =
$AGID")

APPGRP_NAME=`print $APPGRP_NAME | cut -d' ' -f3`

if [${FORMS} = ${APPGRP_NAME}] ; then
 /usr/lpp/ars/bin/arsadmin unload -u ${USERID}
-p ${PASSWD} -h ${HOST} -g "${APPGRP_NAME}" -L ${LOADID} -Q
 rm ${INDEXING_DIR}/${LOADFILEFILENAME}.ind*
 rm ${INDEXING_DIR}/${LOADFILENAME}.out
 rm ${INDEXING_DIR}/${LOADFILENAME}.res
 rm ${TEMP_DIR}/${LOADPROCID}.*
fi
fi

restart the arslload daemon

/usr/lpp/ars/bin/arsload -h ${HOST} -vf -t 240 -c ${INDEXING_DIR} -d
${DWNLD_DIR} &
}
>$LOG 2>$1
exit 0

```

---

Again, this script contains minimal error checking for simplicity and clarity.

This script could be inserted into the start scripts to perform cleanup prior to restarting the ARSLOAD daemon. In order to get OnDemand up and running, these steps should be performed after the OnDemand server processes have been restarted but prior to the ARSLOAD daemon starting.

If any cleanup needs to be done, check for the presence of a load file in the download directory. One of the last things the ARSLOAD process does is to delete the load file after processing. So, its presence indicates that the ARSLOAD process may have been started. It is also possible that the file had been transferred to the download directory, but the polling time has not yet elapsed and therefore processing of the file has not started. In that case, there will be no temporary files found and further processing will not take place.

If a load file has been found, we check for temporary files that indicate the indexing process was in progress when the node failed. We check against

<loadfile>.parm.tmp file because the .ind, .out, and .res files would be present throughout the entire ARSLOAD process, not just indexing. If found, delete the (likely) truncated temporary index files. The original load file will be picked up by the ARSLOAD daemon when it is restarted and the file will be processed for loading again.

Next, we check for temporary files indicating that the load phase was in progress when the load failed. The <loadfile>.load\_id temporary file is created in the ars.cfg's ARS\_TMP directory when database loading/object storage begins. If found, we extract the load ID to get the application group name from the database, and also for use in the **arsadmin unload** command. Then we feed that information to the **arsadmin unload** command with the **-Q** option. The **-Q** option is necessary because the load ID is not found in the system log - the node crashed before it could be stored. Again, use this option with extreme caution to verify that you are not unloading data that should remain. A quick check is done to verify that the application group name retrieved from the database query and the FORMS section of the <loadfile> match (see OnDemand's *Administrator's Guide*, SC27-0840 for more information about the naming convention that OnDemand supports). After the unload, the temporary files are deleted. The original load file remains to be picked up by the ARSLOAD daemon after it is restarted.

Finally, the ARSLOAD daemon is restarted.

## Sample PPRC scripts for failover

This appendix provides the sample PPRC scripts for failover setup. They should be used as reference only. Depend on your specific system requirements, appropriate changes need to be made.

The scripts we provided include:

- ▶ Example B-1 on page 336
- ▶ Example B-2 on page 338

## Sample PPRC scripts

The sample scripts provided here work with “Business continuity example 3: Multi-site primary/secondary with disk replication” on page 155. Figure 4-21 on page 156 shows the configuration setup.

There are several steps that need to be performed before failover and during a failover to the backup site.

Before a failover, you need to perform the following steps to the backup site:

1. Establish path between the primary site A and the secondary (backup) site B (PPRC\_PATH\_AtoB).
2. Establish PPRC synchronization between the primary site A and the secondary (backup) site B (PPRC\_SYNC\_AtoB).
3. Setup HACMP cluster at the primary host.
4. Assign backup site ESS volume to backup host which is part PPRC target volume.

Example B-1 shows the sample script to be executed before failover at backup site. It includes the tasks to establish the paths and their relationships between the set of disks (LUNs) from the primary site A to the secondary (backup) site B. Data will be copied and synchronized. Note, on the secondary site, the data will not be available prior to failover.

*Example: B-1 Sample script to be executed before failover at backup site*

---

```
#####
startup script
#####
echo " Initial setup at Primary Site A"
echo `date`
echo
#####
Environment
#####
CLI_DIR=/usr/opt/ibm2105cli #install directory of the cli software
COPY_SERVICES_SERVER=9.43.226.165#primary copy services server(serverA)
LOG=/var/hacmp/log
MNT=/usr/sbin/mount
UMNT=/usr/sbin/umount
VOFF=/usr/sbin/varyoffvg
VON=/usr/sbin/varyonvg
VEXP=/usr/sbin/exportvg
VIMP=/usr/sbin/importvg
TEE="/usr/bin/tee -a"
USER=hacmp# user in primary copy services server
```

```

PWD=hacmp # password for user in primary copy services server
#####
echo " Establish PPRC path between Site A and Site B"
#####
$CLI_DIR/rsExecuteTask.sh -v -s $COPY_SERVICES_SERVER PPRC_PATH_AtoB
let status=$?
if (($status == 0));
then
echo -----| $TEE $LOG
echo `date` "PPRC_PATH_AtoB path established successfully" | $TEE $LOG
echo -----| $TEE $LOG
else
echo -----| $TEE $LOG
echo `date` "ERROR: PPRC_PATH_AtoB path establish FAILED:" | $TEE $LOG
echo -----| $TEE $LOG
exit 1
fi
#####
echo " Establish PPRC Synchronous Relation between Site A and Site B"
#####
$CLI_DIR/rsExecuteTask.sh -v -s $COPY_SERVICES_SERVER PPRC_SYNC_AtoB | $TEE
$LOG 2>&1
let status=$?
if (($status == 0));
then
echo -----| $TEE $LOG
echo `date` " PPRC_SYNC_AtoB PPRC Sync established successfully" | $TEE
$LOG
echo -----| $TEE $LOG
else
echo -----| $TEE $LOG
echo `date` "ERROR: PPRC_SYNC_AtoB PPRC Sync failed" | $TEE $LOG
echo -----| $TEE $LOG
exit 1
fi
#####

```

---

The following steps should to be performed during failover to backup site provided that both sites are in PPRC synchronization:

1. Establish PPRC freeze at the primary site A (PPRC\_FRZ\_AtoB).  
Make sure the primary host running no or minimum I/O.
