

# IBM System Storage DS8000 Host Attachment and Interoperability

Learn how to attach DS8000 to open systems, IBM System z, and IBM i

See how to gain maximum availability with multipathing

Discover best practices and considerations for SAN boot



Axel Westphal  
Bertrand Dufrasne  
Juan Brandenburg  
Jana Jamsek  
Kai Jehnen  
Steven Joseph  
Massimo Olivieri  
Ulrich Rendels  
Mario Rodriguez

**Redbooks**





International Technical Support Organization

**IBM System Storage DS8000: Host Attachment and Interoperability**

February 2013

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

**Second Edition (February 2013)**

This edition applies to the IBM System Storage DS8000 with licensed machine code (LMC) level 6.6.xxx.xx (bundle version 86.x.xxx.xx) and LMC level 7.6.xxx.xx.

© Copyright International Business Machines Corporation 2011, 2013. 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

<b>Notices</b> .....	ix
Trademarks .....	x
<b>Preface</b> .....	xi
The team who wrote this book .....	xi
Now you can become a published author, too! .....	xiii
Comments welcome .....	xiii
Stay connected to IBM Redbooks .....	xiii
<b>Chapter 1. General considerations</b> .....	1
1.1 DS8000 general topics .....	2
1.2 Client architecture overview .....	3
1.3 SAN considerations .....	3
1.4 Storage tiers .....	3
<b>Chapter 2. Open systems considerations</b> .....	5
2.1 Configuration resources .....	6
2.1.1 IBM resources .....	6
2.1.2 HBA vendor resources .....	7
2.2 Using the DS8000 as a boot device .....	8
2.2.1 Configuring the QLogic BIOS to boot from a DS8000 volume .....	9
2.2.2 Next steps .....	13
2.3 Additional supported configurations .....	13
2.4 Multipathing support for Subsystem Device Drivers .....	14
2.4.1 Subsystem Device Driver .....	14
2.4.2 Other multipathing solutions .....	15
<b>Chapter 3. Windows considerations</b> .....	17
3.1 Attaching HBAs in Windows .....	18
3.1.1 HBA and operating system settings .....	18
3.1.2 SDD versus SDDDSM Multipath Drivers .....	18
3.2 Installing SDD in Windows .....	18
3.2.1 SDD datapath query .....	20
3.2.2 Mapping SDD devices to Windows drive letters .....	21
3.3 Clustering for Windows 2003 Server .....	21
3.3.1 References .....	22
3.3.2 SDD support .....	22
3.4 Using Multipath Input/Output for Windows 2003 and 2008 .....	22
3.5 Partition alignment .....	23
3.6 Installing and configuring SDDDSM in Windows 2003, 2008 .....	23
3.6.1 SDDDSM for DS8000 .....	23
3.6.2 SDDDSM datapath query .....	25
3.6.3 Windows 2008 and SDDDSM .....	26
3.7 Expanding dynamic disk for Windows 2003, Windows 2008 .....	27
3.8 SAN boot support .....	32
3.9 Windows Server 2003 Virtual Disk Service support .....	32
3.9.1 VDS integration with DS8000 storage subsystems .....	33
3.9.2 Volume Shadow Copy Service .....	33
3.9.3 Required components .....	34

3.10 Hyper-V considerations . . . . .	36
3.10.1 Hyper-V introduction . . . . .	36
3.10.2 Storage concepts for virtual machines . . . . .	37
3.10.3 Assigning a VHD to a virtual machine . . . . .	39
3.10.4 Assigning a pass-through disk to a virtual machine . . . . .	42
3.10.5 Cluster Shared Volume (CSV) . . . . .	44
3.10.6 Best practices . . . . .	45
<b>Chapter 4. Virtual I/O Server considerations . . . . .</b>	<b>47</b>
4.1 Working with IBM Virtual I/O Server . . . . .	48
4.2 Using VSCSI with IBM VIOS and DS8000 . . . . .	49
4.3 Using NPIV with IBM VIOS and DS8000 . . . . .	58
<b>Chapter 5. AIX considerations . . . . .</b>	<b>61</b>
5.1 Attaching native Fibre Channel . . . . .	62
5.1.1 Assigning volumes . . . . .	62
5.1.2 Using node port ID virtualization (NPIV) . . . . .	64
5.2 Attaching virtual SCSI . . . . .	65
5.3 Important additional considerations . . . . .	68
5.3.1 Queue depth tuning . . . . .	69
5.3.2 timeout_policy attribute . . . . .	70
5.3.3 max_xfer_size attribute . . . . .	71
5.3.4 Storage unit with multiple IBM Power Systems hosts running AIX . . . . .	71
5.4 Multipathing with AIX . . . . .	72
5.4.1 SDD for AIX . . . . .	72
5.4.2 SDDPCM for AIX . . . . .	74
5.5 Configuring LVM . . . . .	76
5.5.1 LVM striping . . . . .	76
5.5.2 Inter-physical volume allocation policy . . . . .	77
5.5.3 LVM mirroring . . . . .	77
5.5.4 Impact of DS8000 storage pool striping . . . . .	77
5.6 Using AIX access methods for I/O . . . . .	77
5.6.1 Synchronous I/O . . . . .	77
5.6.2 Asynchronous I/O . . . . .	78
5.6.3 Concurrent I/O . . . . .	78
5.6.4 Direct I/O . . . . .	78
5.7 Expanding dynamic volume with AIX . . . . .	79
5.8 SAN boot support . . . . .	80
<b>Chapter 6. Linux considerations . . . . .</b>	<b>81</b>
6.1 Working with Linux and DS8000 . . . . .	82
6.1.1 How Linux differs from other operating systems . . . . .	82
6.1.2 Attaching Linux server to DS8000 resources . . . . .	82
6.1.3 Understanding storage related improvements to Linux . . . . .	85
6.2 Attaching to a basic host . . . . .	86
6.2.1 Platform-specific information . . . . .	86
6.2.2 Configuring Fibre Channel attachment . . . . .	89
6.2.3 Determining the WWPN of installed HBAs . . . . .	93
6.2.4 Checking attached volumes . . . . .	93
6.2.5 Linux SCSI addressing . . . . .	94
6.2.6 Identifying DS8000 devices . . . . .	95
6.2.7 Adding DS8000 volumes to Linux on System z . . . . .	96
6.2.8 Setting up device mapper multipathing . . . . .	98
6.2.9 Attaching DS8000: Considerations . . . . .	104

6.2.10	Understanding non-disruptive actions on attached hosts	106
6.2.11	Adding new DS8000 host ports to Linux on System z	108
6.3	Resizing DS8000 volumes dynamically	109
6.4	Using FlashCopy and remote replication targets	111
6.4.1	Using a file system residing on a DS8000 volume	111
6.4.2	Using a file system residing in a logical volume managed by LVM	112
6.5	Troubleshooting and monitoring	114
6.5.1	Checking SCSI devices alternate methods	115
6.5.2	Monitoring performance with the iostat command	116
6.5.3	Working with generic SCSI tools	116
6.5.4	Booting Linux from DS8000 volumes	116
6.5.5	Configuring the QLogic BIOS to boot from a DS8000 volume	117
6.5.6	Understanding the OS loader considerations for other platforms	118
6.5.7	Installing SLES11 SP1 on a DS8000 volume	119
6.5.8	Installing with YAST	121
<b>Chapter 7.</b>	<b>VMware vSphere considerations</b>	<b>123</b>
7.1	vSphere introduction	124
7.2	vSphere storage concepts	125
7.2.1	VMFS	125
7.2.2	Virtual disks	135
7.2.3	Raw device mappings (RDMs)	138
7.3	Multipathing	142
7.3.1	Pluggable Storage Architecture	142
7.3.2	Path naming	143
7.3.3	Multipathing and DS8000	144
7.4	Storage vMotion	146
7.4.1	Steps to migrate a VM from one datastore to another	146
7.4.2	Limitations	146
7.5	Using SSD volumes as cache	147
7.6	Best practices	148
7.6.1	Datastores	148
7.6.2	Multipathing	149
<b>Chapter 8.</b>	<b>Apple considerations</b>	<b>151</b>
8.1	Available resources	152
8.2	Configuring the Apple host on a DS8000	152
8.3	Installing the ATTO software	152
8.3.1	ATTO OS X driver installation	152
8.3.2	ATTO Configuration Tool Installation and Flash Update	153
8.4	Using the ATTO Configuration Tool	153
8.4.1	Paths	154
8.4.2	Target Base	155
8.4.3	LUN Base	155
8.4.4	Path Actions	155
8.4.5	Host bus adapter configuration	156
8.5	Creating a file system on Apple Mac OS X	156
8.5.1	Using the GUI: Disk Utility	157
8.5.2	Using the CLI: diskutil	157
8.6	Troubleshooting	157
8.6.1	Useful Utilities on Apple Mac OSX	157
8.6.2	Troubleshooting checklist	158
<b>Chapter 9.</b>	<b>Solaris considerations</b>	<b>159</b>

9.1 Working with Oracle Solaris and DS8000 . . . . .	160
9.2 Locating the WWPNs of your HBAs . . . . .	160
9.3 Attaching Solaris to DS8000 . . . . .	161
9.4 Multipathing in Solaris . . . . .	162
9.4.1 Working with IBM System Storage Multipath SDD . . . . .	163
9.4.2 Using MPxIO . . . . .	164
9.4.3 Working with VERITAS Volume Manager dynamic multipathing . . . . .	165
9.5 Expanding dynamic volume with VxVM and DMP . . . . .	167
9.6 Booting from SAN . . . . .	172
9.6.1 Displaying the boot code . . . . .	173
9.6.2 Booting off a DS8000 LUN with Solaris . . . . .	174
9.6.3 Supplying the VID or PID string . . . . .	175
9.6.4 Associating the MPxIO device file and underlying paths . . . . .	179
<b>Chapter 10. HP-UX considerations . . . . .</b>	<b>181</b>
10.1 Working with HP-UX . . . . .	182
10.1.1 The agile view . . . . .	182
10.1.2 Notes on multipathing . . . . .	182
10.1.3 Notes on naming conventions . . . . .	183
10.1.4 Notes on enumeration . . . . .	183
10.2 Available resources . . . . .	183
10.3 Identifying available HBAs . . . . .	184
10.4 Identifying WWPNs of HBAs . . . . .	184
10.5 Configuring the HP-UX host for the DS8000 . . . . .	185
10.5.1 Device special files . . . . .	185
10.5.2 Device discovery . . . . .	185
10.6 Multipathing . . . . .	187
10.6.1 HP-UX multipathing solutions . . . . .	187
10.6.2 Exposing link errors with HP-UX . . . . .	190
10.7 Working with VERITAS Volume Manager on HP-UX . . . . .	191
10.8 Working with LUNs . . . . .	194
10.8.1 Expanding LUNs . . . . .	194
10.8.2 Working with large LUNs . . . . .	197
<b>Chapter 11. IBM i considerations . . . . .</b>	<b>201</b>
11.1 Supported environment . . . . .	202
11.1.1 System hardware . . . . .	202
11.1.2 Software . . . . .	202
11.1.3 Overview of hardware and software requirements . . . . .	203
11.1.4 Useful websites . . . . .	204
11.2 Using Fibre Channel adapters . . . . .	205
11.2.1 Native attachment . . . . .	205
11.2.2 Attachment with VIOS NPIV . . . . .	206
11.2.3 Attachment with VIOS VSCSI . . . . .	206
11.2.4 Overview of the number of LUNs per adapter and the queue depth . . . . .	207
11.3 Sizing and implementation guidelines . . . . .	207
11.3.1 Sizing for natively connected and VIOS connected DS8000 . . . . .	208
11.3.2 Planning for arrays and DDMs . . . . .	208
11.3.3 Cache . . . . .	208
11.3.4 Number of ranks . . . . .	209
11.3.5 Sizing for SSD and hot data management with IBM i . . . . .	210
11.3.6 Number of Fibre Channel adapters in IBM i and in VIOS . . . . .	213
11.4 Sizing and numbering of LUNs . . . . .	215



11.4.1	Logical volume sizes . . . . .	215
11.4.2	Sharing or dedicating ranks for an IBM i workload . . . . .	216
11.4.3	Connecting using SAN switches . . . . .	217
11.5	Using multipath . . . . .	217
11.5.1	Avoiding single points of failure . . . . .	218
11.5.2	Configuring the multipath . . . . .	218
11.5.3	Multipathing rules for multiple System i hosts or partitions . . . . .	219
11.6	Configuration guidelines . . . . .	220
11.6.1	Creating extent pools for IBM i LUNs . . . . .	220
11.6.2	Defining the LUNs for IBM i . . . . .	220
11.6.3	Protected versus unprotected volumes . . . . .	221
11.6.4	Changing LUN protection . . . . .	222
11.6.5	Setting the ports and defining host connections for IBM i . . . . .	222
11.6.6	Zoning the switches . . . . .	223
11.6.7	Setting up VIOS . . . . .	223
11.6.8	Adding volumes to the System i configuration . . . . .	224
11.6.9	Using the 5250 interface . . . . .	225
11.6.10	Adding volumes to an independent auxiliary storage pool . . . . .	228
11.7	Bootting from SAN . . . . .	235
11.7.1	Requirements for boot from SAN . . . . .	235
11.7.2	Tagging the load source LUN . . . . .	235
11.8	Installing IBM i with boot from SAN through VIOS NPIV . . . . .	238
11.8.1	Scenario configuration . . . . .	238
11.8.2	Bootting from SAN and cloning . . . . .	243
11.9	Migrating . . . . .	244
11.9.1	Metro Mirror and Global Copy . . . . .	244
11.9.2	IBM i data migration . . . . .	244
<b>Chapter 12.</b>	<b>IBM System z considerations . . . . .</b>	<b>247</b>
12.1	Connectivity considerations . . . . .	248
12.1.1	FICON . . . . .	248
12.1.2	LINUX FCP connectivity . . . . .	248
12.2	Operating system prerequisites and enhancements . . . . .	248
12.3	Considerations for z/OS . . . . .	249
12.3.1	Program enhancements for z/OS . . . . .	249
12.3.2	DS8000 device definition . . . . .	250
12.3.3	zDAC - z/OS FICON Discovery and Auto-Configuration feature . . . . .	251
12.3.4	Performance statistics . . . . .	253
12.3.5	Resource Measurement Facility . . . . .	253
12.3.6	System Management Facilities . . . . .	254
12.4	Extended Address Volume (EAV) support . . . . .	254
12.4.1	Identifying an EAV . . . . .	257
12.4.2	z/OS prerequisites for EAV . . . . .	259
12.4.3	EAV migration considerations . . . . .	260
12.5	FICON specifics for a z/OS environment . . . . .	261
12.5.1	Overview . . . . .	261
12.5.2	Parallel access volumes (PAV) definition . . . . .	262
12.5.3	HyperPAV z/OS support and implementation . . . . .	264
12.5.4	Extended Distance FICON . . . . .	268
12.5.5	High Performance FICON for System z with multitrack support (zHPF) . . . . .	269
12.5.6	zHPF latest enhancements . . . . .	270
12.6	z/VM considerations . . . . .	273
12.6.1	Connectivity . . . . .	273

12.6.2 Supported DASD types and LUNs .....	273
12.6.3 PAV and HyperPAV z/VM support .....	273
12.6.4 Missing-interrupt handler. ....	274
12.7 VSE/ESA and z/VSE considerations. ....	274
12.8 I/O Priority Manager for z/OS .....	275
12.8.1 Performance groups .....	275
12.8.2 Easy Tier and I/O Priority Manager coexistence. ....	276
12.9 TPC-R V5.1 in a z/OS environment .....	278
12.9.1 Tivoli Storage Productivity Center for Replication for System z (TPC-R) .....	278
12.9.2 References .....	279
12.10 Full Disk Encryption (FDE) .....	280
<b>Chapter 13. IBM SAN Volume Controller considerations .....</b>	<b>281</b>
13.1 IBM System Storage SAN Volume Controller. ....	282
13.2 SAN Volume Controller multipathing. ....	282
13.3 Configuration guidelines for SVC .....	282
13.3.1 Determining the number of controller ports for DS8000 .....	282
13.3.2 Logical volume considerations .....	283
13.3.3 LUN masking .....	283
13.3.4 Double striping issue. ....	284
13.3.5 SVC publications. ....	285
<b>Related publications .....</b>	<b>287</b>
IBM Redbooks publications .....	287
Other publications .....	287
Online resources .....	288
Help from IBM .....	288

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

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

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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

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


## COPYRIGHT LICENSE:

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

# Trademarks

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. These and other IBM trademarked terms are marked on their first occurrence in this information with the appropriate symbol (® or ™), indicating US registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at <http://www.ibm.com/legal/copytrade.shtml>

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

AIX®	Lotus®	Storwize®
BladeCenter®	MVS™	System i®
CICS®	Notes®	System p®
DB2®	Power Architecture®	System Storage DS®
developerWorks®	POWER Hypervisor™	System Storage®
DS8000®	Power Systems™	System x®
Dynamic Infrastructure®	POWER6+™	System z10®
Easy Tier®	POWER6®	System z®
ECKD™	POWER7®	TDMF®
Enterprise Storage Server®	PowerVM®	Tivoli®
FICON®	POWER®	WebSphere®
FlashCopy®	pSeries®	XIV®
GDPS®	Redbooks®	z/OS®
HyperSwap®	Redpaper™	z/VM®
i5/OS®	Redbooks (logo)  ®	z/VSE®
IBM®	Resource Link®	z10™
IMS™	Resource Measurement Facility™	z9®
Informix®	RMF™	zEnterprise®
Lotus Notes®	Smarter Planet™	zSeries®

The following terms are trademarks of other companies:

Intel, Intel logo, Intel Inside logo, and Intel Centrino logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

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

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

Snapshot, and the NetApp logo are trademarks or registered trademarks of NetApp, Inc. in the U.S. and other countries.

Java, and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

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

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

# Preface

This IBM® Redbooks® publication addresses host attachment and interoperability considerations for the IBM System Storage® DS8000® series. Within this book, you can find information about the most popular host operating systems platforms, including Windows, IBM AIX®, VIOS, Linux, Solaris, HP-UX, VMware, Apple, and IBM z/OS®.

The topics covered in this book target administrators or other technical personnel with a working knowledge of storage systems and a general understanding of open systems. You can use this book as guidance when installing, attaching, and configuring System Storage DS8000.

The practical, usage-oriented guidance provided in this book complements the *IBM System Storage DS8000 Host Systems Attachment Guide, SC26-7917*.

## The team who wrote this book

This book was produced by a team of specialists from around the world, working for the International Technical Support Organization (ITSO), at the European Storage Competence Center (ESCC) in Mainz.

**Axel Westphal** is an IT Specialist for IBM Storage Systems at the IBM European Storage Competence Center (ESCC) in Mainz, Germany. He joined IBM in 1996, working for Global Services as a System Engineer. His areas of expertise include setup and demonstration of IBM System Storage products and solutions in various environments. Since 2004, Alex has been responsible for storage solutions and Proof of Concepts conducted at the ESSC with DS8000, SAN Volume Controller, and IBM XIV®. He has been a contributing author to several XIV and DS8000-related IBM Redbooks publications.

**Bertrand Dufrasne** is an IBM Certified Consulting IT Specialist and Project Leader for IBM System Storage disk products at the International Technical Support Organization, San Jose Center. He has worked at IBM in various IT areas. Bertrand has written many IBM Redbooks publications, and has also developed and taught technical workshops. Before joining the ITSO, he worked for IBM Global Services as an Application Architect in the retail, banking, telecommunication, and health care industries. He holds a Masters degree in Electrical Engineering.

**Juan Brandenburg** is a Product Field Engineer for the DS8000 in the USA. He is a graduate from the University of Arizona, holding a Bachelors of Engineer Management in Computer Engineering. His areas of experience for hardware include the DS8000 series and IBM System x® series server. Juan has been working for IBM for seven years in the Global Mirror, ESSNI, and DS8000 departments. He has many years of experience scripting for Linux, IBM AIX®, and Windows environments. Juan has continuously participated in IBM technical competitions, for which he has earned him some awards such as the Distinguished Engineer Award for the 2006 Tech Connect competition, placing him in the IBM Epic Express top 14 coops for July 2006.

**Jana Jamsek** is an IT Specialist for IBM Slovenia. She works in Storage Advanced Technical Support for Europe as a specialist for IBM Storage Systems and the IBM i (formerly known as the IBM i5/OS®) operating system. Jana has eight years of experience in working with the IBM System i® platform and its predecessor models, and eight years of experience in working with storage. She has a Master's degree in Computer Science and a degree in Mathematics from the University of Ljubljana in Slovenia.

**Kai Jehnen** is a Product Field Engineer for the DS8000 and has been working for the past seven years in this area. He holds a degree in Information Technologies from the University of Applied Sciences in Koblenz. His main focus is on solving critical customer situations in open systems environments. He is also a VMware Certified Professional.

**Steven Joseph** is a Staff Software Engineer and Team Leader for the Product Engineering Tools Team in Tucson, Arizona. He has been with IBM for eight years, starting as a Product Field Engineer for ESS and DS8000. Now as a developer he works on RAS features for the DS8000 family and data analysis tools for global Product Engineering teams for all Storage products. He is a Certified Java Developer and has been developing AIX software for over 15 years. He also holds multiple vendor certifications from HP, Cisco, and Sun. Steven currently leads the development team for the ARK project, producing tools for remote support and real-time diagnostics for IBM Disk and Tape products.

**Massimo Olivieri** joined IBM in 1996 to work as Country Specialist for Tape Products. He moved on to the Disks High End family as FE support. He is a DS8000 and XIV Top Gun Specialist and has good knowledge of Storage on IBM z/OS® environments, with 25 years experience. His areas of experience also include critical situation management for storage environments.

**Ulrich Rendels** is an IT Specialist working for IBM Germany. He has been with IBM for 15 years, starting as a Product Field Engineer for 7135 and ESS. For the past 10 years, he has been a member of world wide storage development test-teams supporting ESSNI DS8000-Development and currently qualifying IBM SVC/Storwize® V7000 and Storwize Unified systems.

**Mario Rodriguez** is an IBM Certified Systems Expert working for IBM Uruguay since 2001. He holds a degree in Computer Science and also multiple vendor certifications from IBM, Microsoft, VMware, LPI, and Comptia. His areas of expertise include SAN Switches, Storage Systems, AIX, Linux, and VMware. His main role in IBM Uruguay is to provide technical support services for virtualization and storage products.

Many thanks to the people at the IBM Systems Lab Europe in Mainz, Germany, who helped with the equipment provisioning and preparation:

Uwe Heinrich Müller  
Günter Schmitt  
Mike Schneider  
Dietmar Schniering  
Uwe Schweikhard  
IBM Systems Lab Europe, Mainz, Germany

Special thanks to the Enterprise Disk team manager, Bernd Müller, and the ESCC director, Klaus-Jürgen Rüniger, for their continuous interest and support regarding the ITSO Redbooks projects.

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

Björn Wesselbaum  
Stefen Deierling  
Hans-Paul Drum  
Wilhelm Gardt  
Torsten Rothenwaldt  
Edgar Strubel  
IBM

## Now you can become a published author, too!

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

For more information about the residency program, browse the residency index, or apply online, see the following website:

[ibm.com/redbooks/residencies.html](http://ibm.com/redbooks/residencies.html)

## Comments welcome

Your comments are important to us!

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

- ▶ Use the online **Contact us** review Redbooks form found at this website:

[ibm.com/redbooks](http://ibm.com/redbooks)

- ▶ Send your comments in an email to:

[redbooks@us.ibm.com](mailto:redbooks@us.ibm.com)

- ▶ Mail your comments to:

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

## Stay connected to IBM Redbooks

- ▶ Find us on Facebook:

<http://www.facebook.com/IBMRedbooks>

- ▶ Follow us on Twitter:

<http://twitter.com/ibmredbooks>

- ▶ Look for us on LinkedIn:

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

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

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

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

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







# General considerations

This chapter provides general information for the host attachment of DS8000 storage systems as well as information that pertains to all chapters of this book.

The following topics are covered:

- ▶ DS8000 general topics
- ▶ Client architecture overview
- ▶ SAN considerations
- ▶ Storage tiers

## 1.1 DS8000 general topics

This introductory chapter addresses some of the considerations that apply to all of the different types of hosts that can be attached to a DS8000 storage system. It also contains information covering many of the conventions and names used throughout the rest of this book.

The DS8000 family of storage systems has grown through the years, steadily adding new features and industry-leading performance to a well-established code base.



Figure 1-1 The DS8000 family of storage systems

The focus of this book is on attaching hosts to the latest DS8000 storage system, the DS8800. There are also several areas where the previous generation, the DS8700, is mentioned, because these systems share some common architecture elements.

Throughout this book, if a feature or component is specific to a model, the exact model name is used, such as DS8700 or DS8800. If the information is of a more general nature, then the less-specific term DS8000 is used, indicating that the information is relevant to all the models of the DS8000 family of products.

**Important:** When information in this book applies to only a specific storage system model, such as the *DS8800*, it is referenced explicitly. References to *DS8000* are more general and apply to the entire family of DS8000 storage systems, not a specific model.

## 1.2 Client architecture overview

To assist with upgrade planning or troubleshooting, the client needs to provide current versions of the following items:

- ▶ SAN: A layout or map identifying all the components in the SAN along with contact information for storage administrators
- ▶ SAN: A table that cross-references LUNs from a host to the DS8000
- ▶ PPRC: A layout of which storage subsystems are involved in Copy Services relationships and the type of connectivity between them
- ▶ Encryption: A layout of TKLM servers and their vital information as well as contact information for security administrators

For future planning, IBM can provide assistance in developing these types of documentation. Contact your IBM support representative for more information about services offered.

## 1.3 SAN considerations

The following list contains some general considerations regarding maintenance and configuration for Storage Area Networks (SANs). These “rules” are valid for both open and z/OS environments:

- ▶ Periodically check the SAN from the SAN Administrator for errors. Always do this operation before any hardware change.
- ▶ Fix any errors before making any configuration changes to avoid multiple problems in the SAN environment.
- ▶ Before implementing any new configuration such as installing new hardware, verify the firmware levels on all the equipment. If needed, update the firmware on all SAN switches to be sure that any old compatibility issue was resolved.
- ▶ Before proceeding with any new hardware installation, evaluate the SAN bandwidth to be sure that is not already to close to its limits. This precaution can help avoid problems after the new hardware becomes operational.

If a client needs assistance in performing these checks, IBM provides many levels of SAN evaluations and upgrade planning.

## 1.4 Storage tiers

This publication has several references to drive tiers, storage tiers, and the IBM Easy Tier® application. This section discusses what is meant by the term *tier* relative to the DS8800.

When a workload for storage is laid out, or *provisioned*, there are some types of data that are known ahead of time to be more resource-hungry. That is, the data from some workloads will be accessed more frequently or might change more frequently than other workloads. Some applications require data to be stored and then only occasionally read; other applications require thousands of reads and writes daily.

Because of the differing needs of storage data, the DS8800 supports installation of three physically different classes of storage, all of which use the term Disk Drive Module (DDM). The real world capabilities of these storage types are different enough that they are considered to be in different classes of performance, also known as tiers. Here are the types:

- ▶ Solid State (SSD): NAND-based flash memory is used for storage, no moving parts, 2.5" form factor, 300 GB to 400 GB, considered as the highest tier
- ▶ Enterprise: Traditional electromechanical drive, high (10,000 and 15,000) RPM, 2.5" form factor, 146 GB to 900 GB, considered as a middle tier
- ▶ Nearline: Traditional electromechanical drive, 7,200 RPM, 3.5" form factor, 3 TB, considered as the lowest tier

The busiest workloads, or the most mission-critical data, is said to be *hot* relative to other application data that might be held on the same storage subsystem. Typically, put the hottest data on the fastest storage medium and colder data onto slower storage. The DS8800 system can support the installation of all three storage types within its frames, and then the ranks and extents can be provisioned accordingly. And of course, an application's needs can change over the lifetime of its data, so a layout that makes sense today might not hold true tomorrow.

The Easy Tier software can constantly evaluate the "temperature" of the data as it is accessed and make changes to the storage, between tiers and within a tier, accordingly. The hottest data can be provisioned onto the highest tier and other data can be moved as appropriate. In addition, the workloads on a tier can be balanced out, making certain that storage units are not overused while other capacity exists. Manual arrangement of data storage between the tiers is possible as well.

For more detailed information about Easy Tier and the performance characteristics of the DS8000 storage types, see the Redpaper™ publication, *IBM System Storage DS8000 Easy Tier*, REDP-4667-02.



## Open systems considerations

This chapter provides information about, and general considerations for, attaching IBM System Storage DS8000 series systems to open systems hosts. It includes references to available documentation and additional information.

The following topics are covered:

- ▶ Configuration resources
- ▶ Using the DS8000 as a boot device
- ▶ Additional supported configurations
- ▶ Multipathing support for Subsystem Device Drivers

## 2.1 Configuration resources

This section describes the online resources where you can find detailed and up-to-date information about supported configurations, settings, device driver versions, and so on, for open systems. Because of the high innovation rate in the IT industry, the support information is updated frequently. Visit these resources regularly to check for updates.

### 2.1.1 IBM resources

Resources covered in this section include the IBM System Storage Interoperation Center (SSIC), host systems attachment, installation, and troubleshooting, as provided by IBM.

#### IBM System Storage Interoperation Center

For information about supported Fibre Channel (FC) host bus adapters (HBAs) and the required firmware and device driver levels for all IBM storage systems, visit the IBM SSIC at the following website:

<http://www.ibm.com/systems/support/storage/ssic/interoperability.wss>

For each list box, select a storage system, a server model, an operating system, and an HBA type. Each list box provides a list of all supported HBAs together with the required firmware and device driver levels of your selection. Furthermore, a list of supported storage area network (SAN) switches and directors is displayed.

Alternatively, you can generate and download a Microsoft Excel data file that contains information based on the product family, model family, and product version that you have selected. The information in this file is similar to the information that was in the previous interoperability matrix for System Storage. Select the family, model, and version of your product and click the **Export Selected Product Version (xls)** link, as shown in Figure 2-1, to generate an Excel data file, based on your selections.

IBM Systems > Support >

## System Storage Interoperation Center (SSIC)

[SSIC Education and Help](#)

Please view the details of the interoperability configurations queried. This requires exporting the data, or clicking the Submit button at the bottom of the search interface, then clicking on the details link in the results table.

**Revise Selected Criteria - click link below to change search query**

(1) [Product Family](#), (2) [Product Model](#), (3) [Product Version](#)

**New Search**                      **Configuration Results=**  
14590197

Product Family	Product Model
IBM System Storage Enterprise Disk	DS8700
IBM System Storage Enterprise Tape	DS8100/DS8300
IBM System Storage Entry Disk	DS6000 Series
IBM System Storage LTO Ultrium Tape	ESS 750

**Product Version**

- DS8100/DS8300 R3.1 (bundle 63.10.xx)
- DS8100/DS8300 R4 (bundle 64.00.xx)
- DS8100/DS8300 R4.21 (bundle 64.21.xx)
- DS8100/DS8300 R4.3 (bundle 64.30.xx)**

[Export Selected Product Version \(xls\)](#)

Figure 2-1 Generating an Excel data file based on your selections

**Tip:** In order to obtain the most accurate results, it is highly advisable to select as much detail as possible in the interoperability website.

## DS8000 Host Systems Attachment Guide

For information about how to attach an open systems host to your DS8000 storage system, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917, at the following website:

<http://www.ibm.com/support/docview.wss?uid=ssg1S7001161>

**Tip:** The DS8000 Host Attachment and Interoperability book includes information for the most widely used host. Make sure to use the DS8000 host Systems Attachment Guide book to review the full list of supported host by the DS8000 systems.

## Installation and troubleshooting documentation

For general installation and troubleshooting documentation, see the IBM Support Portal site:

<http://www.ibm.com/support/us/en/>

### 2.1.2 HBA vendor resources

All of the Fibre Channel HBA vendors have websites that provide information about their products, facts and features, and support information. These sites can be helpful when you need details that cannot be supplied by IBM resources, for example, when troubleshooting an HBA driver. IBM cannot be held responsible for the content of these sites.

#### QLogic Corporation

See the QLogic website at the following website:

<http://www.qlogic.com>

QLogic maintains a page that lists all the HBAs, drivers, and firmware versions that are supported for attachment to IBM storage systems, which you can find at the following website:

[http://support.qlogic.com/support/oem\\_ibm.asp](http://support.qlogic.com/support/oem_ibm.asp)

#### Emulex Corporation

You can find the Emulex home page at this website:

<http://www.emulex.com>

Emulex also has a page with content specific to IBM storage systems at the following website:

<http://www.emulex.com/products/host-bus-adapters/ibm-branded.html>

#### Oracle

Oracle ships its own HBAs, which are Emulex and QLogic based. However, the *native* HBAs from Emulex and QLogic can be used to attach servers running Oracle Solaris to disk systems. In fact, these native HBAs can even be used to run StorEdge Traffic Manager software. For more information about the Oracle HBAs, see the following websites:

- For Emulex:

<http://www.oracle.com/technetwork/server-storage/solaris/overview/emulex-corporation-136533.html>

- ▶ For QLogic:

<http://www.oracle.com/technetwork/server-storage/solaris/overview/qlogic-corp--139073.html>

### **Hewlett-Packard**

HP ships its own HBAs, which are Emulex and QLogic based:

- ▶ Emulex publishes a cross reference at the following website:

<http://www.emulex-hp.com/interop/matrix/index.jsp?mfgId=26>

- ▶ QLogic publishes a cross reference at the following website:

[http://driverdownloads.qlogic.com/QLogicDriverDownloads\\_UI/Product\\_detail.aspx?oemid=21](http://driverdownloads.qlogic.com/QLogicDriverDownloads_UI/Product_detail.aspx?oemid=21)

### **Atto Technology, Inc.**

Atto Technology supplies HBAs for Apple Macintosh, which is supported by IBM for the attachment of the DS8000.

The Atto Technology home page is found at the following website:

<http://www.attotech.com>

You can find the Atto Technology support page at the following website:

<http://www.attotech.com/solutions/ibm.html>

You must register with Atto Technology to download drivers and utilities for HBAs.

### **Platform and operating system vendor pages**

The platform and operating system vendors also provide support information for their clients. See their information for general guidance about connecting their systems to SAN-attached storage. If you cannot find information that can help you with third-party vendors, check with your IBM representative about interoperability, and support from IBM for these products. It is beyond the scope of this book to list all of the vendor websites.

## **2.2 Using the DS8000 as a boot device**

For most of the supported platforms and operating systems, you can use the DS8000 as a boot device. The IBM SSIC provides detailed information about boot support for specific operating systems. For open systems, the first step is to configure the HBA to load a BIOS extension. It provides the basic input and output capabilities for a SAN-attached disk, assuming that the host is equipped with QLogic HBAs.

**Tip:** For IBM branded QLogic HBAs, the BIOS setup utility is called *FAST!Util*.



## 2.2.1 Configuring the QLogic BIOS to boot from a DS8000 volume

To enable the QLogic BIOS to boot from a DS8000 volume, follow these steps:

1. When the QLogic HBAs are initialized during the server startup process, press Ctrl+Q (Figure 2-2). When the message “<CTRL-Q> Detected, Initialization in progress, Please wait...” is displayed, the QLogic BIOS setup utility is being loaded.

```
QLogic Corporation
QLE2460 PCI Fibre Channel ROM BIOS Version 2.02
Copyright (C) QLogic Corporation 1993-2008. All rights reserved.
www.qlogic.com

Press <CTRL-Q> or <ALT-Q> for Fast!UTIL

<CTRL-Q> Detected, Initialization in progress, Please wait...

BIOS for Adapter 0 is disabled

BIOS for Adapter 1 is disabled
ROM BIOS NOT INSTALLED
```

Figure 2-2 Entering the QLogic BIOS setup utility

Use these methods to navigate the menu items:

- Press the arrow keys to move between menu items.
  - Press Enter to select or change an item.
  - Press Esc to leave a window and return to the previous level.
2. When the QLogic BIOS setup utility opens, select a host adapter, if more than one is installed. If there is only one installed, it will be selected by default. After the HBA that you want to configure is selected, press Enter.
  3. In the main Fast!UTIL Options panel (upper panel of Figure 2-3), leave the cursor on **Configuration Settings** (default), and press Enter.
  4. In the Configuration Settings panel (middle panel of Figure 2-3), leave the cursor on **Adapter Settings** (default) and press Enter.

5. In the Adapter Settings panel (bottom panel of Figure 2-3) for Host Adapter BIOS, select **Enabled** and then press Enter.