2. Establish PPRC consistency created at the primary site A (PPRC\_CONST\_AtoB).

If not, the primary host I/O will fail because ESS prohibit write operation due to the freeze operation.

3. Establish PPRC terminate between the primary site A and the backup site B (PPRC\_TER\_AtoB).
4. Import backup ESS volumes at backup host.
5. Mount file systems at backup host.
6. Setup HACMP cluster at backup host.
7. Valid data integrity.

Example B-2 shows the sample script to be executed after failover at backup site.

*Example: B-2 Sample script to be executed after failover at backup site*

---

```
#####
echo " Failover from Primary Site A to Backup Site B"
echo `date`
echo
#####
Environment
#####
CLI_DIR=/usr/opt/ibm2105cli#install directory of the cli software
COPY_SERVICES_SERVER=9.43.226.165#primary copy services server
LOG=/var/hacmp/log
MNT=/usr/sbin/mount
UMNT=/usr/sbin/umount
VOFF=/usr/sbin/varyoffvg
VON=/usr/sbin/varyonvg
VEXP=/usr/sbin/exportvg
VIMP=/usr/sbin/importvg
TEE="/usr/bin/tee -a"
USER=hacmp# user in primary copy services server
PWD=hacmp # password for user in primary copy services server
#####
echo " Establish PPRC Freeze between Site A and Site B"
#####
$CLI_DIR/rsExecuteTask.sh -v -s $COPY_SERVICES_SERVER PPRC_FRZ_AtoB | $TEE $LOG
2>&1
let status=$?
if (($status == 0));
then
 echo -----| $TEE $LOG
 echo `date` " PPRC_TER_AtoB PPRC Freeze established successfully" | $TEE
$LOG
 echo -----| $TEE $LOG
else
 echo -----| $TEE $LOG
 echo `date` "ERROR: PPRC_TER_AtoB PPRC Freeze failed" | $TEE $LOG
 echo -----| $TEE $LOG
 exit 1
```

```

fi
#####
echo " Establish PPRC Consistency Created between Site A and Site B"
#####
$CLI_DIR/rsExecuteTask.sh -v -s $COPY_SERVICES_SERVER PPRC_CONST_AtoB | $TEE
$LOG 2>&1
let status=$?
if (($status == 0));
then
 echo -----| $TEE $LOG
 echo `date` " PPRC_CONST_AtoB PPRC Consistency Created established
successfully" | $TEE $LOG
 echo -----| $TEE $LOG
else
 echo -----| $TEE $LOG
 echo `date` "ERROR: PPRC_CONST_AtoB PPRC Consistency Created failed" |
$TEE $LOG
 echo -----| $TEE $LOG
 exit 1
fi
#####

#####
echo " Establish PPRC Terminate between Site A and Site B"
#####
$CLI_DIR/rsExecuteTask.sh -v -s $COPY_SERVICES_SERVER PPRC_TER_AtoB | $TEE $LOG
2>&1
let status=$?
if (($status == 0));
then
 echo -----| $TEE $LOG
 echo `date` " PPRC_TER_AtoB PPRC Terminate established successfully" |
$TEE $LOG
 echo -----| $TEE $LOG
else
 echo -----| $TEE $LOG
 echo `date` "ERROR: PPRC_TER_AtoB PPRC Terminate failed" | $TEE $LOG
 echo -----| $TEE $LOG
 exit 1
fi
#####

```

---





# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

## IBM Redbooks

For information on ordering these publications, see “How to get IBM Redbooks” on page 343. Note that some of the documents referenced here may be available in softcopy only.

- ▶ *Content Manager OnDemand Guide*, SG24-6915
- ▶ *Striving for Optimal Journal Performance on DB2 Universal Database for iSeries*, SG24-6286
- ▶ *IBM TotalStorage Business Continuity Solutions Guide*, SG24-6547
- ▶ *IBM Certification Study Guide - pSeries AIX System Administration*, SG24-6191

## Other publications

These publications are also relevant as further information sources:

- ▶ *DFSMS Object Access Method Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426
- ▶ *DB2 Content Manager OnDemand for Multiplatforms V8.3 Installation and Configuration Guide*, SC18-9232
- ▶ *DB2 Content Manager OnDemand for Multiplatforms V8.3 Introduction and Planning Guide*, GC18-9236
- ▶ *Planning and Installation Guide for HACMP*, SC23-4861
- ▶ *HACMP Administration Guide*, SC23-4862
- ▶ *HACMP Concepts and Facilities*, SC23-4864
- ▶ *HACMP/XD: ESS PPRC Planning and Administration Guide*, SC23-4863
- ▶ *Tivoli Storage Manager for Windows Administrator's Guide*, GC35-0410
- ▶ *Tivoli Storage Manager for Windows Administrator's Reference*, GC35-0411

- ▶ *IBM Content Manager OnDemand for iSeries V5.3 Installation Guide*, SC41-5333
- ▶ *IBM Content Manager OnDemand for iSeries V5.3 Common Server Installation and Configuration Guide*, SC27-1158
- ▶ *IBM Content Manager OnDemand for iSeries V5.3 Common Server Administrator's Guide*, SC27-1161
- ▶ *IBM Content Manager OnDemand for iSeries V5.3 Administration Guide*, SC41-5325
- ▶ *OS/400 Integrated File System Introduction Guide*, SC41-5711
- ▶ *OS/400 Optical Support*, SC41-5310
- ▶ *IBM DB2 Content Manager OnDemand for z/OS: Web Enablement Kit Implementation Guide*, SC27-1376
- ▶ *IBM DB2 Universal Database Installation and Configuration Supplement for Version 8*, GC09-4837
- ▶ *iSeries Backup and Recovery Guide*, SC41-5304
- ▶ *DB2 UDB for z/OS Administration Guide V8*, SC18-7413
- ▶ *OS/390 MVS System Messages, Volume 2 (ASB-EWX)*, GC28-1785
- ▶ *z/OS DFSMS OAM Planning, Installation, and Storage Administration Guide for Object Support*, SC35-0426
- ▶ *DFSMS/MVS Access Method Services for ICF*, SC26-4906
- ▶ *z/OS DFSMSHsm Storage Administration Guide*, SC35-0421
- ▶ *DB2 UDB for z/OS and OS/390 V7 Administration Guide*, SC26-9931

## Online resources

These Web sites and URLs are also relevant as further information sources:

- ▶ Content Manager OnDemand information center  
<http://publib.boulder.ibm.com/infocenter/cmod83/index.jsp>
- ▶ Information for Subsystem Device Driver:  
<http://www.ibm.com/servers/storage/support/software/sdd/index.html>
- ▶ Information about iSeries navigator  
<http://www.ibm.com/servers/eserver/iseriess/navigator>
- ▶ Information about WebSphere Application Server for iSeries  
<http://www.ibm.com/servers/eserver/iseriess/software/websphere/wsappserver>

- ▶ Information about IBM HTTP Server for iSeries:  