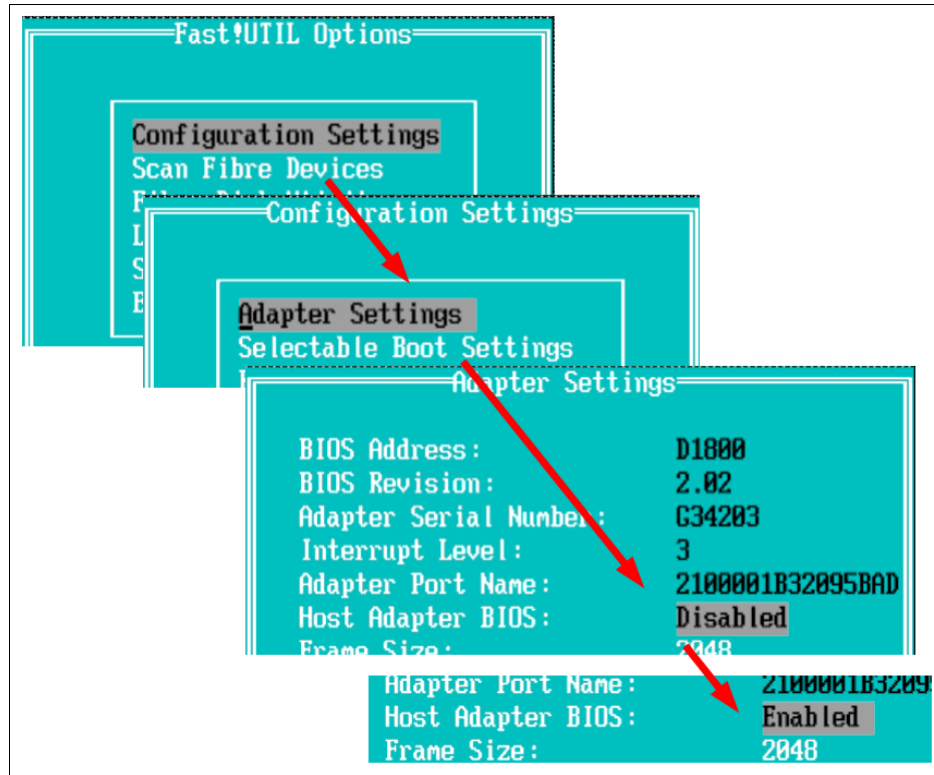


Figure 2-3 Enabling the QLogic HBA BIOS extension

6. In the Adapter Settings panel (bottom panel of Figure 2-3) for Host Adapter BIOS, select **Enabled** and then press Enter.
7. In the Adapter Settings panel, press Esc to return to the Configuration Settings panel.
8. In the Configuration Settings panel, select **Selectable BIOS Settings** and press Enter.
9. In the Selectable BIOS Settings panel, press Enter.

10. In the Selectable Boot Settings panel, modify the boot settings:
  - a. Change Selectable Boot to **Enabled** and press Enter, as shown in the middle and lower panels of Figure 2-4.

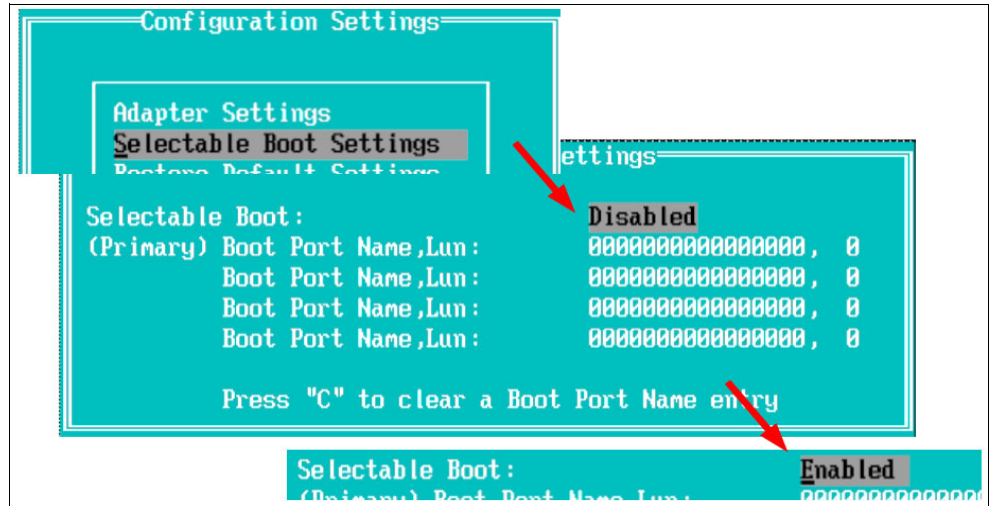


Figure 2-4 Enabling the selectable boot

- b. Select the first boot device entry and press Enter, as shown in the upper panel of Figure 2-5.

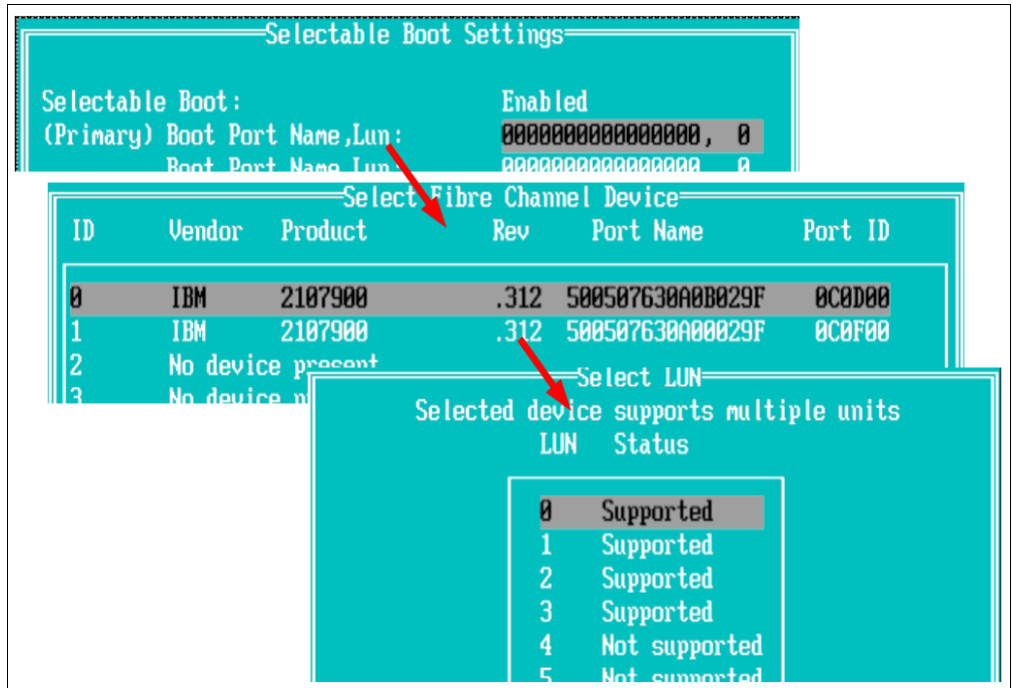


Figure 2-5 Select boot device

- c. In the Select Fibre Channel Device panel, as shown in the middle panel of Figure 2-5 on page 11, select the DS8000 port that you want to use to access the boot volume. Press Enter.

- d. In the Select LUN panel, which shows the available ports with their worldwide port name (WWPN), as shown in the lower panel of Figure 2-5 on page 11, select the logical unit number (LUN) that represents the boot volume. Then press Enter.
  - e. Check the boot entry that you just created.
11. Optional: Configure a second boot entry, for example, to have redundant links to the boot volume. To configure a second HBA, complete the following steps:
- a. On the main option panel of the utility, move the cursor down to the **Select Host Adapter** entry and press Enter.
  - b. Select the same DS8000 volume, but a separate host port for the redundant connection.
  - c. Repeat steps 1 on page 9 through 10 on page 11.
  - d. When BIOS configurations are complete, leave the utility and reboot the server.
12. To exit the configuration for the HBA, press ESC twice.
13. When you see the Configuration settings modified message in the lower panel, shown in Figure 2-6, select **Save changes** (default) and press Enter.

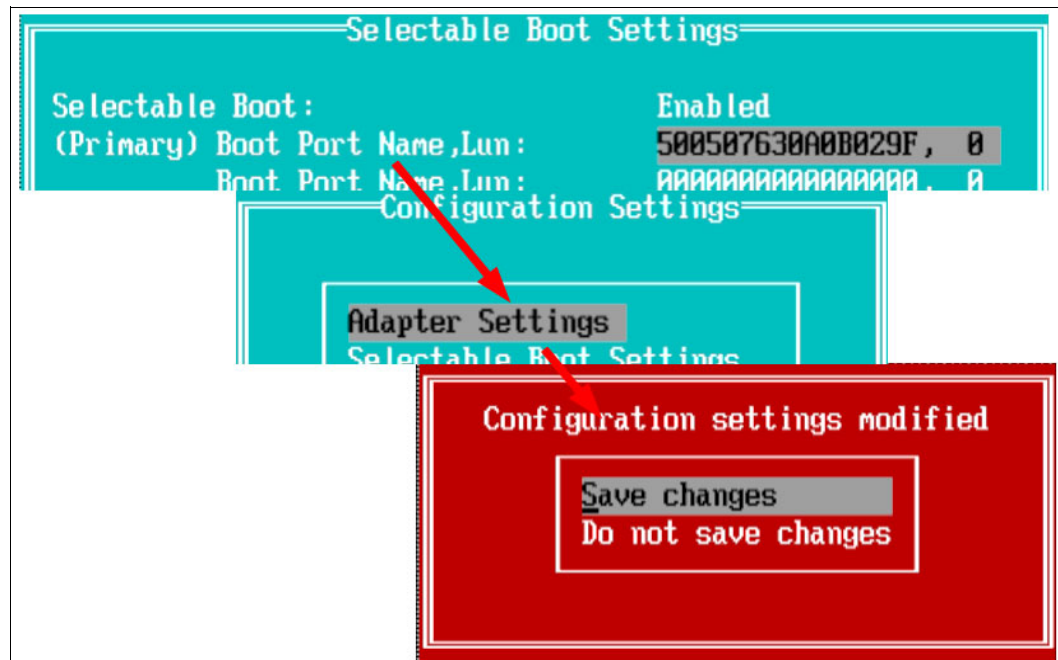


Figure 2-6 Checking and saving changes

Configuring the BIOS for both HBAs gives you the advantage of a redundant connection to the boot device, and enables the system to start even if part of the connectivity, for example a switch, is unavailable.

**Important:** Redundant BIOS connections to the boot volume enable the system to start even if a path to the storage system is missing. A failover capability is not provided when a path fails during the boot process. Between the start of the boot process and the activation of the operating system's multipathing, the system is not protected against a path failure.

If you see messages such as the messages in Figure 2-7, you have configured the BIOS correctly. The QLogic BIOS is enabled and the system attempts to start from the defined DS8000 volume.

```
Drive Letter C: is Moved to Drive Letter D:
LOOP ID 1,,0 is Installed As Drive C:

Device Device Adapter Port Lun Vendor Product Product
Number Type Number ID Number ID ID Revision
00 Disk 0 0C0D00 0 IBM 2107900 .312
ROM BIOS Installed
```

Figure 2-7 QLogic BIOS enabled messages

**Booting:** Emulex HBAs also support booting from SAN disk devices. Press Alt+E or Ctrl+E to enable and configure the Emulex BIOS extension when the HBAs are initialized during server startup.

For more detailed instructions, see the following Emulex publications:

- ▶ Supercharge Booting Servers Directly from a Storage Area Network:  
<http://www.emulex.com/artifacts/fc0b92e5-4e75-4f03-9f0b-763811f47823/bootingServersDirectly.pdf>
- ▶ Enabling Emulex Boot from SAN on IBM BladeCenter®:  
[http://www.emulex.com/artifacts/4f6391dc-32bd-43ae-bcf0-1f51cc863145/enabling\\_boot\\_ibm.pdf](http://www.emulex.com/artifacts/4f6391dc-32bd-43ae-bcf0-1f51cc863145/enabling_boot_ibm.pdf)

## 2.2.2 Next steps

For selected operating systems, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917. This guide provides procedures that are needed to boot a host off the DS8000.

For information about identifying an optimal configuration and how to boot from multipathing devices, see the *IBM System Storage Multipath Subsystem Device Driver User's Guide*, GC52-1309. You can download this guide from the following FTP site:

<ftp://ftp.software.ibm.com/storage/subsystem/UG/1.8--3.0/>

## 2.3 Additional supported configurations

For instances where the configuration that you want is not covered by the SSIC, you can use the *Storage Customer Opportunity Request (SCORE)* process, formerly known as Request for Price Quotations (RPQ). Contact your IBM storage sales specialist or IBM Business Partner for submission of a SCORE.

Initiating the process does not guarantee that the desired configuration will be supported. Any approval depends on the technical feasibility and the required test effort. A configuration that equals, or is similar to, one of the already approved configurations is more likely to become approved, than a completely separate configuration.

## 2.4 Multipathing support for Subsystem Device Drivers

To ensure maximum availability, most clients choose to connect their open systems hosts through more than one Fibre Channel path to their storage systems. An intelligent SAN layout helps protect from failures of Fibre Channel HBAs, SAN components, and host ports in the storage system.

Certain operating systems, however, cannot work natively with multiple paths to a single disk, putting the data's integrity at risk. The risk occurs because multiple write requests can be issued to the same data, and nothing takes care of the correct order of writes.

To use the redundancy and increased input/output (I/O) bandwidth when using multiple paths, an additional layer is needed in the operating system's disk device representation concept. This additional layer recombines the multiple disks seen by the HBAs into one logical disk. This layer also manages path failover when a path becomes unusable and balances I/O requests across the available paths.

To evaluate the customer configuration to ensure proper performance and installation of the devices, see the following SAN Zoning documentation:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP101914>

### 2.4.1 Subsystem Device Driver

For most operating systems that are supported for DS8000 attachment, IBM provides the IBM Subsystem Device Driver (SDD), at no cost, to help support the following functionality:

- ▶ Enhanced data availability through automatic path failover and failback
- ▶ Increased performance through dynamic I/O load balancing across multiple paths
- ▶ The ability for concurrent download of licensed internal code
- ▶ User configurable path selection policies for the host system

IBM Multipath SDD provides load balancing and enhanced data availability in configurations with more than one I/O path between the host server and the DS8000. SDD performs dynamic load balancing across all available preferred paths to ensure full utilization of the SAN and HBA resources.

You can download SDD from the following website:

<http://www.ibm.com/support/dlsearch.wss?rs=540&tc=ST52G7&dc=D430>

For help with SDD, see the *IBM System Storage Multipath Subsystem Device Driver User's Guide*, GC52-1309. It contains all the information that is needed to install, configure, and use SDD for all supported operating systems. You can download it from this website:

<ftp://ftp.software.ibm.com/storage/subsystem/UG/1.8--3.0/>

**Important:** IBM does not provide an SDD for each operating system described in this book. For instance, an SDD for HP-UX 11iv3 is not available.

## 2.4.2 Other multipathing solutions

Certain operating systems come with native multipathing software such as the following examples:

- ▶ StorEdge Traffic Manager for Oracle Solaris:

This software is for Solaris 10 only. If you use an older Solaris version, you must explicitly download this software.

- ▶ HP-UX in version 11iv3:

While HP still provides PVLinks for HP-UX 11iv3, it is generally not advisable to use PVLinks, but rather the native multipathing that is provided with the operating system. Native multipathing can help balance the I/O load across multiple paths.

- ▶ IBM AIX native multipathing I/O support (MPIO)
- ▶ IBM i multipathing support

In addition, third-party multipathing solutions are available such as Veritas DMP, which is part of Veritas Volume Manager.

Most of these solutions are also supported for DS8000 attachment, although the scope can vary. Certain HBAs or operating system versions might have limitations.

For the latest information, see the IBM SSIC at this website:

<http://www.ibm.com/systems/support/storage/config/ssic>







## Windows considerations

This chapter provides details for attaching IBM System Storage DS8000 series systems to host systems for Windows Server 2003 and Windows Server 2008.

The following topics are covered:

- ▶ Attaching HBAs in Windows
- ▶ Installing SDD in Windows
- ▶ Clustering for Windows 2003 Server
- ▶ Using Multipath Input/Output for Windows 2003 and 2008
- ▶ Installing and configuring SDDDSM in Windows 2003, 2008
- ▶ Expanding dynamic disk for Windows 2003, Windows 2008
- ▶ SAN boot support
- ▶ Windows Server 2003 Virtual Disk Service support
- ▶ Hyper V considerations

For additional information about the advantages, disadvantages, potential difficulties, and troubleshooting with SAN booting, see the Microsoft document, *Boot from SAN in Windows Server 2003*, at the following addresses:

<http://technet.microsoft.com/en-us/windowsserver/bb512919>

<http://technet.microsoft.com/en-us/library/cc786214%28v=WS.10%29.aspx>

## 3.1 Attaching HBAs in Windows

DS8000 supports Fibre Channel attachment to Microsoft Windows Server 2003 and Windows Server 2008 servers. For details regarding operating system versions and Host Bus Adapter (HBA) types, see the IBM SSIC website

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

Attachment support includes cluster service. The DS8000 can also act as a boot device. Booting is supported currently with host adapters QLA23xx (32-bit or 64-bit) and LP9xxx (32-bit only).

### 3.1.1 HBA and operating system settings

Depending on the HBA type, several HBA and driver settings might be required. For a complete description of these settings, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917. Although you can access the volumes by using other settings, the values provided in this guide were tested.

Set the Time Out Value parameter associated with the host adapters to 60 seconds, when attaching a storage unit to one of the following Microsoft systems:

- ▶ Windows 2003 Server
- ▶ Windows 2008 Server

This helps ensure optimum availability and recoverability.

The operating system uses the Time Out Value parameter to bind its recovery actions and responses to the disk subsystem. The value is stored in the Windows registry in the HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Disk\TimeOutValue path. The value has the data type REG-DWORD and must be set to 0x0000003c hexadecimal (60 decimal).

### 3.1.2 SDD versus SDDDSM Multipath Drivers

There are some big differences to point out between SDD and SDDDSM Multipath Drivers. If you are installing Windows Server 2003, it is highly advisable to move to SDDDSM because SDD is an older application version compatible with Windows 2000 Servers, which is no longer supported by IBM. If you install SDD in Windows Server 2003 you will have many multipathing limitations, which are enhanced with SDDDSM because it has module integration and works closer to the OS. Windows Server 2008 builds upon that and works natively by integrating MPIO and further supporting IBM disk drive configurations. Make sure to read the differences in the following section links for SDD and SDDDSM.

## 3.2 Installing SDD in Windows

An important task with a Windows host is the installation of the SDD multipath driver. Ensure that SDD is installed before adding additional paths to a device; otherwise, the operating system can lose the ability to access existing data on that device. For details, see the *IBM System Storage Multipath Subsystem Device Driver User's Guide*, SC30-4131. To make sure the SDD system is installed, you can verify if its installation files exist under C:\\Program Files\\ibm\\SDD.

SDD and all the required guide documentation can be downloaded from the following link:

<http://www.ibm.com/support/docview.wss?uid=ssg1S4000054>

**Support:** Windows 2000 is no longer supported. This chapter includes SDD information about Windows Server 2003 and 2008.

**Tip:** During the SDD installation, the option “Cache on Disk” might be enabled, in which case the system will use the server’s cache. The DS8k already has cache enabled, therefore, it is not necessary to use the servers cache. You can disable this option by going to **disk manager** → **right-click the partition** → **Properties** → **Policies** → **uncheck “Enable write caching on the Disk”**. See Figure 3-1.

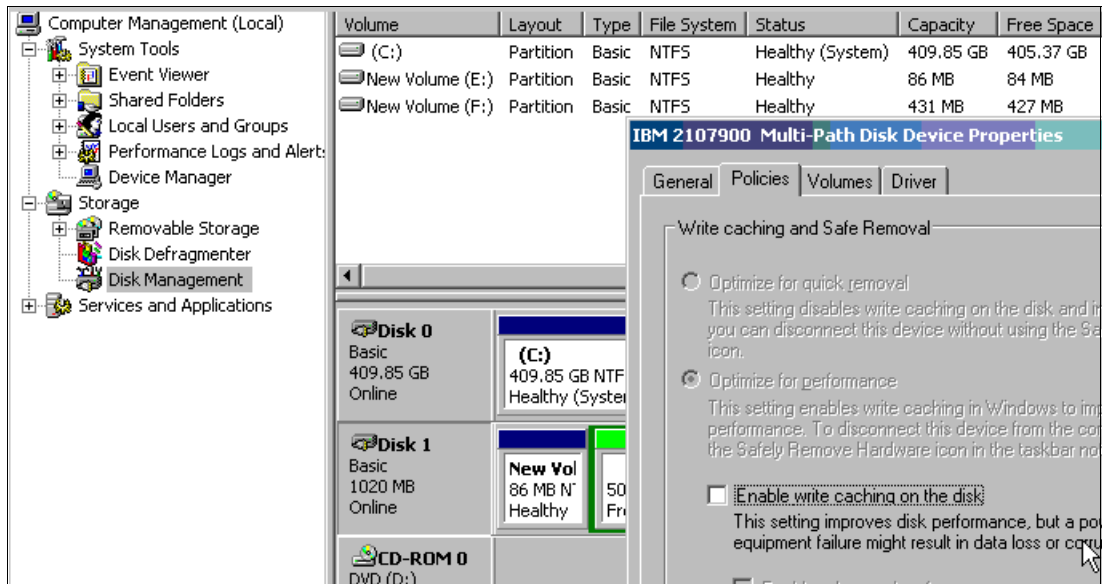


Figure 3-1 Shows how to disable the server cash write

Upon connecting to the DS8k, you will see this appear under the Device Manager view. These two disks are connected by four paths to the server. The IBM 2107900 Small Computer System Interface (SCSI) disk device is hidden by the SDD. See Figure 3-2.

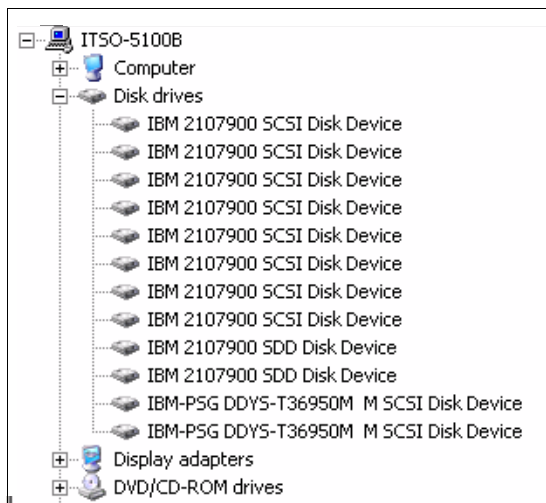


Figure 3-2 Device Manager view with SDD paths

Figure 3-3 shows two SDD devices with the Disk Management view. They can be allocated by initializing the disk drives and formatting the proper partition size.

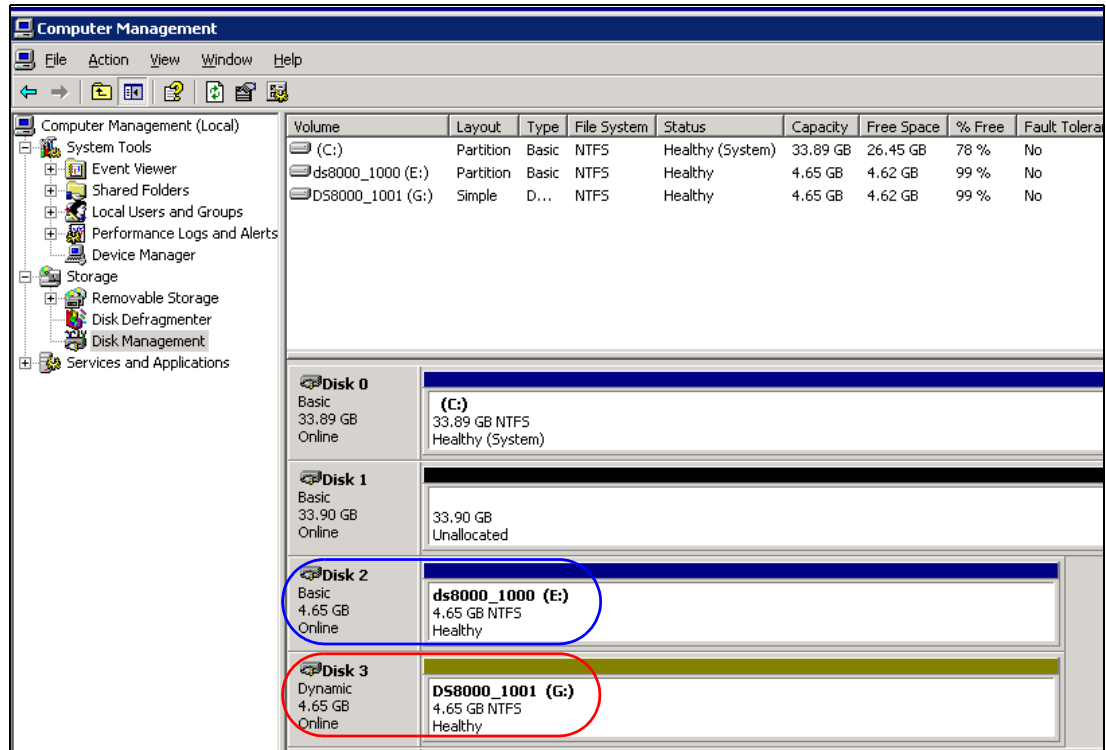


Figure 3-3 Disk Management view with two devices

**Tip:** Newly assigned disks will be discovered; if not, go to the Disk Manager window and rescan the disks or go to the Device Manager window and scan for hardware changes.

### 3.2.1 SDD datapath query

When using the **datapath device** query command, option **-1** is added to mark the non-preferred paths with an asterisk. For example, with **datapath query device -1**, all paths of the DS8000 are used, because the DS8000 does not implement the concept of preferred path, as the SAN volume controller (SVC) does. Example 3-1 shows the output of the **datapath query device -1** command.

Example 3-1 Output of datapath query device -1 command

```
Microsoft Windows [Version 5.2.3790]
(C) Copyright 1985-2003 Microsoft Corp.

C:\Program Files\IBM\Subsystem Device Driver>datapath query device -1

Total Devices : 2

DEV#: 0 DEVICE NAME: Disk2 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75065711000
LUN IDENTIFIER: 6005076303FFC0B60000000000001000
=====
```

Path#	Adapter/Hard Disk	State	Mode	Select	Errors
0	Scsi Port2 Bus0/Disk2 Part0	OPEN	NORMAL	22	0
1	Scsi Port2 Bus0/Disk2 Part0	OPEN	NORMAL	32	0
2	Scsi Port3 Bus0/Disk2 Part0	OPEN	NORMAL	40	0
3	Scsi Port3 Bus0/Disk2 Part0	OPEN	NORMAL	32	0

DEV#: 1 DEVICE NAME: Disk3 Part0 TYPE: 2107900 POLICY: OPTIMIZED  
 SERIAL: 75065711001  
 LUN IDENTIFIER: 6005076303FFC0B60000000000001001

=====

Path#	Adapter/Hard Disk	State	Mode	Select	Errors
0	Scsi Port2 Bus0/Disk3 Part0	OPEN	NORMAL	6	0
1	Scsi Port2 Bus0/Disk3 Part0	OPEN	NORMAL	4	0
2	Scsi Port3 Bus0/Disk3 Part0	OPEN	NORMAL	4	0
3	Scsi Port3 Bus0/Disk3 Part0	OPEN	NORMAL	2	0

To obtain the WWPN of your Fibre Channel adapter, use the **datapath query wwpn** command as shown in Example 3-2.

*Example 3-2 The datapath query command*

```
C:\Program Files\IBM\Subsystem Device Driver>datapath query wwpn
Adapter Name      PortWWN
Scsi Port2:      210000E08B037575
Scsi Port3:      210000E08B033D76
```

The **datapath query essmap** and **datapath query portmap** commands are not available.

### 3.2.2 Mapping SDD devices to Windows drive letters

When assigning DS8000 LUNs to a Windows host, you must understand which Windows drive letter corresponds to which DS8000 LUN. To determine which Windows drive corresponds to which DS8000 LUN, evaluate the output from running the **datapath query device** command and from the Windows Disk Management window.

In Example 3-1 on page 20, if you listed the vpaths, you can see that SDD DEV#: 0 has DEVICE NAME: Disk2. You can also see the serial number of the disk is 75065711000, which is displayed as LUN Identifier 1000 on DS8000 serial 7506571.

Then look at the Windows Disk Management window, an example of which is shown in Figure 3-3 on page 20. In this example, Disk 2 is Windows drive letter E: circled in blue. The Windows drive letter G: circled in red corresponds to the SDD DEV#.

Now that you have mapped the LUN ID to a Windows drive letter, if drive letter E was no longer required on this Windows server, you can safely unassign LUN ID 1000 on the DS8000 with serial number 7506571, with the understanding that you have removed the correct drive.

## 3.3 Clustering for Windows 2003 Server

SDD 1.6.0.0 or later is required to support load balancing in Windows clustering.

When running Windows clustering, clustering failover might not occur when the last path is being removed from the shared resources.

### 3.3.1 References

For additional information, see the Microsoft article “Removing the HBA cable on a server cluster”, at the following website:

<http://support.microsoft.com/default.aspx?scid=kb;en-us;Q294173>

You can check your SDD version and MSCS settings to reduce reservation conflicts. More information can be found in the following website:

<http://technet.microsoft.com/en-us/library/cc739757%28v=ws.10%29.aspx>

### 3.3.2 SDD support

SDD handles path reclamation in a Windows cluster environment subtly different from a non-clustering environment. When the Windows server loses a path in a non-clustering environment, the path condition changes from open to dead, and the adapter condition changes from active to degraded. The adapter and path condition will not change until the path is made operational again. When the Windows server loses a path in a clustering environment, the path condition changes from open to dead and the adapter condition changes from active to degraded. However, after a period of time, the path condition changes back to open and the adapter condition changes back to normal, even if the path has not been made operational again.

**Tip:** The adapter status changes to a degraded state when paths are active and to a failed state when there are no active paths.

The **datapath set adapter # offline** command operates separately in a clustering environment, as compared to a non-clustering environment. In a cluster environment, the **datapath set adapter offline** command does not change the condition of the path, if the path is active, or being reserved.

## 3.4 Using Multipath Input/Output for Windows 2003 and 2008

Microsoft’s Multipath Input/Output (MPIO) solutions work in conjunction with device-specific modules (DSMs) written by vendors, but the MPIO driver package by itself, does not form a complete solution. This joint solution helps storage vendors provide device-specific solutions that are tightly integrated with the Windows operating system.

**MPIO drivers:** MPIO is not shipped with the Windows operating system. Storage vendors must pack the MPIO drivers with their own DSM. IBM Subsystem Device Driver Device Specific Module (SDDDSM) is the IBM multipath I/O solution, based on Microsoft MPIO technology. It is a device-specific module specifically designed to support IBM storage devices on Windows 2003 servers.

MPIO helps integrate a multipath storage solution with the operating system and provides the ability to use multipathing in the SAN infrastructure during the boot process for SAN boot hosts.

## 3.5 Partition alignment

Partition alignment is an important performance process that is often overlooked. Windows 2003 can become considerably degraded when working with multiple disks if the partition alignment is not set properly. Windows 2008 attempts to set it up by default, but Windows 2003 does not cover this concept. This process was noted to be required by Windows servers below 2008. Setting up the proper disk alignment can create a proper Master Boot Record, MBR, which can then allocate the proper unit size per partition created. See the following links for additional information about disk partition alignment terminology:

<http://support.microsoft.com/kb/929491>

<http://technet.microsoft.com/en-us/library/dd758814%28v=sql.100%29.aspx>

## 3.6 Installing and configuring SDDDSM in Windows 2003, 2008

The Subsystem Device Driver Device Specific Module (SDDDSM) installation is a package for DS8000 devices on the Windows Server 2003 and Windows Server 2008.

With MPIO, SDDDSM supports the multipath configuration environments in the IBM System Storage DS8000. It resides in a host system with the native disk device driver and provides the following functions:

- ▶ Enhanced data availability
- ▶ Dynamic I/O load-balancing across multiple paths
- ▶ Automatic path failover protection
- ▶ Concurrent download of licensed internal code
- ▶ Path selection policies for the host system

Also be aware of the following limitations:

- ▶ SDD is *not* supported on Windows 2008.
- ▶ For an HBA driver, SDDDSM requires the Storport version of the HBA Miniport driver.

SDDDSM can be downloaded from the following website:

<http://www.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=D430&uid=s5g1S4000350>

### 3.6.1 SDDDSM for DS8000

Ensure that SDDDSM is installed before adding additional paths to a device; otherwise, the operating system can lose the ability to access existing data on that device. You will also see repeated disks.

For details, see the *IBM System Storage Multipath Subsystem Device Driver User's Guide*, SC30-4131.

Figure 3-4 shows three disks connected by two paths to the server.

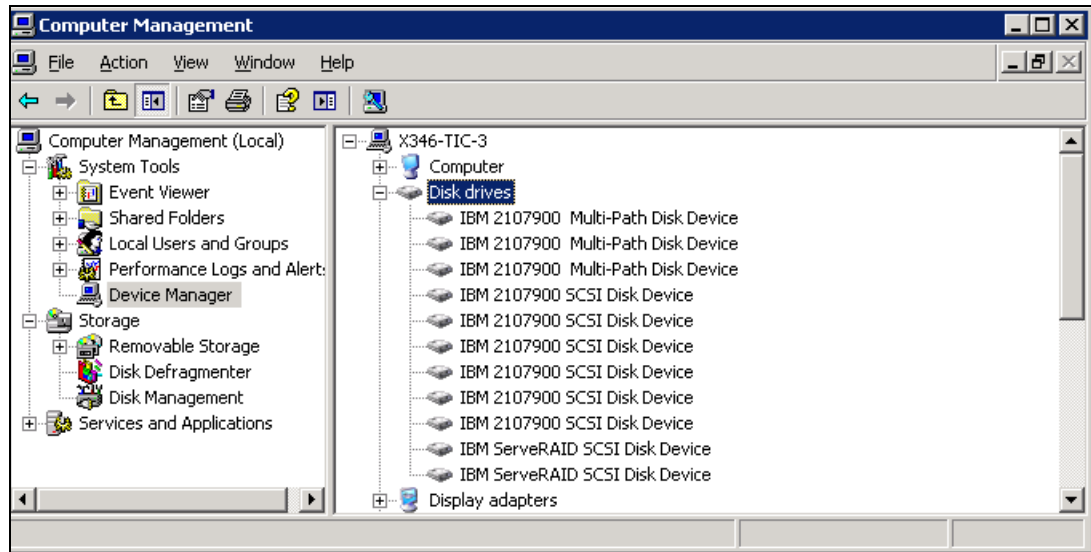


Figure 3-4 Connecting SDDDSM devices in Windows Device Manager view

In Figure 3-4, the three IBM 2107900 Multi-Path Disk Devices are shown as real disks in Windows Device Manager, under Disk drives. The IBM 2107900 SCSI Disk Device is hidden by SDDDSM. Figure 3-5 shows initialized disks in the Disk Management view.

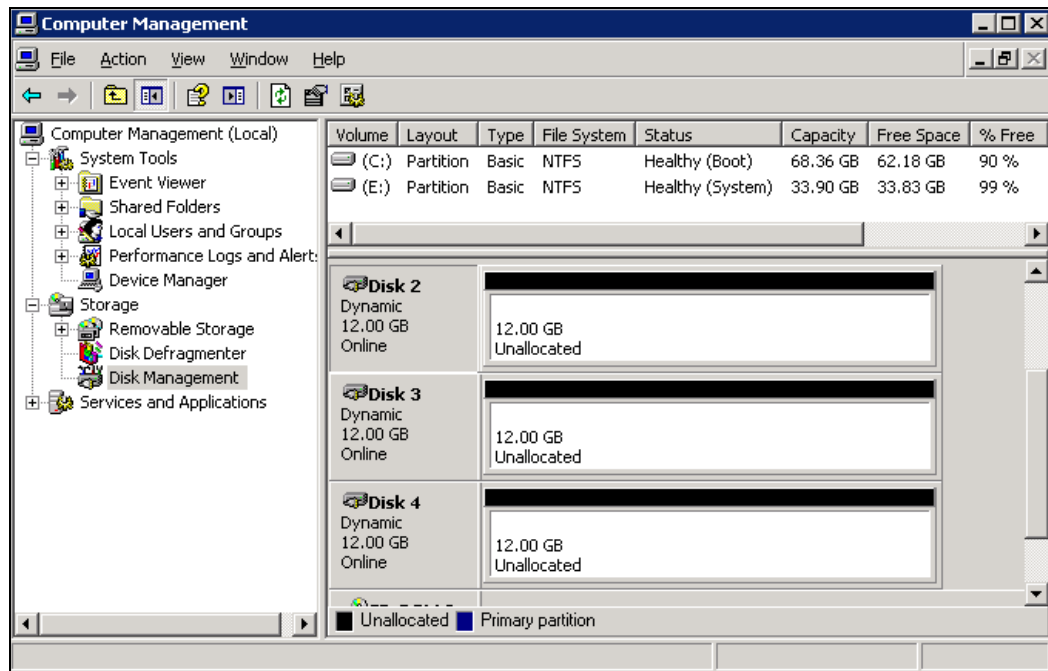


Figure 3-5 Initializing disks in the Disk Management view

**Tip:** Newly assigned disks will be discovered. If they are not discovered, go to the Disk Manager window and rescan disks, or go to the Device Manager window and scan for hardware changes.



### 3.6.2 SDDDSM datapath query

Example 3-3 shows the output of the **datapath query device** command.

*Example 3-3 datapath query device with SDDDSM*

```
C:\Program Files\IBM\SDDDSM>datapath query device
Total Devices : 3
DEV#: 0 DEVICE NAME: Disk2 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75207814703
=====
Path#          Adapter/Hard Disk          State Mode      Select  Errors
   0      Scsi Port1 Bus0/Disk2 Part0  OPEN  NORMAL    203    4
   1      Scsi Port2 Bus0/Disk2 Part0  OPEN  NORMAL    173    1
DEV#: 1 DEVICE NAME: Disk3 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75ABTV14703
=====
Path#          Adapter/Hard Disk          State Mode      Select  Errors
   0      Scsi Port1 Bus0/Disk3 Part0  OPEN  NORMAL    180    0
   1      Scsi Port2 Bus0/Disk3 Part0  OPEN  NORMAL    158    0
DEV#: 2 DEVICE NAME: Disk4 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75034614703
=====
Path#          Adapter/Hard Disk          State Mode      Select  Errors
   0      Scsi Port1 Bus0/Disk4 Part0  OPEN  NORMAL    221    0
   1      Scsi Port2 Bus0/Disk4 Part0  OPEN  NORMAL    159    0
```

All paths of the DS8000 will be used, because the DS8000 does not implement the concept of a preferred path. Additionally, the **datapath query wwpn** command (Example 3-4) shows the WWPN of your Fibre Channel adapter.