<http://www.ibm.com/servers/eserver/series/software/http/docs/doc.htm>
- ▶ Optical device support information on iSeries:  
<http://www-03.ibm.com/servers/eserver/series/optical/>
- ▶ Information about Backup, Recovery and Media Services (BRMS) for iSeries:  
<http://www.ibm.com/servers/eserver/series/service/brms/>

## How to get IBM Redbooks

You can search for, view, or download Redbooks, Redpapers, Hints and Tips, draft publications and Additional materials, as well as order hardcopy Redbooks or CD-ROMs, at this Web site:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Help from IBM

IBM Support and downloads

[ibm.com/support](http://ibm.com/support)

IBM Global Services

[ibm.com/services](http://ibm.com/services)



# Index

## A

- ACIF 34, 240–241
- ACIF indexer 32
- active/active mode
  - business continuity 148
  - cluster configuration 91
- active/standby mode
  - business continuity 148
  - cluster configuration 91
- add
  - application server 139
- administration client 187
- administrative client program 47
- Advanced Function Presentation 28, 162
- affordable level 13
- AFP 28, 162, 244
- AFP Conversion and Indexing Facility 34
- AFP file 34
- AFP Web Viewer 246, 248
- AFP2HTML Java applet 246, 249
- AFP2PDF Transform 246, 249
- AG
  - index tables 37
- AIX 5L 181
- AIX Logical Volume Mirroring 21
- almost continuous availability 11
- annotation 28, 149
- API 246
- application 31, 35–36, 238, 242
  - definition 35
- application definition 70, 187
- application group 5, 31, 35, 39, 77, 238, 242–243
  - definition 36
  - index tables 37
  - tablespace container 103
- application group definitions 187
- application monitor
  - set up 146
- application monitoring
  - configure 145
- Application Programming Interface 246
- application programs 186
- application server
  - add 139
  - configure 123, 139
- approach
  - case study 162
- APYJRNCHG 220
- architecture
  - iSeries 172
- archive
  - iSeries 184
- archive copy group
  - TSM object 48
- archive retention period
  - report 46
- archive storage 32, 42–43, 45, 245
  - definition 43
- Archive Storage Manager 202
- archive storage manager
  - definition 34
- archive storage node 239
- archived log files 9–10
- ars.cache
  - sample 108
- ars.cfg 130
  - sample 108
- ars.dbfs 130
  - sample 108
- ars.ini 129
  - sample 107
- ars.ls.cache
  - sample 134
- ars.ls.cfg
  - sample 132
- ars.ls.dbfs
  - sample 133
- ars.ls.ini
  - sample, Library Server only role 131
- ars.ls\_and\_os.ini
  - sample, consolidated system role 131
- ars.os.cache
  - sample 134
- ars.os.cfg
  - sample 132, 136
- ars.os.dbfs
  - sample 133

- ars.os.ini
  - sample, Object Server only role 131
- ARS\_DB2\_ARCHIVE\_LOGPATH 55
- ARS\_DB2\_PRIMARY\_LOGPATH 55
- arsadmin unload 331, 334
- ARSDB 41, 55, 61, 64, 147
- arsdb 109, 116, 134
- arsdb -m 87
- arsdb -s 87
- ARSDOC 149
- ARSLoad 147, 149, 152–153, 244, 329–331, 333–334
- ARSMaint 40, 45–46, 87, 147
- arsmaint -r 87
- ARSOBJD 140
- arsobjd 136
- ARSSOCKD 61–62, 140, 318, 325
- arssockd 109, 116, 134
- arssyscr 109, 134
- ASCII data 34
- ASM 202
- ASPs 173
- assessment 4
- Asynchronous Cascading PPRC 20–21
- authorization lists 186
- automated failover 151
- auxiliary storage pools 173
- availability 11, 16
  - cost of 23–24
  - five levels of 14
  - higher levels of 16
  - highest level of 15
  - level of 95
  - measuring 16
  - optimum investment point 24
- availability techniques 16

## B

- background
  - case study 165
- backup 5, 22, 194
  - cache 52
  - cache, procedure 64
  - cache, using TSM 53
  - changed objects 205
  - data 10
  - database 53
    - procedures 60

- database log 55
- database, incremental 54
- database, offline 54
- database, offline procedure 61
- database, online 54
- database, online procedure 61
- general considerations 211
- incremental 10
- iSeries 220
- nature 9
- OnDemand 91
- OnDemand configuration 70–71
- OnDemand definition 71
- OnDemand libraries and directories 205
- online 10
- operating system 11, 52
- operating system, practical procedures 60
- optical media 205
- purpose 56
- restoration of DB2 offline 85
- restoration of DB2 online 85
- storage groups 209
- strategies and options 52
- strategy 55
- tablespace 10
- tablespace, incremental 55
- TSM 57
  - TSM database 57, 67
  - TSM, procedure 66
  - types 8
- backup and recovery plan 194
- backup and recovery strategies 51
- backup cache
  - sample script 65
- backup copies 9
- backup db 67–68
- backup devconf 68
- backup images
  - using TSM to maintain 63
- backup options
  - migration policy 207
- backup procedure
  - case study 162, 165, 308, 311, 313
- backup process 9
- backup strategy 8
- backup volhist 68
- backup window 4, 9, 56
- Backup, Recovery and Media Services 214
- backup, recovery, and high availability approach

- 165, 308, 311, 313
- base product package
  - OnDemand 32, 34
- basic systems 14
- batch-based
  - processing 26
- battery backup unit 173
- BBU 173
- BCP 17, 22
- Binary Large Objects 184
- BLOBS 184
- bookmark 236
- BPCP 17
- BRMS 214
- business continuity 22, 148
  - configuration examples 150
  - planning 24–25
- business continuity example
  - multi-site active/active 153
  - multi-site active/standby 150
  - multi-site primary/secondary with disk replication 155
- business continuity plan
  - success factor 26
- business continuity plan (BCP) 17, 22
- business process contingency plan (BPCP) 17

## C

- cache
  - backup 52
  - backup procedure 64
  - backup, using TSM 53
- cache copy
  - expires 41
- cache directory 77
  - backup 65
  - recovery 86
- cache file system storage