*Example 3-4 The datapath query wwpn command with SDDDSM*

```
C:\Program Files\IBM\SDDDSM>datapath query wwpn
Adapter Name      PortWWN
Scsi Port1:      210000E08B1EAE9B
Scsi Port2:      210000E08B0B8836
```

The **datapath query portmap** command provides a map of the DS8000 I/O ports and shows on which I/O ports your HBAs are connected (see Example 3-5).

*Example 3-5 The datapath query portmap command*

```
C:\Program Files\IBM\SDDDSM>datapath query portmap
          BAY-1(B1)          BAY-2(B2)          BAY-3(B3)          BAY-4(B4)
ESSID  DISK  H1 H2 H3 H4  H1 H2 H3 H4  H1 H2 H3 H4  H1 H2 H3 H4
          ABCD ABCD ABCD ABCD  ABCD ABCD ABCD ABCD  ABCD ABCD ABCD ABCD  ABCD ABCD ABCD ABCD
          BAY-5(B5)          BAY-6(B6)          BAY-7(B7)          BAY-8(B8)
          H1 H2 H3 H4  H1 H2 H3 H4  H1 H2 H3 H4  H1 H2 H3 H4
          ABCD ABCD ABCD ABCD  ABCD ABCD ABCD ABCD  ABCD ABCD ABCD ABCD  ABCD ABCD ABCD ABCD
7520781  Disk2  ---- ---- ---- ----  ---- ---- ---- ----  ---- ---- ---- ----  ---- ---- ---- ----
75ABTV1  Disk3  Y--- ---- ---- ----  ---- ---- Y--- ----  ---- ---- ---- ----  ---- ---- ---- ----
7503461  Disk4  ---- ---- ---- ----  ---- ---- ---- ----  ---- ---- ---- ----  ---- ----Y---- ----
Y = online/open          y = (alternate path) online/open
O = online/closed        o = (alternate path) online/close
N = offline              n = (alternate path) offline
- = path not configured
? = path information not available
PD = path down
Note: 2105 devices' essid has 5 digits, while 1750/2107 device's essid has 7 digits.
```

The **datapath query essmap** command provides additional information about LUNs and I/O port numbers, as shown in Example 2-6.

*Example 3-6 datapath query essmap*

```
C:\Program Files\IBM\SDDDSM>datapath query essmap
```

Disk	Path	P	Location	LUN SN	Type	Size	LSS	Vol	Rank	C/A	S	Connection	Port	RaidMode
Disk2	Path0	Port1	Bus0	75207814703	IBM 2107900	12.0GB	47	03	0000	2c	Y	R1-B2-H4-ZD	143	RAID5
Disk2	Path1	Port2	Bus0	75207814703	IBM 2107900	12.0GB	47	03	0000	2c	Y	R1-B4-H4-ZD	343	RAID5
Disk3	Path0	Port1	Bus0	75ABTV14703	IBM 2107900	12.0GB	47	03	0000	0b	Y	R1-B1-H1-ZA	0	RAID5
Disk3	Path1	Port2	Bus0	75ABTV14703	IBM 2107900	12.0GB	47	03	0000	0b	Y	R1-B2-H3-ZA	130	RAID5
Disk4	Path0	Port1	Bus0	75034614703	IBM 2107900	12.0GB	47	03	0000	0e	Y	R1-B2-H4-ZD	143	RAID5
Disk4	Path1	Port2	Bus0	75034614703	IBM 2107900	12.0GB	47	03	0000	0e	Y	R1-B4-H2-ZD	313	RAID5

### 3.6.3 Windows 2008 and SDDDSM

Windows 2008 only supports SDDDSM. The datapath query commands in Windows 2008 have not been changed, in comparison to Windows 2003. The Server Manager window has changed slightly on Windows 2008 servers, as shown in Figure 3-6.

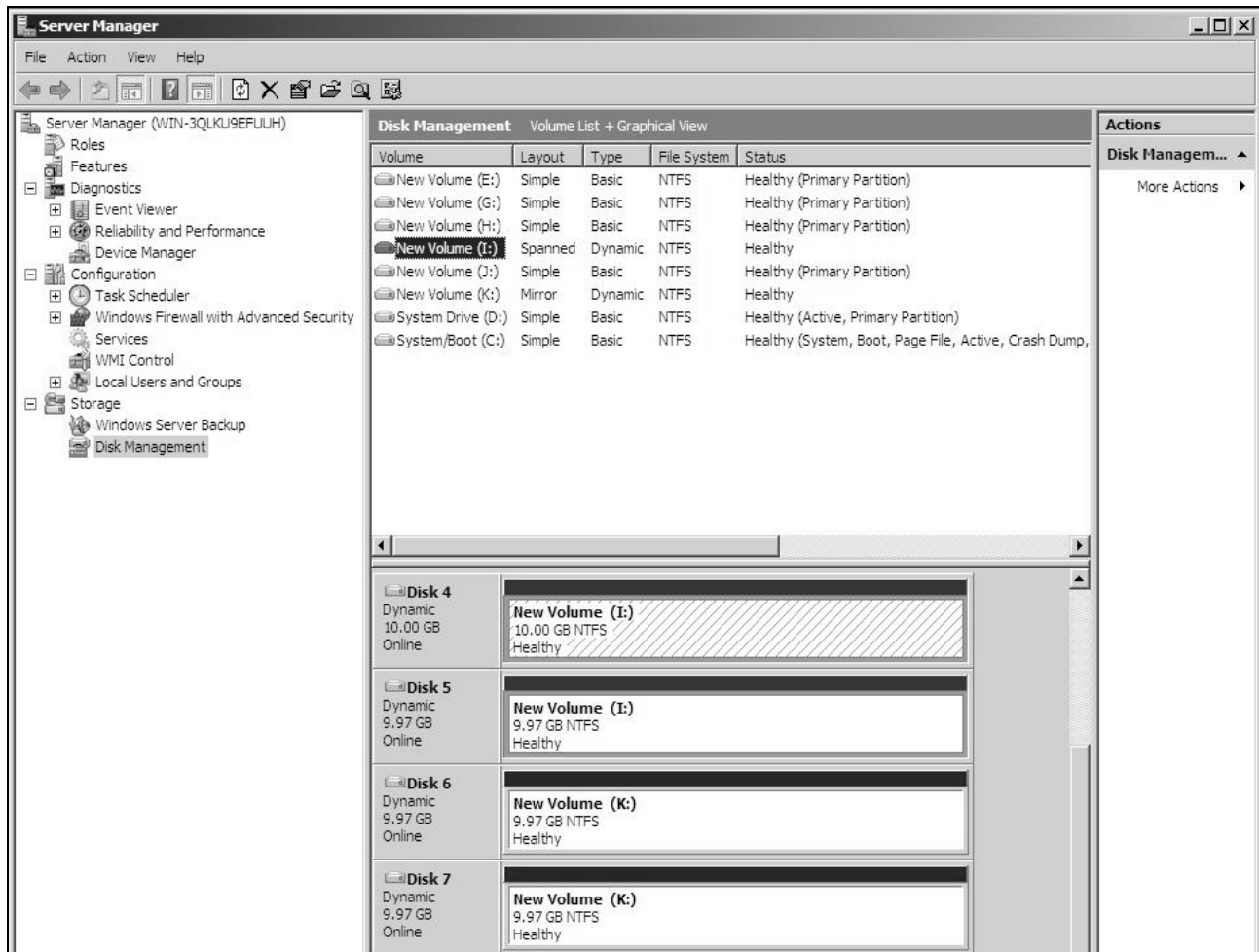


Figure 3-6 Server Manager view

## 3.7 Expanding dynamic disk for Windows 2003, Windows 2008

It is possible to expand a volume in the DS8000, even if it is mapped to a host. Operating systems, such as Windows 2003 and Windows 2008, can handle the volumes being expanded, even if the host has applications running. A volume that is in an IBM FlashCopy®, Metro Mirror, or Global Mirror relationship cannot be expanded unless the relationship is removed. Before you expand the volume, remove FlashCopy, Metro Mirror, or Global Mirror on that volume.

If the volume is part of a Microsoft Cluster Service (MSCS), Microsoft advises that you shut down all nodes except one. Applications in the resource using the volume that is going to be expanded must be stopped before expanding the volume. However, applications running in other resources can continue. After expanding the volume, restart the applications and the resource, and then restart the other nodes in the MSCS.

To expand a volume while it is in use on Windows 2003 and Windows 2008, use either *DiskPart* for basic disks or *Disk Manager* for dynamic disks. The DiskPart tool is part of Windows 2003. For other Windows versions, you can download it at no cost from Microsoft.

DiskPart is a tool developed to ease administration of storage. It is a command-line interface (CLI) where you can manage disks, partitions, and volumes, using scripts or direct input on the command line. You can list disks and volumes, select them, and, after selecting them, obtain more detailed information, create partitions, extend volumes, and more. For more information, see the following Microsoft websites:

- ▶ Microsoft home page:  
<http://www.microsoft.com>
- ▶ Microsoft support page:  
<http://support.microsoft.com/default.aspx?scid=kb;en-us;304736&sd=tech>

To list the volume size, use the **lsfbvol** command, as shown in Example 3-7.

### Example 3-7 *lsfbvol -fullid* before volume expansion

```
dscli> lsfbvol -fullid 4703
Name          ID          accstate  datastate  configstate  deviceMTM  datatype  extpool          cap(2^30B)  cap(10^9B)  cap(blocks)
-----
ITS0_x346_3_4703  IBM.2107-7520781/4703 Online    Normal    Normal      2107-900  FB 512  IBM.2107-7520781/P53  12.0      -      25165824
```

In Example 3-7 here, you can see that the capacity is 12 GB, and also what the volume ID is.

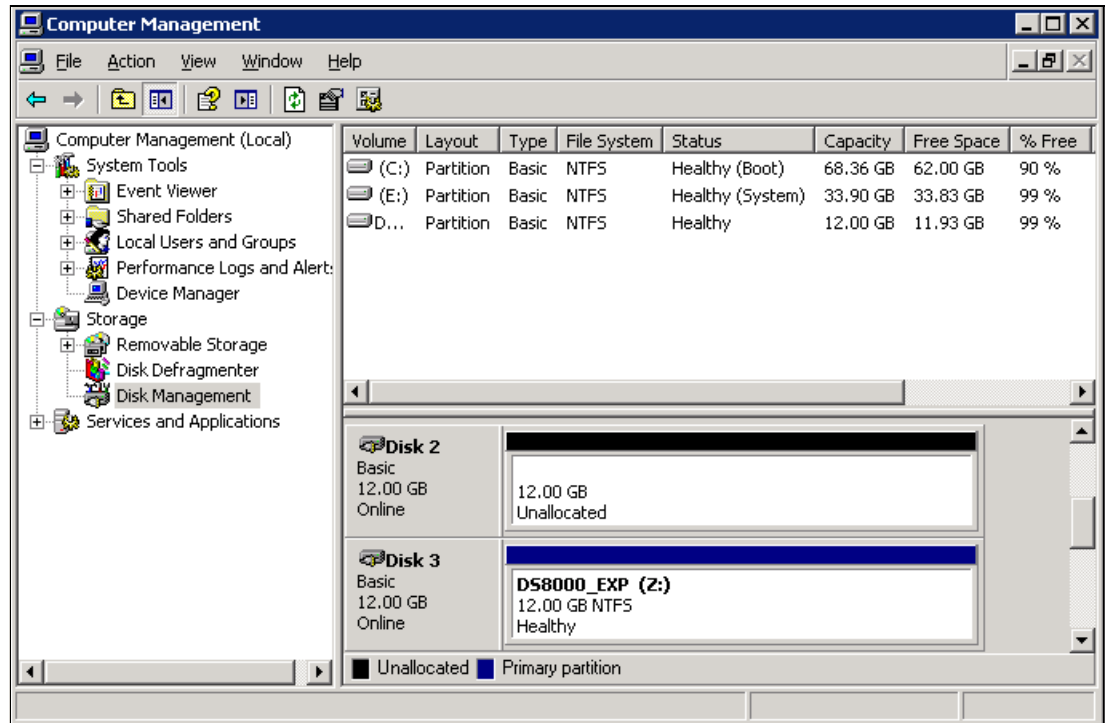
To discover what disk this volume is on in the Windows 2003 host, use the **datapath query device SDDDSM** command, as shown in Example 3-8.

*Example 3-8 The datapath query device command before expansion*

```
C:\Program Files\IBM\SDDDSM>datapath query device
Total Devices : 3
DEV#: 0 DEVICE NAME: Disk2 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75034614703
=====
Path#          Adapter/Hard Disk          State Mode      Select  Errors
  0      Scsi Port1 Bus0/Disk2 Part0  OPEN  NORMAL    42      0
  1      Scsi Port2 Bus0/Disk2 Part0  OPEN  NORMAL    40      0
DEV#: 1 DEVICE NAME: Disk3 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75207814703
=====
Path#          Adapter/Hard Disk          State Mode      Select  Errors
  0      Scsi Port1 Bus0/Disk3 Part0  OPEN  NORMAL   259      0
  1      Scsi Port2 Bus0/Disk3 Part0  OPEN  NORMAL   243      0
DEV#: 2 DEVICE NAME: Disk4 Part0 TYPE: 2107900 POLICY: OPTIMIZED
SERIAL: 75ABTV14703
=====
Path#          Adapter/Hard Disk          State Mode      Select  Errors
  0      Scsi Port1 Bus0/Disk4 Part0  OPEN  NORMAL    48      0
  1      Scsi Port2 Bus0/Disk4 Part0  OPEN  NORMAL    34      0
```

In Example 3-8, you can see that the volume with ID 75207814703 is Disk3 on the Windows host because the volume ID matches the SERIAL.

You can see the size of the volume by using Disk Management (Figure 3-7).



*Figure 3-7 Volume size before expansion on Windows 2003: Disk Manager view*

Figure 3-8 shows the volume size before expansion on the Disk Properties view in Windows 2003.

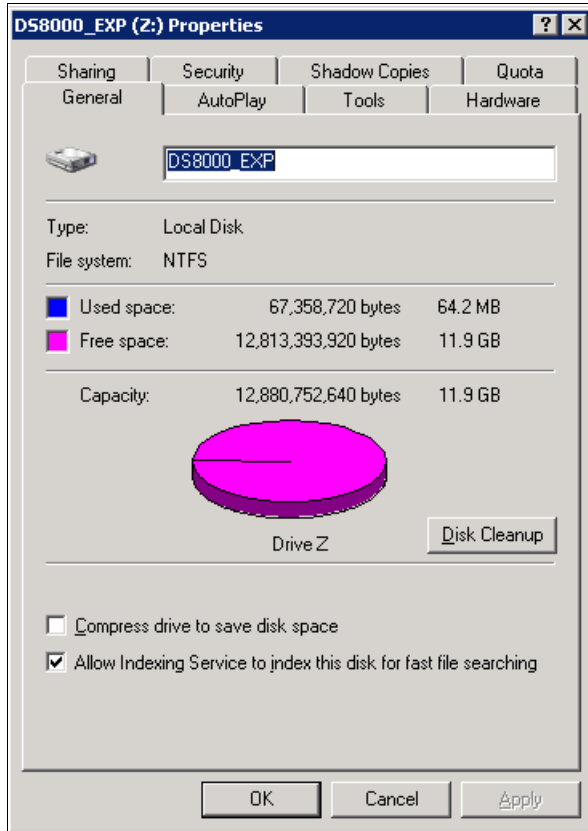


Figure 3-8 Volume size in Disk Properties view

The volume size in Figure 3-7 and Figure 3-8 is 11.99 GB, which rounded up is 12 GB.

To expand this volume on the DS8000, use the **chfbvol** command, as shown in Example 3-9.

*Example 3-9 Expanding a volume*

```
dscli> chfbvol -cap 18 4703
CMUC00332W chfbvol: Some host operating systems do not support changing the volume
size. Are you sure that you want to resize the volume? [y/n]: y
CMUC00026I chfbvol: FB volume 4703 successfully modified.
```

The new capacity must be larger than the previous one; you *cannot* shrink the volume.

To ensure that the volume was expanded, use the **lsfbvol** command, as shown in Example 3-10. In this example, you can see that the volume with the ID of 4703 was expanded to 18 GB in capacity.

*Example 3-10 lsfbvol after expansion*

```
dscli> lsfbvol 4703
Name          ID  accstate  datastate  configstate  deviceMTM  datatype  extpool  cap (2^30B)  cap (10^9B)  cap (blocks)
-----
ITS0_x346_3_4703 4703 Online    Normal    Normal      2107-900  FB 512    P53         18.0         -            37748736
```

In Disk Management, perform a rescan for the disks to see the new capacity for disk1. Figure 3-9 shows the disks available after a rescan is performed.

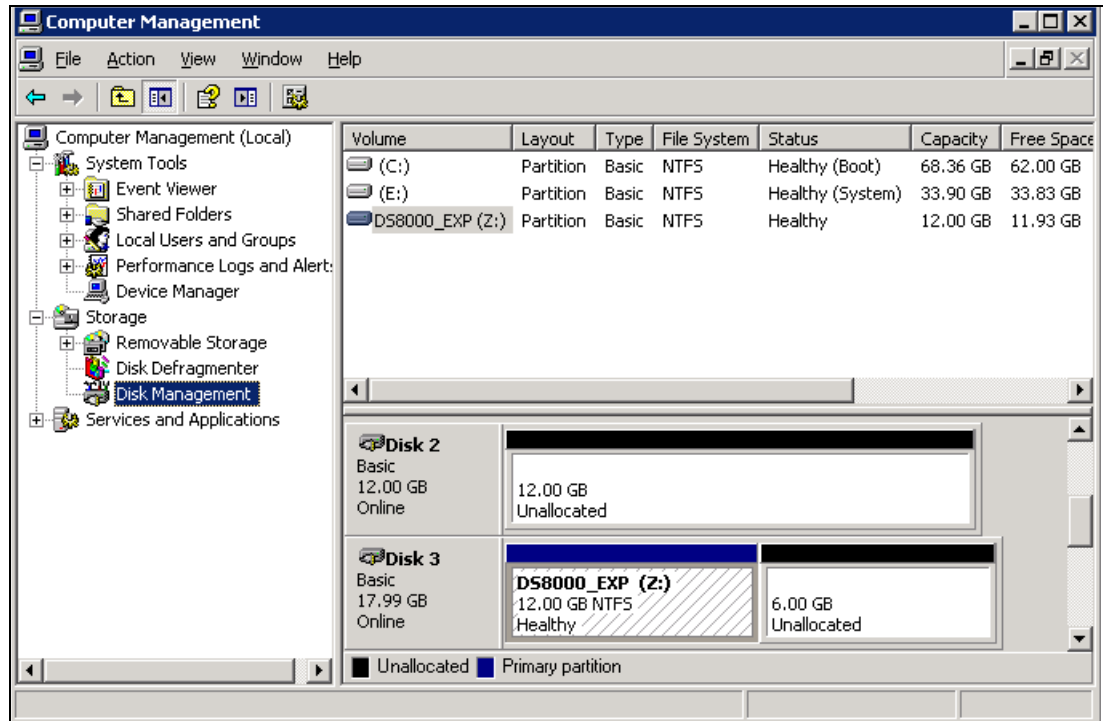


Figure 3-9 Expanded volume in Disk Manager

Figure 3-9 also shows that Disk3 now has 6 GB of unallocated new capacity. To make this capacity available for the file system, use the `diskpart` command.