  - OnDemand 103
- cache storage 31, 42, 45, 52, 245, 331
  - definition 43
  - removing reports from 46
- cache storage manager 32, 46
  - definition 34
- cache storage node 239
- cached archived objects 187
- cache-only storage node 44
- CALL 213

- campus/local
  - physical distance 22
- cascading without fallback
  - resource group configuration 97
- case study 162–167, 308–314
  - background 165
  - backup procedure 162, 165, 308, 311, 313
  - backup, recovery, and HA approach 162, 165, 308, 311, 313
  - disaster recovery plan 165–166, 310, 312, 314
  - high availability configuration 163, 166, 308, 311, 313
- CCEVS 174
- CGI 245
- change
  - resource group attribute 124, 141
- changed objects
  - backup 205
- chdev 113
- check
  - cluster definition 125, 141
  - configuration 125, 141
  - resource group definition 127, 143
  - topology configuration 125, 142
- checkin libvol 82
- chmod 106
- chown 106
- chvg 114
- cleanup script
  - load daemon 329
- client node
  - TSM object 47
- client/server
  - iSeries 175
- CLSTRMGR 144
- cltopinfo 125
- cluster 90, 97
- cluster configurations 91
- cluster definition
  - check 125, 141
- cluster multi-processing 90
- cluster service
  - start 144
  - stop 144
- cluster topology and networks
  - configure 118
- clustering 173
- code page 31, 35
- collecting statistics 40

- Common Criteria Evaluation and Validation Scheme 174
- Common Gateway Interface 245
- common indexing 36
- Common Server 184, 189
- Common Server administration 188
- communication devices
  - configure 119
- complex print stream 236
- component redundancy 14–15
- components
  - involved in OnDemand installation 52
  - OnDemand 13
- concurrent users
  - case study 162
- configuration
  - check 125, 141
  - database 54
  - examples, business continuity 150
  - HACMP 137
  - network 7
- configuration file 70
  - OnDemand 103
- configuration files 9
  - environment 7
  - HACMP, verification and synchronization 127, 144
  - OnDemand 7
  - TSM 7, 66
- configuration objects 196
- configurator program
  - OnDemand 32
- configure
  - application monitoring 145
  - application server 123, 139
  - cluster topology and networks 118
  - communication devices 119
  - devices 119
  - HACMP 116
  - HACMP topology, persistent adapters 122
  - host names and order resolution 114
  - IP address 115
  - non-service adapters 120, 137
  - OnDemand instance 107, 110, 135
  - resource group 123, 141
  - serial networks 119
  - service adapter 122, 138
- container 38
- continuous availability 12
- continuous operations 12
- continuously powered main storage 173
- control tables 15
- conversion program 34
- copy
  - storage pool 69
- Copy Services Server 159
- CPM 173
- CPU consumption 220
- CPYOPT 207
- create
  - database instance 107, 109, 134
  - DB2 and OnDemand instance 107
  - file system 99, 103
  - group 106
  - JFS log 99
  - logical volume 99
  - user 106
  - users and permissions 105
  - Volume group 99
- CRM 236
- CSS 159
- CSS pair 159
- Customer Relationship Management 236
- customized programs and resources 70
- CVTOPTBKU BKUVOL 211

**D**

- DASD 173, 220
- data
  - amount to reload 56
  - code page 35
  - consistency 21
  - format 35
  - life of 40
  - loss 20
  - orientation on the page 35
  - paper size 35
  - record length 35
  - recreation 20
  - unloading 331
- data backup 194–195
- data center 25
- data failover
  - set up 156
- data indexing 240–241
- data indexing program 34
- data integrity



- maintain 10
- data load 147
- data loaded 5
- data loading 6, 32
- data loading program 35
- data loss 21
- data protection cost 24
- data queues 186
- data recovery 194
- data redundancy 14–15
- data security model 195
- data set 8
- data source 6
- data type 236
- database 38
  - backup 53
  - definition 34
  - expiration 9
  - incremental backup 54
  - loading 9
  - offline backup 54
  - online backup 54
  - size 56
- database administration 32
- database archive logs 103
- database engine 32
- database expiration 9
- database fields 243
- database instance
  - create 107, 109, 134
- database loading 9
- database log 54
  - backup 55
- database log files 6
- database management product 36
- database manager 29–30, 32, 38, 237
  - definition 34
- database primary logs 103
- database shadowing 20
- database statistics
  - run 87–88
- database utility 41
- DB2
  - creation 107
- DB2 backup 9
- db2 connect 85
- DB2 database
  - recovery 84
  - restoration to another server 86
- DB2 instance directory 103
- DB2 offline backup
  - restoration of 85
- DB2 online backup
  - restoration of 85
- DB2 UDB 174
- DB2 Universal Database 36, 174
- db2adutl 84
- db2icrt 107, 110
- db2uext2 55
- db2uext2.adsm 55, 61
- db2uext2.disk 55, 61
- define dbcopy 69
- define logcopy 70
- define stgpool 69
- delete volhist 67–68
- device class
  - TSM 48
- device name 38
- device support modules 47
- devices
  - configure 119
  - recreating in TSM 80
- df 88
- direct access storage devices 173
- directories
  - restore last backup of changed 213
  - restore last complete save of 213
- directory name 38
- disaster 7, 17, 20, 78
  - protection against 57
- disaster recovery 14–16, 25, 212
  - 7 tiers 25
  - guidelines 18
  - tier 0 19
  - tier 1 19
  - tier 2 20
  - tier 3 20
  - tier 4 20
  - tier 5 21
  - tier 6 21
  - tier 7 21
  - tier levels 18
- Disaster Recovery Manager (DRM) 20
- disaster recovery objectives 16
- disaster recovery plan 22, 194
  - case study 165–166, 310, 312, 314
- disaster recovery plan (DRP) 16–17
- disaster recovery planning

- trend in 21
- disaster recovery requirements 18
- disaster recovery server 15