Using the `diskpart` command, list the volumes, select the pertinent volume, check the size of the volume, extend the volume, and check the size again, as shown in Example 3-11.

*Example 3-11 The diskpart command*

```
C:\Documents and Settings\Administrator>diskpart
```

```
Microsoft DiskPart version 5.2.3790.3959
Copyright (C) 1999-2001 Microsoft Corporation.
On computer: X346-TIC-3
```

```
DISKPART> list volume
```

Volume ###	Ltr	Label	Fs	Type	Size	Status	Info
Volume 0	Z	DS8000_EXP	NTFS	Partition	12 GB	Healthy	
Volume 1	E		NTFS	Partition	34 GB	Healthy	System
Volume 2	D			DVD-ROM	0 B	Healthy	
Volume 3	C		NTFS	Partition	68 GB	Healthy	Boot

```
DISKPART> select volume 0
```

Volume 0 is the selected volume.

```
DISKPART> detail volume
```

```

Disk ### Status      Size      Free      Dyn  Gpt
-----
* Disk 3  Online      18 GB    6142 MB

```

```

Readonly      : No
Hidden        : No
No Default Drive Letter: No
Shadow Copy   : No

```

DISKPART> **extend**

DiskPart successfully extended the volume.

DISKPART> **detail volume**

```

Disk ### Status      Size      Free      Dyn  Gpt
-----
* Disk 3  Online      18 GB      0 B

```

```

Readonly      : No
Hidden        : No
No Default Drive Letter: No
Shadow Copy   : No

```

Figure 3-10 shows the Disk Manager view of the result for the volume expansion.

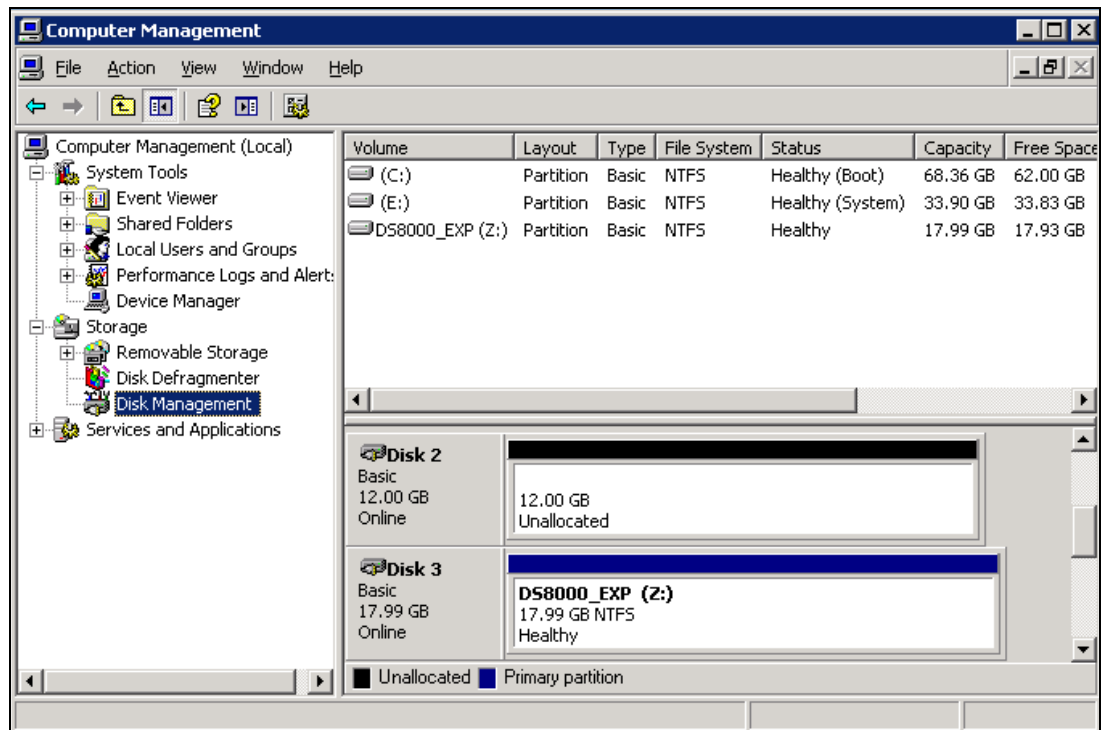


Figure 3-10 Disk Manager after expansion in diskpart

The disk storage expanded in this example is referred to as a basic disk. Dynamic disks can be expanded by expanding the underlying DS8000 volume. The new space is displayed as unallocated space at the end of the disk.

In this case, you do not need to use the Diskpart tool. Use the Windows Disk Management functions to allocate the new space. Expansion works and is not dependent on the volume type, for example, simple, spanned, mirrored, and so on, on the disk. Dynamic disks can be expanded without stopping the I/O in most cases.

**Important:** Always back up your data before upgrading your basic disk to dynamic disk, or in reverse order, as this operation is disruptive to the data due to the separate position of the logical block address (LBA) in the disks.

## 3.8 SAN boot support

When booting from the Fibre Channel storage systems, special restrictions apply:

- ▶ With Windows Server 2003 and 2008, MSCS uses target resets. See the Microsoft technical article “Microsoft Windows Clustering: *Storage Area Networks*”, at the following website:  
<http://support.microsoft.com/kb/818668>
- ▶ Windows Server 2003 and 2008 enable boot disk and the cluster server disks to be hosted on the same bus. However, you need to use Storport Miniport HBA drivers for this functionality to work. The boot disk and cluster server disks hosted on the same bus is *not* a supported configuration in combination with drivers of other types, for example, SCSIport Miniport or full port drivers.
- ▶ If you reboot a system with adapters while the primary path is in a failed state, manually disable the BIOS on the first adapter and manually enable the BIOS on the second adapter. You cannot enable the BIOS for both adapters at the same time. If the BIOS for both adapters are enabled at the same time and there is a path failure on the primary adapter, the system stops with an INACCESSIBLE\_BOOT\_DEVICE error upon reboot.

## 3.9 Windows Server 2003 Virtual Disk Service support

With Windows Server 2003, Microsoft introduced the *virtual disk service (VDS)*. It unifies storage management and provides a single interface for managing block storage virtualization. This interface is vendor and technology transparent and is independent of the layer of the operating system software, RAID storage hardware, and other storage virtualization engines, where virtualization is done.

VDS is a set of APIs that uses two sets of providers to manage storage devices. The built-in *VDS software providers* enable you to manage disks and volumes at the operating system level. *VDS hardware providers* supplied by the hardware vendor enable you to manage hardware RAID arrays. Windows Server 2003 and 2008 components that work with VDS, include the Disk Management Microsoft Management Console (MMC) snap-in, the DiskPart command-line tool, and the DiskRAID command-line tool, which is available in the Windows Server 2003 and 2008 deployment kit.



Figure 3-11 shows the VDS architecture.

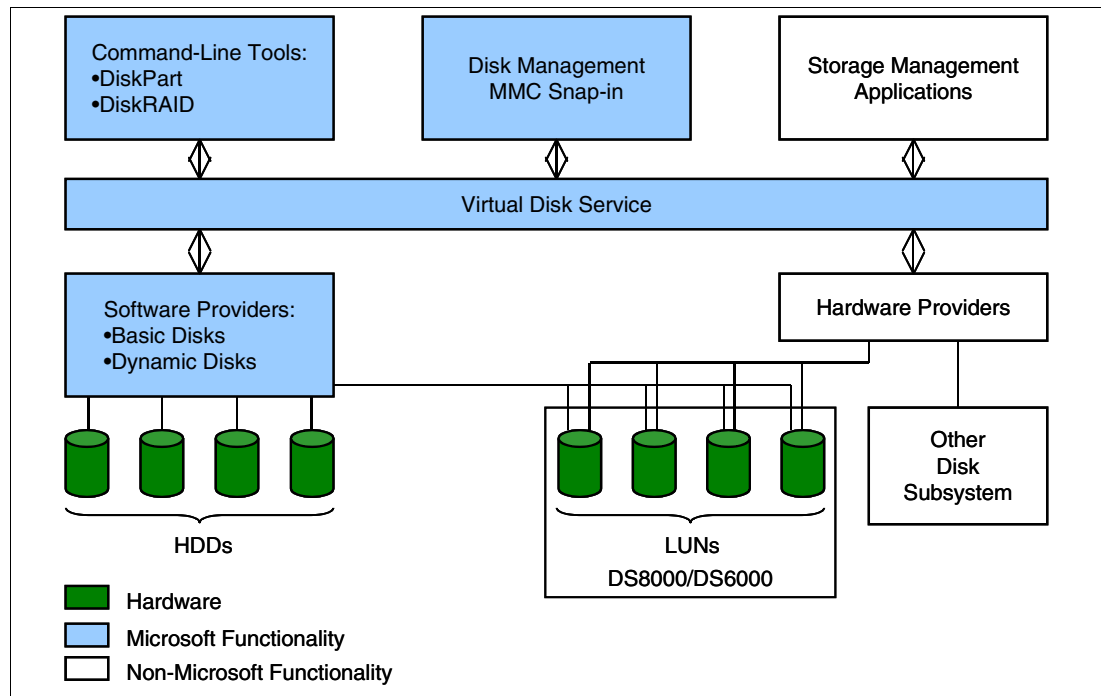


Figure 3-11 Example of Microsoft VDS architecture

For a detailed description of VDS, see the *Microsoft Windows Server 2003 Virtual Disk Service Technical Reference*, found at the following website:

[http://technet.microsoft.com/en-us/library/cc776012\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc776012(WS.10).aspx)

The DS8000 can act as a VDS hardware provider. The implementation is based on the DS8000 common information model (CIM) agent, a middleware application that provides a CIM-compliant interface. The VDS uses CIM technology to list information and manage LUNs. For information about how to install and configure VDS support, see the *IBM System Storage DS Open Application Programming Interface Reference*, GC35-0516.

### 3.9.1 VDS integration with DS8000 storage subsystems

The following sections present examples of VDS integration with advanced functions of the DS8000 storage subsystems when leveraging the DS CIM agent.

### 3.9.2 Volume Shadow Copy Service

The *Volume Shadow Copy Service*, also known as *Volume Snapshot Service (VSS)*, provides a mechanism for creating consistent real-time copies of data, known as *shadow copies*. It integrates IBM System Storage FlashCopy to produce consistent shadow copies, while also coordinating with business applications, file system services, backup applications, and fast recovery solutions.

For more information, see the following website:

<http://msdn.microsoft.com/en-us/library/aa384649%28v=vs.85%29.aspx>

### 3.9.3 Required components

To use VSS functions, you need an installed CIM client. This CIM client requires the IBM System Storage DS® command-line interface (CLI) client to communicate with the DS8000 (or ESSCLI for an ESS). On each server, the IBM API support is required for Microsoft Volume Shadow Copy Service, as shown in Figure 3-12.

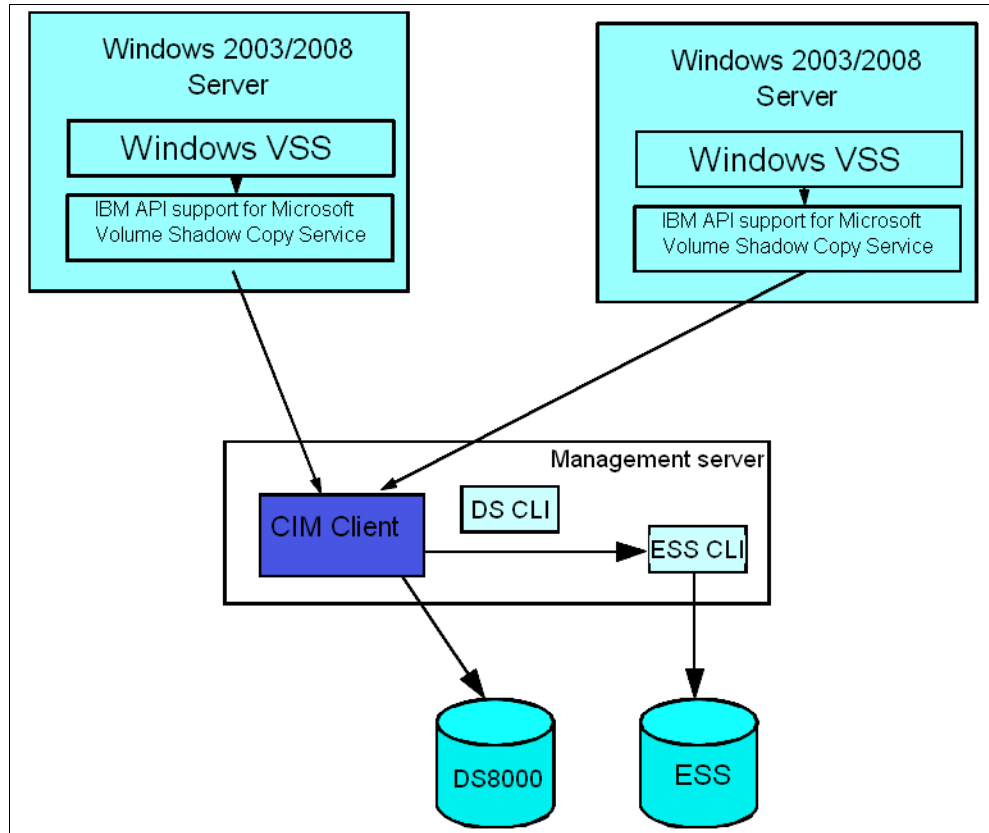


Figure 3-12 VSS installation infrastructure

After the installation of these components, which is described in *IBM System Storage DS Open Application Programming Interface Reference*, GC35-0516, complete the following steps:

1. Define a VSS\_FREE volume group and virtual server.
2. Define a VSS\_RESERVED volume group and virtual server.
3. Assign volumes to the VSS\_FREE volume group.

The WWPN default for the VSS\_FREE virtual server is 50000000000000; the WWPN default for the VSS\_RESERVED virtual server is 50000000000001. These disks are available for the server as a pool of free available disks. If you want to have separate pools of free disks, you can define your own WWPN for another pool, as shown in Example 3-12.

*Example 3-12 ESS Provider Configuration Tool Commands help*

---

```
C:\Program Files\IBM\ESS Hardware Provider for VSS>ibmvssconfig.exe /?
```

```
ESS Provider Configuration Tool Commands
```

```
-----  
ibmvssconfig.exe <command> <command arguments>
```

```
Commands:
```

```
/h | /help | -? | /?
```

```
showcfg
```

```
listvols <all|free|vss|unassigned>
```

```
add <volumeID list> (separated by spaces)
```

```
rem <volumeID list> (separated by spaces)
```

```
Configuration:
```

```
set targetESS <5-digit ESS Id>
```

```
set user <CIMOM user name>
```

```
set password <CIMOM password>
```

```
set trace [0-7]
```

```
set trustpassword <trustpassword>
```

```
set truststore <truststore location>
```

```
set usingSSL <YES | NO>
```

```
set vssFreeInitiator <WWPN>
```

```
set vssReservedInitiator <WWPN>
```

```
set FlashCopyVer <1 | 2>
```

```
set cimomPort <PORTNUM>
```

```
set cimomHost <Hostname>
```

```
set namespace <Namespace>
```

---

With the **ibmvssconfig.exe listvols** command, you can also verify what volumes are available for VSS in the VSS\_FREE pool, as shown in Example 3-13.

*Example 3-13 VSS list volumes at free pool*

---

```
C:\Program Files\IBM\ESS Hardware Provider for VSS>ibmvssconfig.exe listvols free
```

```
Listing Volumes...
```

LSS	Volume	Size	Assigned to
10	003AAGXA	5.3687091E9 Bytes	5000000000000000
11	103AAGXA	2.14748365E10 Bytes	5000000000000000

---

Also, disks that are unassigned in your disk subsystem can be assigned with the **add** command to the VSS\_FREE pool. Example 3-14 lists the volumes available for VSS.

*Example 3-14 VSS list volumes available for VSS*

---

```
C:\Program Files\IBM\ESS Hardware Provider for VSS>ibmvssconfig.exe listvols vss
```

```
Listing Volumes...
```

LSS	Volume	Size	Assigned to
10	001AAGXA	1.00000072E10 Bytes	Unassigned
10	003AAGXA	5.3687091E9 Bytes	5000000000000000
11	103AAGXA	2.14748365E10 Bytes	5000000000000000

---

## 3.10 Hyper-V considerations

This section provides information for the specifics of attaching IBM System Storage DS8000 systems to host systems running Microsoft Hyper-V Server 2008 R2.

This section includes the following topics:

- ▶ Hyper-V introduction
- ▶ Storage concepts for virtual machines
- ▶ Cluster Shared Volumes (CSV)
- ▶ Best practices

### 3.10.1 Hyper-V introduction

Hyper-V is Microsoft platform for x86 based server virtualization. Hyper-V exists in two versions:

- ▶ As Microsoft Hyper-V Server 2008 R2:

This version provides a core installation of Microsoft Windows Server 2008 with Hyper-V functionality. Other Windows Server roles are not included and it comes with limited Windows Services for management. This version is free and can be downloaded at the following website:

<http://www.microsoft.com/en-us/download/details.aspx?id=20196>

- ▶ As a server role in Microsoft Windows Server 2008 R2:

Hyper-V can be added as a server role in Windows Server 2008 R2. It provides the same virtualization functionality as the plain Hyper-V Server, but also provides a full Windows Server 2008 including other server roles (such as web server, Domain controller, and so on) and a build in graphical management solution called “Hyper-V Manager”.

Hyper-V uses the term “partition” for virtual machines. The Hyper-V (Hyper Visor) requires at least one *parent partition*, which hosts Windows Server 2008. Inside this parent partition the virtualization software is running which controls access of the *child partitions* to the physical hardware. The child partitions have virtual hardware components that are dynamically mapped through the *virtual machine bus (VM bus)* to the physical hardware by the parent partition.

Figure 3-13 shows the architecture of Hyper-V.

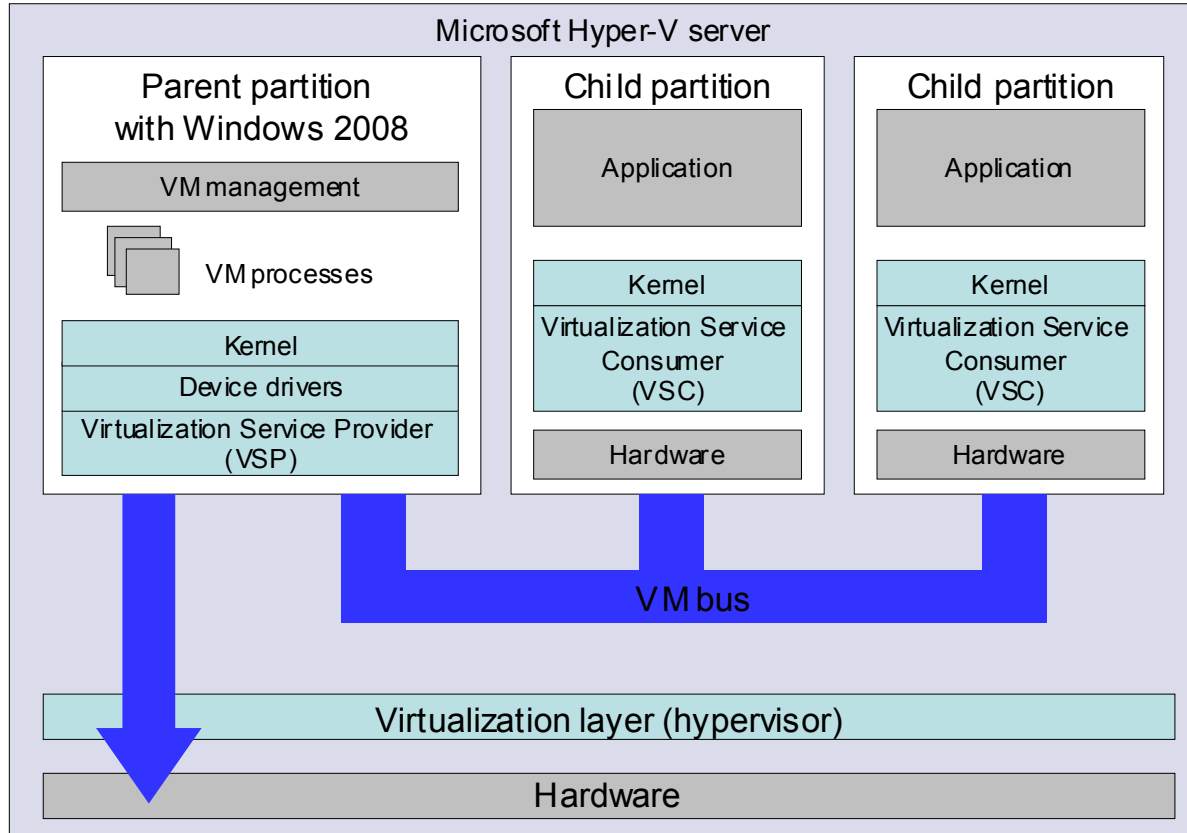


Figure 3-13 Microsoft Hyper-V architecture

### 3.10.2 Storage concepts for virtual machines

Virtual machines consist of several files, which can contain the data of one or more virtual hard drives, saved memory, saved device states, and so on. All these files are stored on NTFS partitions that are managed by the Windows Server 2008 inside the parent partition. The NTFS partitions can reside on any of these storage devices:

- ▶ Local hard drives
- ▶ iSCSI LUNs
- ▶ Fibre Channel LUNs

For information about how to create an NTFS partition on a Windows Server host, see Chapter 3, "Windows considerations" on page 17.

Hyper-V offers two different ways to provide disk storage to virtual machines:

- ▶ Virtual hard disks (VHD):  
VHDs are files that reside on an NTFS partition. They offer extended features such as snapshots. The maximum size of a VHD is 2 TiB.
- ▶ Pass-through disks:  
Pass-through disks are physical disks (either local or SAN LUNs) that are exclusively given to a virtual machine. Access is routed through the parent partition.

The storage concept of Hyper-V is outlined in Figure 3-14.

Virtual machines can be equipped with either virtual IDE adapters or virtual SCSI adapters. The differences are shown in Table 3-1.

Table 3-1 virtual adapter types

Adapter type	Maximum number of adapters	Maximum number of disk drives per adapter	Can a virtual machine boot from this adapter?
virtual IDE	2	2	yes
virtual SCSI	4	64	no

**Tip:** Hard disks can only be added to IDE adapters if the virtual machine is powered off.

If a special software package called *integration services* is installed inside the virtual machine, both virtual IDE and virtual SCSI adapters provide the same performance. The integration services are required for the virtual SCSI adapters because they include the drivers for the virtual adapters.

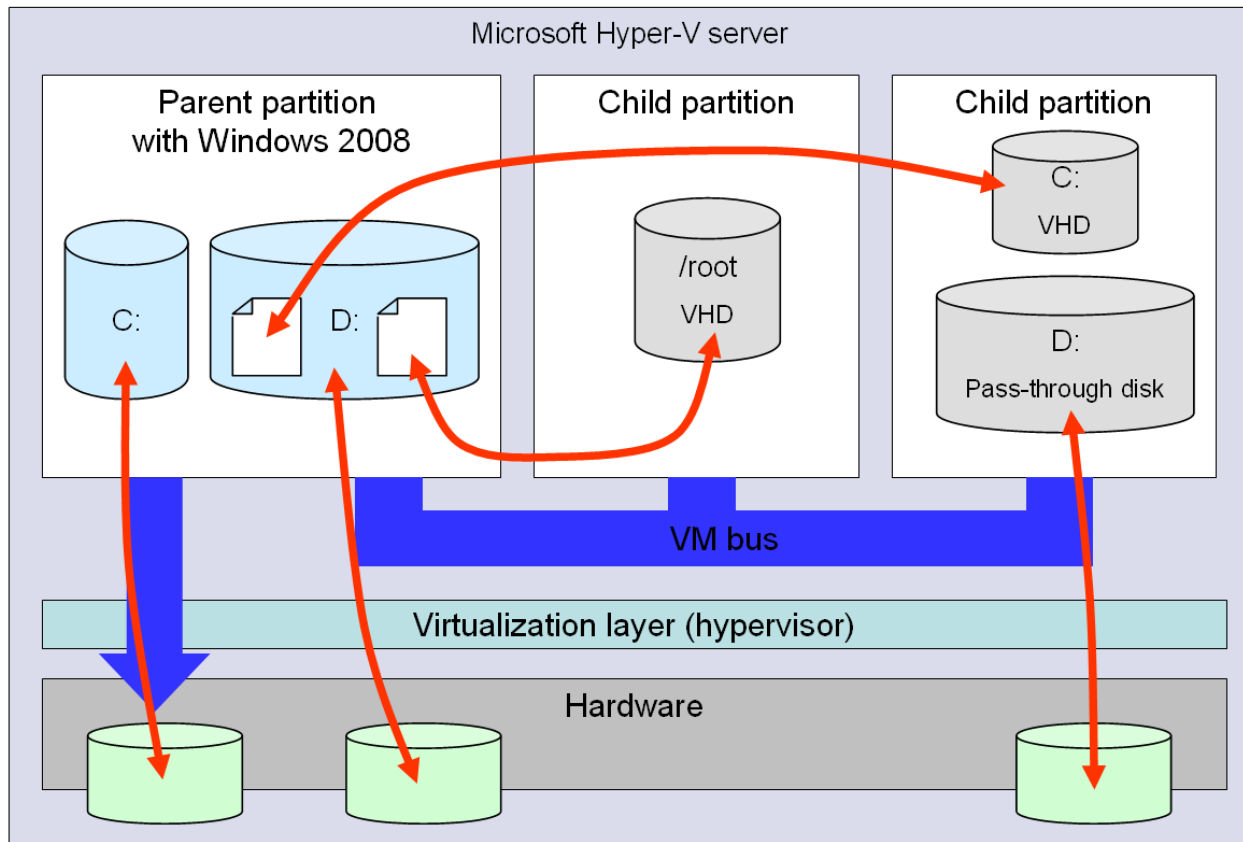


Figure 3-14 Hyper-V storage concept

### 3.10.3 Assigning a VHD to a virtual machine

The following steps are required to attach a new VHD to a virtual machine:

1. In Hyper-V manager, select the virtual machine, right-click, and select **Settings**.

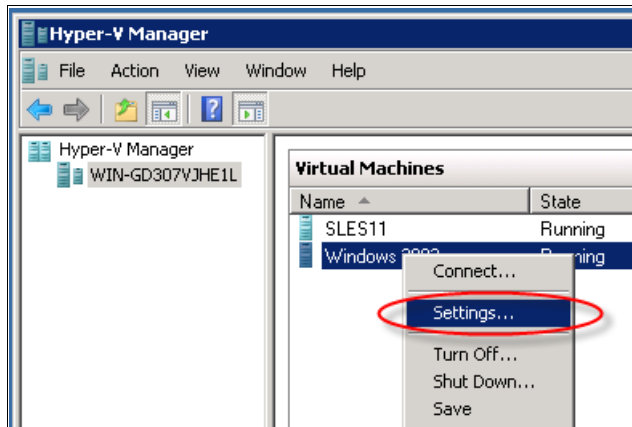


Figure 3-15 Virtual machine settings

2. Select the controller to which the new VHD needs to be attached, select **Hard Drive**, and click **Add**.

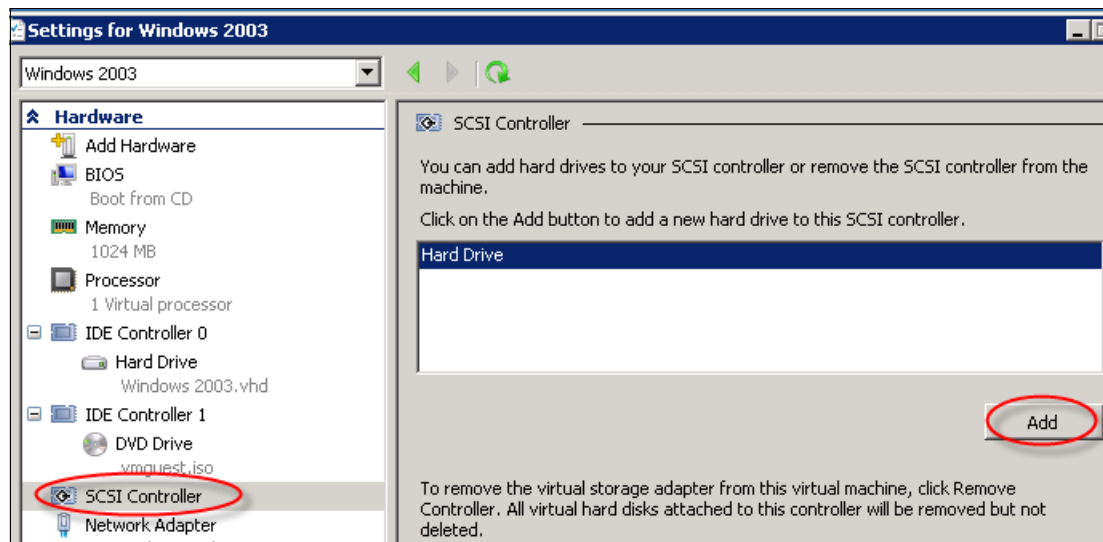


Figure 3-16 Add hard drive

3. To add a new hard drive, click **New**, as shown in Figure 3-17.

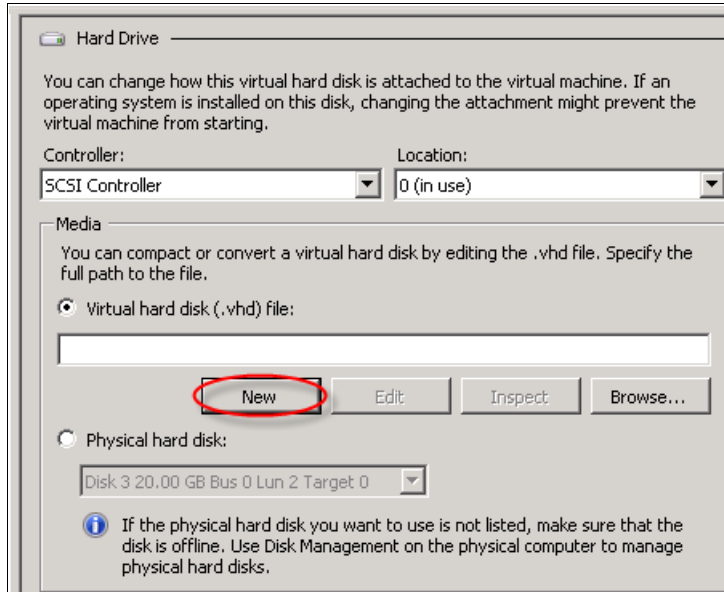


Figure 3-17 New hard drive properties

4. The **New Virtual Hard Disk** wizard is launched. Click **Next** on the initial panel.

Select the type of virtual hard disk and click **Next**, the available hard disk types are shown in Figure 3-18:

a. Fixed size:

The space required for the VHD is allocated during creation. This type is preferable for production workloads.

b. Dynamically expanding:

The capacity for the VHD is allocated when the virtual machine writes to it. This type provides a type of thin provisioning.

c. Differencing:

This VHD is based on an existing virtual disk called *parent*. The parent disk is read-only, and all changes to the differencing VHD are written into a separate file. This type offers the possibility to revert changes.

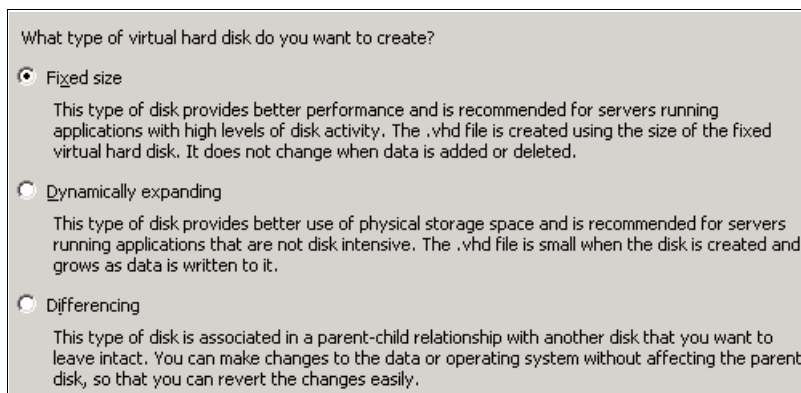
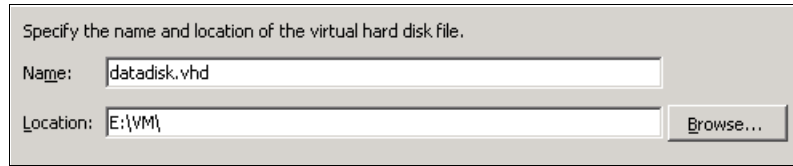


Figure 3-18 Virtual hard disk type



5. Enter a filename for the new VHD and specify where to store it. Click **Next**. See Figure 3-19.



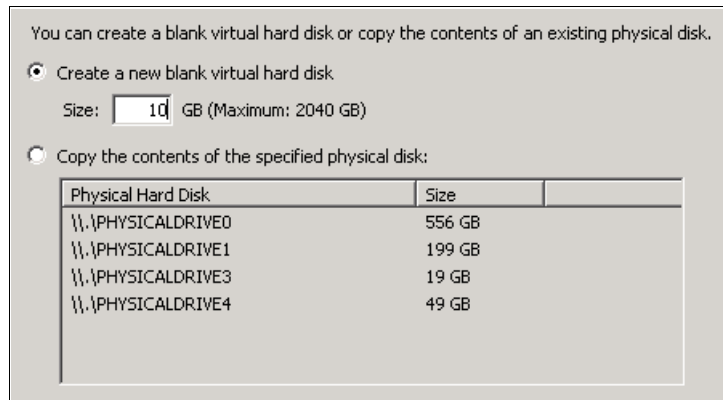
Specify the name and location of the virtual hard disk file.

Name:

Location:

Figure 3-19 Filename and location for VHD

6. Specify the size of the VHD or choose an existing physical disk from which to clone the VHD. See Figure 3-20.



You can create a blank virtual hard disk or copy the contents of an existing physical disk.

Create a new blank virtual hard disk

Size:  GB (Maximum: 2040 GB)

Copy the contents of the specified physical disk:

Physical Hard Disk	Size
\\.\PHYSICALDRIVE0	556 GB
\\.\PHYSICALDRIVE1	199 GB
\\.\PHYSICALDRIVE3	19 GB
\\.\PHYSICALDRIVE4	49 GB

Figure 3-20 VHD size

**Tip:** To virtualize a physical host, it is possible to create the new VHD as a copy of an existing physical disk by selecting **Copy the contents...**

7. On the summary page, click **Finish**.

**Attention:** If a VHD is removed from a virtual machine, the VHD file is not automatically deleted. To free up the space occupied, delete it manually from the partition.

As illustrated in Figure 3-13 on page 37, the Windows Server 2008 used for management of the Hyper-V setup is also running a virtual machine. Therefore it is also possible to attach a VHD to it.

Follow these steps:

1. Inside the disk management, right-click **More Actions** and select **Create VHD**. See Figure 3-21.

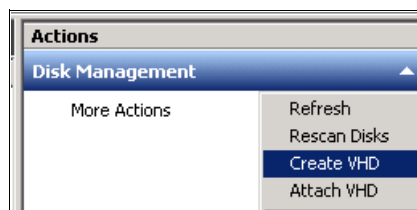


Figure 3-21 Create VHD

- Specify location and filename, type, and size.
- The new VHD will then be listed in disk management of the Windows Server 2008 host. See Figure 3-22.

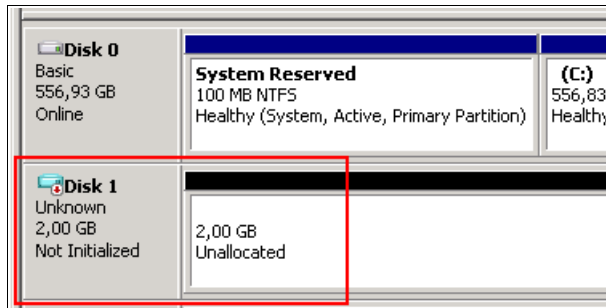


Figure 3-22 VHD in disk management

This feature provides an option for the Windows Server 2008 to benefit from advanced functions of VHDs such as snapshots or dynamic expansion.

### 3.10.4 Assigning a pass-through disk to a virtual machine

The following steps are required to attach a physical disk or LUN as a pass-through disk to a virtual machine:

- In the disk management of the Windows 2008 server, ensure that the LUN is visible and in state **offline**, as shown in Figure 3-23.