- disaster recovery strategy 25
- disasters 4, 6, 12
- discover
  - HACMP-related information 118
- disk mirroring 14, 195
- disk redundancy 14–15
- disk replication 155
- disk write operation 220
- distributed system architecture 29
- distributed system configuration 240
- distributed work environment 26
- distribution objects 196
- document 39
- document retrieval request 32
- documents and folders 196
- downtime 11–12
- drive
  - TSM 48
- DRM 20
- DRP 17
- dsmfmt 69, 80
- dsmserv format 80
- dsmserv restore 80
- dsmserv.dsk 79
- dsmserv.opt 79
- dual access paths 4
- dual I/O controllers 4
- dual load 148
- dual loading
  - case study 166
- dual power supplies 4
- duplicate copy
  - create, OnDemand definition 71
- DUOPT 205, 211

**E**

- electronic information archive 236
- electronic statement presentment solution 236
- electronic vaulting 20
- enterprise level
  - eRCMF, level of management 159
- enterprise Remote Copy Management Facility 158
- Enterprise Report Management 236
- Enterprise Storage Server 90, 149
- eRCMF 20–21, 158–159

- ERP 236
- error log
  - operating system 88
- errpt 5
- eServer i5 180
- ESS 90, 149
- ESS Shark disk subsystem 96
- event logs 5
- expiration policy 40
- expiration processing 32
- expiration program 32
- expiration type 40
- expire
  - indexes 40
  - reports 45

**F**

- failover 14–15, 92, 145, 325, 329, 336
  - automated 151
- failure 4
  - hardware 11, 78
  - hardware, prevention against 57
  - software 11
  - transaction 78
- fallback 325
- FASiT family storage servers 90
- features
  - OnDemand 28
- file system 98
  - commands to create 104
  - create 99, 103
  - mount 114
  - size 103
  - unmount 111
- file system layout
  - OnDemand 98
- firmware 176
- FlashCopy 20
- FlashCopy Manager 20
- folder 31, 238, 242–243
  - definition 36
- folder definitions 187
- form definition 244
- freeze operation
  - PPRC 337
- full backup 8
- full fidelity reprinting 28
- full offline backup 8

- full online backup 8
- full system save 11
- functionalities
  - OnDemand 28
- fuser 111

## G

- GDPS/PPRC 21
- GDPS/PPRC Storage Manager 21
- GDPS/PPRC with HyperSwap 21
- GDPS/XRC 21
- Generic Indexer 32, 35
- geographic diversity 22
- GeoRM 21
- global voice and data communications company 162
- graphical user interface 173
- group
  - create 106
- GUI 173

## H

- HA BP 222
- HACMP 97–98, 115, 156
  - configuration 137
  - configuration steps 116
  - mutual takeover configuration scripts 321
- HACMP cluster 336
- HACMP configuration files
  - verification and synchronization 127, 144
- HACMP setup
  - mutual takeover configuration 137
- HACMP topology, persistent adapters
  - configure 122
- HACMP/XD 21
- HACMP-related information
  - discover 118
- hard disk 9
- hardware 176
- hardware failure 5, 78
  - prevention against 57
- hardware failures 4
- hardware resources 6
- heart beat 97
- HFS 239, 245
- Hierarchical File System 239
- high availability 5, 11–13, 16, 22, 24–25
  - definition 90

- levels 13
- multiple levels 15
- Mutual takeover site configuration 95
- practical procedures 96
- standby site configuration 93
- tiers 14
- High Availability Business Partner 222
- high availability configuration
  - case study 163, 166, 308, 311, 313
- high availability example
  - Mutual takeover configuration 93
  - standby configuration 92
- high availability implementation 15
- history logs 5
- holistic approach 18
  - solution design 26
- host name
  - configure 114
- hot site
  - backup 20
- human errors 4, 11, 76
- human factor 4

## I

- I/O 5
- i5/OS 180
- IASP 21
- IBM Tivoli Storage Manager 20
- IBM Virtualization Engine system 180
- IFS 174, 186, 217
- IFS objects 218
- importvg 113
- incremental
  - backup 10
  - database backup 54
  - tablespace backup 55
- incremental backup 8
- incremental offline backup 8
- incremental online backup 8
- index 39
  - definition 39
- index data 29, 36, 237
  - optimizing 40
  - removing 40
- index field 36
- index tables 39
  - application group 37
- index value 243

- indexer
  - OnDemand 35
- indexes
  - expire 40
  - life of 40
- indexing flexibility 236
- indexing schemes 36
- information-based
  - business model 25
- input/output processors 173
- instance 38, 238
- Integrated File System 174, 186, 217
- integration
  - iSeries 173
- international financial services company
  - case study 165
- interoperability
  - iSeries 175
- intuitive management tools 181
- IO processors 179
- IOPs 173
- IP address
  - configure 115
- IP label/address
  - definition 120
- iSeries
  - architecture 171–173, 176–181
  - client/server 175
  - installation and configuration 185
  - interoperability 175
  - object types and associated operations 178
  - object-based system 177
  - OnDemand 30, 32–33, 35, 37–39, 41, 43–44, 47, 184
  - OnDemand introduction 184
  - open standards 179
  - overview 172
  - performance 175
  - scalability 175
- iSeries IASPs 20
- iSeries IASPs with PPRC 21
- iSeries Navigator 173, 188, 214, 217–218
- iSeries OnDemand
  - introduction 184–185
  - user interface 189

## J

- JFS 101

- JFS log 101
  - create 99
- JFS2 101
- JFS2Log
  - create 101
- JFS2Log volume
  - formatting 102
- job descriptions 186
- job logs 5
- journal asynchronous approach 222
- journal changes
  - recovering 219
- journal receiver 219–220
- journaling 6, 173, 195, 216, 219
  - database 20
  - remote 220
- journals 186

## L

- libraries 196–197, 211
  - restore last full save of 213
- library
  - TSM 48
- library objects
  - restore last backup set of changed 213
- Library Server 29, 31, 33, 44, 97, 237–238
  - start script 321
  - startup script 318
  - stop script 326
- Library Server database 103
- Licensed internal code 196
- licensed program libraries 196
- life cycle management
  - policy driven 184
- line data 34
- Line Data Java applet 246, 249
- Linux 181
- load daemon
  - cleanup script 329
- load daemon monitored directory 103
- load data 32
- loading program 240–241
- log
  - OnDemand system log 88
  - operating system error log 88
  - TSM activity log 88
- log volumes
  - mirroring 69

- logform 102
- logical container 97
- logical item 243
- logical partitioning 180
- logical partitions 172–173
- logical server 31
- logical volume 98–99, 101
  - commands to create 100
- loss 24
- loss of service 11
- Lotus Domino 181
- LPAR 33, 180
- lspv 112
- lsvg 111
- LTO tape drives 165

## M

- maintenance program 147
- maintenance on OnDemand database 87
- MAN 22
- management class
  - TSM object 48
- management program 35
- manual failover 151
- measuring availability 16
- Medium area network (MAN)
  - physical distance 22
- memory 5
- metadata 15
- Microsoft Windows 181
- migrate
  - reports 45
- migration 32
- migration policies 187–188, 209
- migration policy
  - backup options 207
- migration utility 185
- mirroring
  - TSM database and log volumes 69
- mission-critical applications 11
- mkgroup 106
- mklv 102
- mksysb 60
- mkuser 106
- mkvg 99
- mount 114
- mount file system 114
- move

- resource from one node to another 145
- multiple data busses 172, 179
- multi-site active/active
  - business continuity example 153
  - disaster recovery configuration, case study 167
- multi-site active/standby
  - business continuity example 150
- multi-site primary/secondary with disk replication
  - business continuity example 155
- mutual takeover configuration
  - HACMP setup 137
  - high availability example 93
- mutual takeover site configuration
  - high availability 95

## N

- natural disasters 11
- nature of backup 9
- nature of failure 4
- netmask 116
- network interface
  - definition 120
- network name
  - definition 120
- network type
  - definition 120
- networks 9
- node 238
- node name
  - definition 120
- non-service adapter
  - configure 137
- non-service adapters
  - configure 120
- notes 236
- notify 145

## O

- OAM 32, 236, 239, 245
- Object Access Method 32, 239
- Object Data Manager 113
- Object Database Manager 113
- Object Server 30–31, 33, 43–44, 97, 237, 239
  - start script 324
  - stop script 327
- object types
  - iSeries 178
- object-based system 172

- iSeries 177
- ODM 113
- ODWEK 29–30, 149, 185, 190, 203, 245
- offline
  - database backup 54
- offline backup 8
- offsite storage 57
- OnDemand
  - base product package 32, 34
  - client 32
  - components 13
  - configurator program 32
  - distributed system 33
  - distributed system with TSM 34
  - features 28, 236
  - instance 31
  - introduction, iSeries 184
  - iSeries 184
  - Overview 29, 237
  - product options 186
  - System components 30
  - system maintenance 147
  - System tables 37
  - terminology 34
  - user interface, iSeries 189
- OnDemand client 240–241
- OnDemand configuration
  - backup 70–71
- OnDemand configuration and definition 58
- OnDemand configuration files 59
- OnDemand customized files 59
- OnDemand database
  - create 109
- OnDemand definition
  - backup 71
  - create duplicate copy 71
  - export to other server 73
  - generate summary, report 74
- OnDemand definitions 59
- OnDemand directories, iSeries 198
- OnDemand Image Web Viewer 248
- OnDemand indexer 35
- OnDemand installation 52
- OnDemand instance
  - configure 107, 110, 135
  - creation 107
  - test 109
- OnDemand instance directory
  - make a backup copy of 201
- OnDemand instance libraries 197–198
- OnDemand libraries 197
- OnDemand Navigator 189
- OnDemand object data 9
- OnDemand resources and customized programs 74
- OnDemand server
  - shutdown script 320
- OnDemand system directory
  - iSeries 199
  - make a backup copy of 200
- OnDemand system log 88
- OnDemand user directories 200
- OnDemand user information 187
- OnDemand Web Enablement Kit 30, 149, 185, 203, 245
- OnDemand Windows client 190
- online
  - database backup 54
- online backup 8, 10
- open standards 172, 179
- operating system 5
  - backup 52
  - recovery 79
- operating system backup 11
- operating system error log 88
- optical 41
- optical index database 213
- optical library 213
- optical media
  - duplicate 205
- optical media backup 205
- optical media content
  - copy 207
- optical storage devices 188
- optical storage groups 208
- optimize
  - index data 40
- optimizer
  - SQL 39
- Option 21 7, 11, 195, 212
- optional libraries 196
- Oracle 31
- order resolution
  - configure 114
- OS/390 32
- OS/390 line data 34
- OS/400 32, 172, 216
- OS/400 data security model 195

- OS/400 journaling feature 195
- OS/400 libraries 197
- OS/400 licensed program 32, 185
- OS/400 object types 186
- OS/400 objects 196
- OS/400 optional libraries 196
- outages 11–12, 14, 158
  - consequences of 23
  - minimize 23
  - unplanned 90
- output queue monitors 188
- output queues 186
- overview
  - OnDemand 29
- ownership
  - set 106

## P

- page definition 244
- paper size 35
- passwd archive 106
- password 5
- password files 7
- pax 53, 64
- PCL 236
- PCM pair 159
- PCMs 159
- PDF 184
- PDF Indexer 35
- peak volume
  - case study 162
- Peer-to Peer Remote Copy 155
- Peer-to-Peer Virtual Tape Server 20
- Peer-to-Peer VTS 21
- people-based business 26
- performance
  - iSeries 175
  - journal 221–222
- permission 5, 31
  - create 105
  - set 106
- physical disk 38
- physical distance
  - between sites 22
- physical storage device 38
- PIT 20
- planned outages 11–12, 158
- point-in-time (PIT) 20

- point-in-time copies 20
- policy 243
  - define 10
  - expiration 40
- policy domain
  - TSM object 48
- policy set
  - TSM object 48
- policy-driven life cycle management 184
- Portable Document Format 184
- PPRC 21, 155
- PPRC consistency 337
  - establish 157
- PPRC freeze 157, 337
- PPRC Migration Manager 21
- PPRC scripts 336
- PPRC state 158
- PPRC synchronization 336–337
- practical procedures
  - high availability 96
- pre-indexed files 330
- prevention
  - against hardware failure 57
- primary storage pool 58
- print stream 236
- printer 238
- private authorities 196
- problem determination 88
- processors 5, 9
- Productivity Center Machines 159
- protection
  - against disaster 57
- PTFs 197
- PVID 113

## Q

- QRDARS 197, 199
- queries 243
- query actlog 88
- query option 68
- QUSRRDARS 197

## R

- RAID 4, 14, 41, 90, 195
- RAID-5 173
- RCLOPT 213
- record length 35
- recover a system 7

- recovering
  - journal changes 219
- recovery 21–22
  - cache directory 86
  - data 6
  - DB2 database 84
  - disaster 212
  - iSeries 220
  - operating system 79
  - speed of 7
  - time 6
- recovery plan 75
- recovery point 10
- recovery point objective (RPO) 10, 24
- recovery points 4
- recovery procedure 78
  - TSM 79
- recovery procedures
  - document 7
  - prioritize 7
- recovery process 11
- recovery time 7
- recovery time objective (RTO) 6, 24
- Redbooks Web site 343
  - Contact us xviii
- redundancy 14
  - disk, component, data, site 14
- Redundant Arrays of Independent Disks 90
- redundant power supplies 195
- regulatory compliance 17
- reload
  - amount of data to 56
- remote journal 220
  - synchronous mode 220
- remote journaling 221
  - overview 221
- remove
  - index data 40
- removing reports 45
  - from cache storage 46
- replication 14–15
  - annotation 149
- report 32, 36, 42, 236, 242–243
  - archive retention period 46
  - backup copies of 43
  - expire 45
- report index information 187
- report migration 45
- report wizard 187

- reproduce
  - data 6
- resource 244
  - move from one node to another 145
- resource group 97
  - configuration 97
  - configure 123, 141
- resource group attribute
  - change 124, 141
- resource group definition
  - check 127, 143
- resource virtualization 181
- restoration 10
  - DB2 database to another server 86
  - DB2 offline backup 85
  - DB2 online backup 85
  - online backup 85
- restore
  - last backup of changed directories 213
  - last backup set of changed libraries objects 213
  - last complete save of directories 213
  - last full save of libraries 213
  - tablespace level 11
- restore stgpool 82
- restored database 10
- restoring
  - data 5
  - TSM configurations and database 79
- restrict access 5
- retrieve 147
- Retrieves data 236
- rm 110
- rolling-out transactions 6
- root access 5
- RPO 10, 24
- RS/232 cable 97
- RST 213
- RSTLIB 212–213
- RSTOBJ 213
- RTO 6, 24
- rue availability 16
- runstat 9

## S

- SAN switch 93
- SAV 196, 206, 211
- SAV DEV 200
- SAVCFG 196



- SAVDLO 196
- SAVLIB 196, 212
- SAVLIB LIB 197–198
- SAVOBJ 219
- SAVSECDTA 196
- SAVSYS 196
- scalability
  - iSeries 175
- scheduled events 11
- script
  - HACMP standby configuration scripts 318
  - Library Server start script 321
  - Library Server startup script 318
  - Library Server stop script 326
  - mutual takeover configuration scripts 321
  - Object Server start script 324
  - Object Server stop script 327
  - OnDemand load daemon cleanup script 329
  - OnDemand server shutdown script 320
  - sample PPRC scripts for failover 335
- SCSI 93
- SCSI disks 90
- SDD 113
- search 147
- SEC 22
- Securities and Exchange Commission (SEC) 22
- security 31, 236
- security breaches 11
- segment table 37
- separate I/O processors 172
- serial networks
  - configure 119
- Serial Storage Architecture 90
- server control information 29, 237
- server program 47
- service adapter 115
  - configure 122, 138
- set
  - ownership and permissions 106
- set up
  - application monitor 146
- seven tiers
  - disaster recovery 18
- shared external disk device 90
- shared volume group 98
- shortcut 236
- shutdown script
  - OnDemand server 320
- single point of failure 216
- single point of failure (SPOF) 11, 15
- Single-level storage 178
- single-level storage 172
- site redundancy 14
- SLS 178
- smit 99
- smit clstart 144
- smit clstop 144
- smit crjfs2lvstd 103
- smit mklv 100–101
- smitty hacmp 117
- SMP 173
- SMS 37
- SMS tablespace 38
- space allocation 88
- SPOF 11, 15, 216
- Spool File Archive 185, 189
- spool file data
  - access 187
- spooled files 184
- SQL optimizer 39
- SQL Server 31
- SSA 90
- standby adapter 116
- standby configuration
  - cluster 91
  - high availability example 92
- standby site configuration
  - high availability 93
- start script
  - Library Server 321
  - Object Server 324
- startup script
  - Library Server 318
- state
  - database 6
- statement 243
- statistics
  - collecting 40
- stop script
  - Library Server 326
  - Object Server 327
- storage
  - archive storage 43
  - cache and archive 245
  - cache storage 43
- storage area 52
- storage devices 5, 47
- storage devices and media

- TSM 48
- storage groups
  - backup 209–210
- storage libraries 9, 47
- storage management 18
  - objects 42
  - policy 42
- storage management attribute 36
- storage management information 36
- storage management objects 42
- storage management overview 42
- storage management policy
  - TSM 47
- storage manager 29–30, 41, 237
- storage manager libraries 9
- storage node 31, 238, 244–245
  - cache-only 44
  - definition 44
- storage nodes
  - collection of 44
- storage of data
  - distribute 31
- storage pool 58
  - copy 69
- storage pools 5
- storage pools and volumes
  - TSM 48
- storage pools volume
  - restore into TSM 82
- storage set 31, 238, 244
  - definition 44
- storage space 155
- storage volumes 188, 241
- strategy
  - backup 8, 55
- STRJRN 219
- STRJRNOBJ 219
- STRJRNPf 218
- STRMONOND 213
- structured data type 236
- subnet 116
- Subsystem Device Driver 113
- symmetric multiprocessing 173
- synchronization
  - online backup 9
  - PPRC 336
- synchronous mode
  - remote journal 220
- system 38

- system administrator 5
- system architecture
  - OnDemand 29
- system availability 16
- system available time 56
- system catalog tables
  - definition 39
- system components
  - OnDemand 30
- system configuration data 199
- system configuration files 7
- system directory
  - OnDemand
    - iSeries 199
- system log 32, 35
  - OnDemand 88
- system logs 5
- system maintenance 195
  - OnDemand 147
- System Managed Space 37
- system save 11, 195
- system tables
  - OnDemand 37
- system tuning 173
- system upgrade 75
- system uptime 90
- systems management procedures 16

## T

- table 37–38
- table structure 243
- tablespace 10, 37–38
- tablespace backup 10
  - incremental 55
- tablespace level restore 11
- takeover 97
- takeover configuration
  - cluster 91
- tape 41
- tape library 9
- tape media 219
- tar 53, 64
- TCP/IP protocol 29, 237
- Technology 177
- Technology Independent Machine Interface 172, 176
- temporary files 330, 333
- temporary index file location directory 103

- temporary storage volume 240–241
- terminal IO 179
- terminology
  - OnDemand 34
- thumbnail 236
- time for recovery 6
- TIMI 172, 176
- Tivoli Storage Manager 32, 46, 93–94, 239
- topology configuration
  - check 125, 142
- total availability 16
- transaction failure 4, 6, 78
- transaction integrity 21
- transaction-based
  - processing 26
- transactions 10
- TSM 5, 7, 9, 21, 32–34, 44, 46, 93–94, 236, 239, 245, 331
  - backing up, for disaster recovery 58
  - backup 57
  - backup procedure 66
  - maintain backup images 63
  - manage cache backup 53
  - recovery procedure 79
  - recreating devices 80
  - restore storage pool volumes 82
  - restoring cache from 87
  - storage devices and media 48
  - storage management objects 47
  - storage management policy 47
- TSM activity log 88
- TSM API 47
- TSM backup commands 7
- TSM configuration
  - restoring 79
- TSM configuration files 66
- TSM copy storage pools
  - case study 162
- TSM database
  - backup 57, 67
  - mirroring 69
  - restore command 80
  - restoring 79
- TSM device class 57
- TSM file device class 57
- TSM libraries 9, 58
- twin-tail connection 93
- two-for-one standby
  - high availability configuration, case study 164

- two-part operating system
  - iSeries feature 172
- two-phase commit 21

## U

- umount 110
- unavailability
  - cost of 23
- unforeseen events 11
- uninterruptable power supply 4, 173
- Uninterrupted Power Supply 195
- unloading data 331
- unmount file system 111
- unplanned downtime 12
- unplanned outages 11–12, 90, 158
- unstructured data type 236
- update vol 82
- UPS 173, 195
- user 31, 238
  - create 105–106
- user expectations 7
- user group 31, 238
- user ID 5
- user libraries 196
- user objects 196
- user profiles 186, 196
- user spaces 186
- user table 39
- USERSPACE1 37
- utility
  - database 41

## V

- varyoffvg 111
- VG
  - create 99
- view 39
- Virtual Storage Access Method 239
- volume group 101
  - create 99
  - deactivate 111
- volume group definition
  - import 112
- volumeSet level
  - eRCMF, level of management 159
- VSAM 32, 236, 239, 245

## **W**

WAN 22

WebSphere 181

Wide Area Network (WAN)  
    physical distance 22

## **X**

Xerox 236

XRC 21

## **Y**

year-2000 bug 17

## **Z**

zFS 239, 245



## Content Manager OnDemand Backup, Recovery, and High Availability

(0.5" spine)  
0.475" <-> 0.875"  
250 <-> 459 pages







# Content Manager OnDemand Backup, Recovery, and High Availability



**Redbooks**

**Introducing basic  
concepts, strategies,  
options, and  
procedures**

**Covering  
multiplatforms,  
iSeries, and z/OS**

**Including real world  
case studies**

This IBM Redbook helps you understand backup, recovery, high availability, business continuity strategies, and options available for IBM DB2 Content Manager OnDemand. We begin with an introduction of the basic concepts of backup and recovery, high availability, disaster recovery, and business continuity. We also provide an overview of IBM DB2 Content Manager OnDemand.

Because OnDemand is available on multiplatforms, iSeries, and z/OS, we address each platform separately, and discuss the backup and recovery strategies and options for each platform. In addition, we discuss various high availability and business continuity strategies and options. When applicable, we provide practical procedures and steps to accomplish backup, recovery, and high availability with sample commands and scripts. In some instances, case studies are presented to show you how real-world businesses implement backup procedures, high availability configurations, and disaster recovery plans.

This redbook is intended for IT architects, IT specialists, and OnDemand system administrators who are responsible for designing, implementing, and maintaining OnDemand systems for various platforms.

**INTERNATIONAL  
TECHNICAL  
SUPPORT  
ORGANIZATION**

**BUILDING TECHNICAL  
INFORMATION BASED ON  
PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**  
[ibm.com/redbooks](http://ibm.com/redbooks)