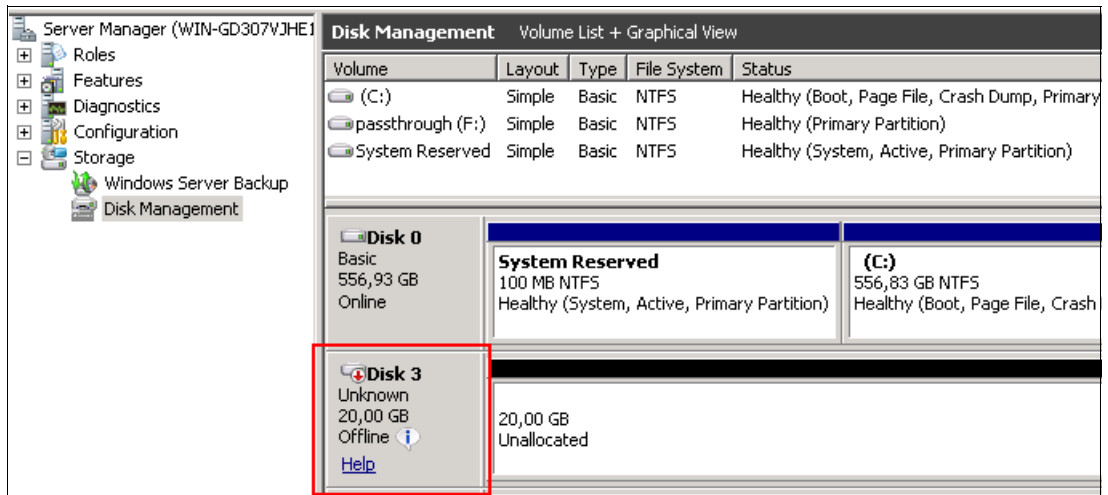


Figure 3-23 Disk management

2. In Hyper-V manager select the virtual machine, right-click and select **Settings**.

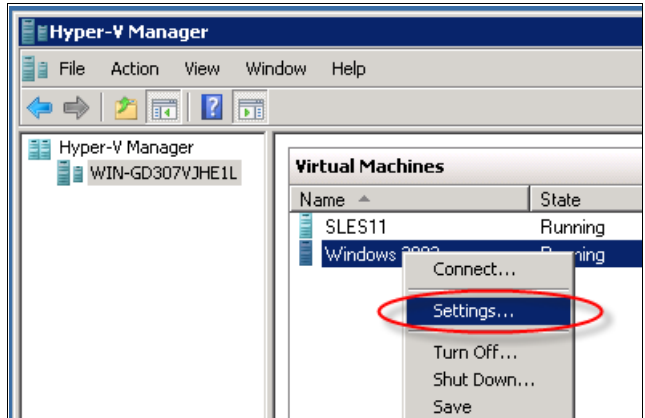


Figure 3-24 Virtual machine settings

3. Select the controller to which the new VHD needs to be attached, select **Hard Drive**, and click **Add**.

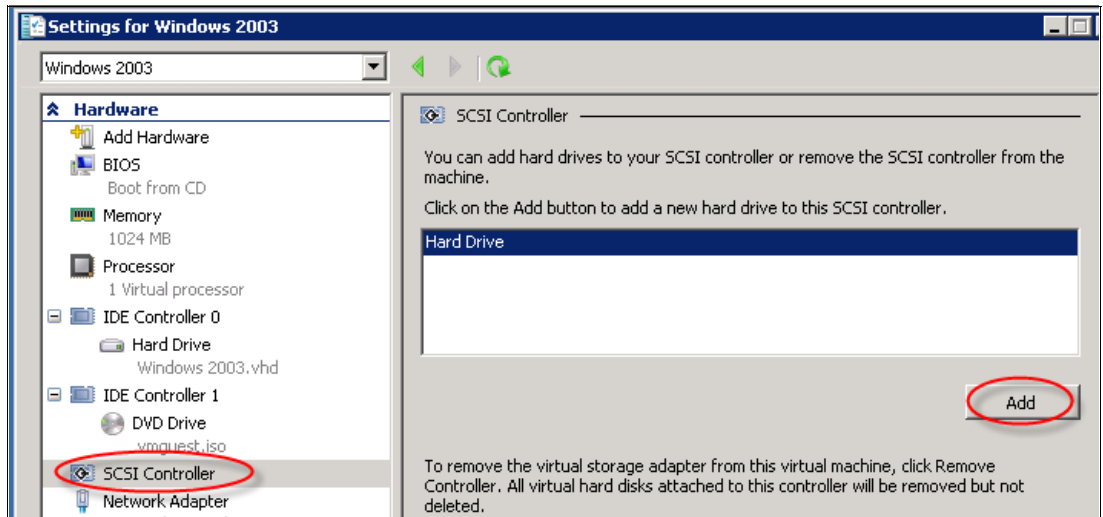


Figure 3-25 Add hard drive

4. Select **Physical hard disk** and select the disk from the drop-down list, then click **OK**.

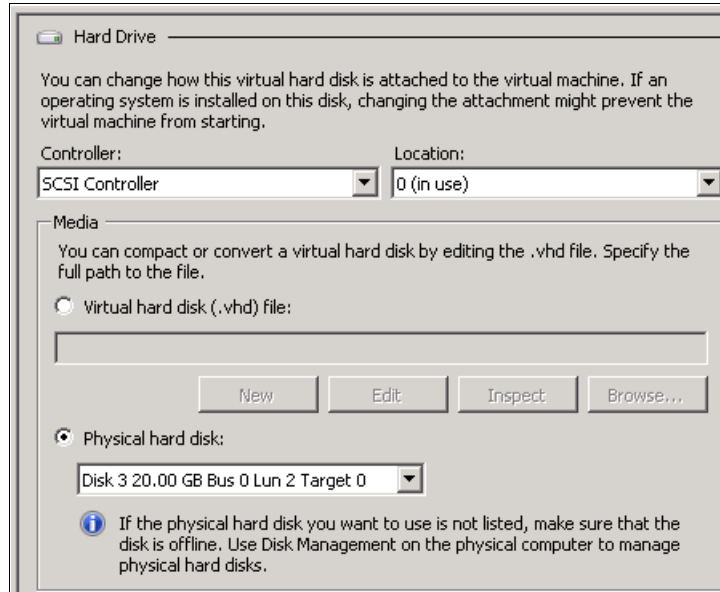


Figure 3-26 Add physical hard disk

**Tip:** The disk to be used as a pass-through disk can contain a file system and data when being attached. This method can be used to move data between different servers, both physical and virtual.

### 3.10.5 Cluster Shared Volume (CSV)

Microsoft Server 2008 R2 Hyper-V provides a functionality called Cluster Shared Volume, which allows concurrent access of multiple Hyper-V hosts to the same NTFS volume.

To use CSV, the following prerequisites need to be fulfilled:

- ▶ All hosts involved are configured in a failover cluster.
- ▶ All hosts involved have access to the LUN that the shared NTFS resides on.
- ▶ The same drive for the volume letter is used on all hosts.

Access to the CSV is controlled by a coordinator node that is chosen automatically. Every node can directly write to the volume by its iSCSI, SAS, or Fibre Channel connection, but file system metadata is routed over a private network to the coordinator node that makes the required file system metadata updates. If the coordinator node fails, the role is moved to any of the remaining nodes.

For more information about CSV, see this website:

<http://technet.microsoft.com/en-us/library/dd630633%28WS.10%29.aspx>

### 3.10.6 Best practices

Device management of Hyper-V is done by the Windows Server 2008. Therefore all steps required to attach DS8000 LUNs to a Hyper-V server are identical to those required for Windows Server 2008. It includes installing the correct HBA drivers and a multipathing driver such as SDDDSM.

To avoid virtual machines impacting the performance of the overall Hyper-V installation, it is advisable to store VHD files on dedicated NTFS partitions on separate LUNs.

VHD files can be migrated between partitions using the regular Windows copy and move functions. The VHD file, however, must not be in use during this operation.





## Virtual I/O Server considerations

This chapter provides an overview of attaching IBM Virtual I/O Server (VIOS) on IBM Power Systems™ to DS8000 storage systems.

The following topics are covered:

- ▶ Working with IBM Virtual I/O Server
- ▶ Using VSCSI with IBM VIOS and DS8000
- ▶ Using NPIV with IBM VIOS and DS8000

For more information about IBM PowerVM® virtualization and the management of virtual storage devices on IBM Power Systems with VIOS, see the following documentation:

- ▶ *IBM PowerVM Getting Started Guide*, REDP-4815:  
<http://www.redbooks.ibm.com/abstracts/redp4815.html?Open>
- ▶ *IBM PowerVM Live Partition Mobility*, SG24-7460:  
<http://www.redbooks.ibm.com/abstracts/sg247460.html?Open>
- ▶ *IBM PowerVM Virtualization Managing and Monitoring*, SG24-7590:  
<http://www.redbooks.ibm.com/abstracts/sg247590.html?Open>
- ▶ *PowerVM and SAN Copy Services*, REDP-4610:  
<http://www.redbooks.ibm.com/abstracts/redp4610.html?Open>
- ▶ *PowerVM Migration from Physical to Virtual Storage*, SG24-7825:  
<http://www.redbooks.ibm.com/abstracts/sg247825.html?Open>
- ▶ *PowerVM Virtualization on IBM System p: Introduction and Configuration Fourth Edition*, SG24-7940:  
<http://www.redbooks.ibm.com/abstracts/sg247940.html?Open>

## 4.1 Working with IBM Virtual I/O Server

Virtualization is a key factor of IBM Smarter Planet™ - IBM Dynamic Infrastructure® strategy. On IBM Power Systems, the VIOS enables sharing of physical resources between *logical partitions* (LPARs), including virtual SCSI, virtual Fibre Channel using *node port ID virtualization* (NPIV), and virtual Ethernet, providing a more efficient utilization of physical resources and facilitating server consolidation.

VIOS is part of the IBM PowerVM editions hardware feature on IBM Power Systems. The VIOS technology facilitates the consolidation of network and disk I/O resources and minimizes the number of required physical adapters in the IBM Power Systems server. It is a special-purpose partition that provides virtual I/O resources to its client partitions. The VIOS actually owns the physical resources that are shared with clients. A physical adapter assigned to the VIOS partition can be used by one or more other partitions.

VIOS can provide virtualized storage devices, storage adapters, and network adapters to client partitions running an AIX, IBM i, or Linux operating environment. The VIOS includes the following core I/O virtualization capabilities:

- ▶ Virtual SCSI
- ▶ Virtual Fibre Channel using NPIV
- ▶ Virtual Ethernet bridge using a shared Ethernet adapter (SEA)

The core I/O virtualization capabilities are represented in Figure 4-1.

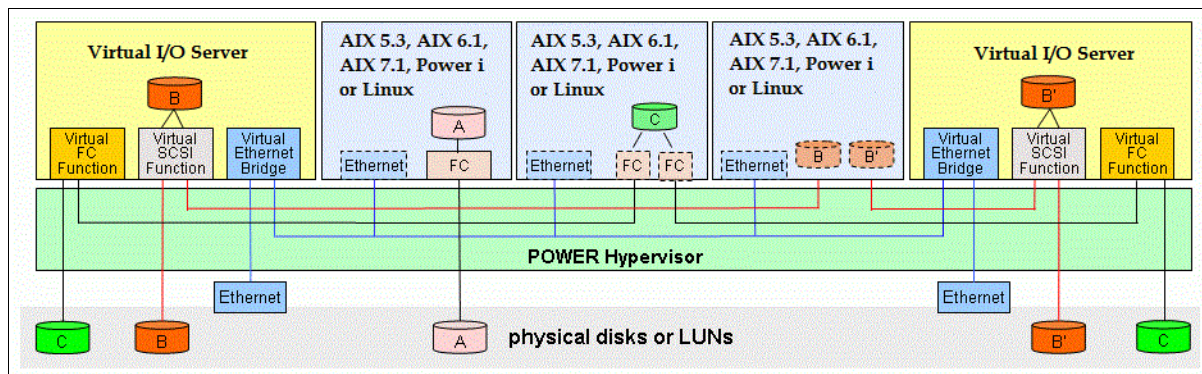


Figure 4-1 Storage and network virtualization with IBM PowerVM and VIOS (VIOS)

Figure 4-1 shows a redundant dual VIOS setup on an IBM Power System providing virtual I/O resources to client partitions:

- ▶ LUN A is directly accessed by the logical partition using a physical Fibre Channel adapter assigned to the partition.
- ▶ LUN B is mapped to both VIOS using the physical Fibre Channel adapter in the VIOS partition. The LUN is used as a virtual SCSI backing device on both VIOS partitions and provided to the client partition as a generic virtual SCSI device through two paths with two configured virtual SCSI server (VSCSI) and client adapter pairs. On the client partition, this VSCSI device is displayed as a generic SCSI disk without any storage subsystem specific properties. On VIOS, the supported storage system specific multipathing driver must be installed to process the I/O to the LUN.



- ▶ LUN C is accessed through virtual Fibre Channel adapters in the client partition with their own specific WWPNs. The LUN is directly mapped to the client partitions' WWPNs and is not accessed by the virtual I/O server. The virtual Fibre Channel adapters are using NPIV and are assigned to a physical Fibre Channel port on the VIOS to physically connect to the SAN. The client partition's Fibre Channel I/O is only passed to the physical Fibre Channel adapters in the VIOS partition. The VIOS does not have access to the client volumes. The LUN is displayed in the client partition with the same storage system specific properties as with physical Fibre Channel attachment, and the supported storage system specific multipathing driver must be installed on the client LPAR.

**NPIV:** For IBM i Power Systems, NPIV requires IBM i LIC 6.1.1 or later with exceptions as detailed in the following links. NPIV capable Fibre Channel adapters and SAN switches are also required.

For a quick reference, click the following link:

<http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS4563>

For further details, see the following website:

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

The virtual Ethernet function, also shown in Figure 4-1 on page 48, is basically provided by the IBM POWER® Hypervisor™ enabling secure communication between logical partitions without the need for a physical I/O adapter or cabling. With shared Ethernet adapters on the VIOS, the internal network traffic from the virtual Ethernet networks can be bridged out to physical Ethernet networks using a physical network adapter in the VIOS partition.

The storage virtualization capabilities by PowerVM and the VIOS are supported by the DS8000 series using DS8000 LUNs as VSCSI backing devices in the VIOS. It is also possible to attach DS8000 LUNs directly to the client LPARS using virtual Fibre Channel adapters though the NPIV.

With regard to continuous availability for virtual I/O, typically two VIOS partitions are deployed on a managed system to provide highly available virtual SCSI, virtual Fibre Channel and shared Ethernet services to client partitions. The main reason to implement two VIOS servers (dual VIOS setup) is to provide continuous access when a VIOS must be taken offline for planned outages, such as software updates.

## 4.2 Using VSCSI with IBM VIOS and DS8000

VIOS enables virtualization of physical storage resources, using generic SCSI block level emulation. Virtualized storage devices are accessed by the client partitions through virtual SCSI devices. All virtual SCSI devices are accessed as standard SCSI compliant LUNs by the client partition.

Virtual SCSI enables client logical partitions (LPARs) to share disk storage and tape, or optical devices, that are assigned to the VIOS logical partition. The functionality for virtual SCSI is provided by the IBM POWER Hypervisor. Virtual SCSI helps enable secure communications between partitions and a VIOS that provides storage backing devices. Disk, tape, or optical devices attached to physical adapters in the VIOS logical partition can be shared by one or more client logical partitions. The VIOS acts as a standard storage subsystem that provides standard SCSI-compliant LUNs. The VIOS is capable of exporting a pool of heterogeneous physical storage as a homogeneous pool of block storage in the form of SCSI disks.

Virtual SCSI is based on a client-server relationship, as shown in Figure 4-2. The VIOS owns the physical resources and the virtual SCSI server adapter, and acts as a server or SCSI target device. The client logical partitions have a SCSI initiator referred to as the virtual SCSI client adapter and access the virtual SCSI targets as standard SCSI LUNs.

The following SCSI peripheral device types are generally supported by the VIOS as backing devices for VSCSI emulation:

- ▶ Disk backed by logical volume
- ▶ Disk backed by physical volume
- ▶ Disk backed by file
- ▶ Optical CD-ROM, DVD-RAM, and DVD-ROM
- ▶ Optical DVD-RAM backed by file
- ▶ Tape devices with virtual tape support that enables serial sharing of selected SAS tape devices

Figure 4-2 shows virtual SCSI emulation on a VIOS server.

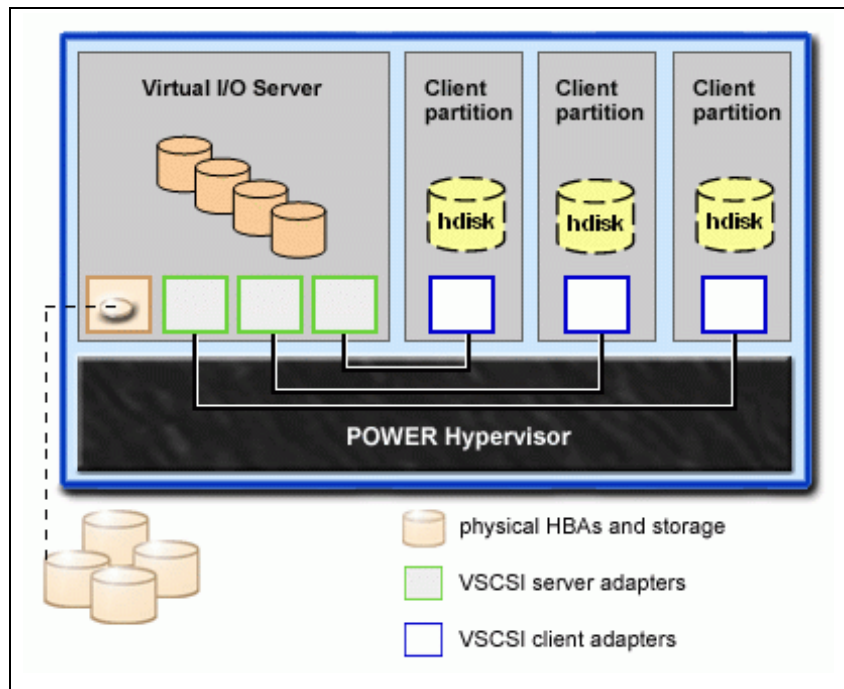


Figure 4-2 VSCSI emulation on VIOS

When using virtual SCSI emulation on the VIOS with DS8000 storage system volumes, all the DS8000 volumes that are to be used as backing devices for the client partitions need to be mapped directly to the VIOS partition using the WWPNs of the physical Fibre Channel adapters in the VIOS partition. You can find the WWPNs of the VIOS Fibre Channel adapters using the `lsdev -dev fcsX -vpd` command for the particular adapter (X=0, 1, 2, and so on), in the VIOS and look for the network address, as shown in Example 4-1. In this example, all the Fibre Channel adapters are shown with the corresponding WWN.

*Example 4-1 List WWPNs of the Fibre Channel adapters in the Virtual I/O Server*

```
$ lsdev -dev fcs* | while read a b; do echo $a $(lsdev -dev $a -vpd | grep Network); done
name
fcs0 Network Address.....10000000C9998F6E
fcs1 Network Address.....10000000C9998F6F
```

To assign volumes to the VIOS on the DS8000 storage system, complete the following steps:

1. Create the DS8000 volume group for the volumes accessed by the VIOS using the **mkvolgrp** command with the **hosttype IBM pSeries®** parameter.
2. Create DS8000 host connections for the Fibre Channel adapters in VIOS using the **mkhostconnect** command with the **hosttype pSeries** parameter and an unused port group, for example, port group 60, to group multiple host connections of the same host or hosts, which share the same volumes together.
3. Create and assign DS8000 volumes to the DS8000 volume group using the **mkfbvol** command. Example 4-2 shows the use of extent pools P8 and P9 as storage pools and volume IDs from the logical subsystem (LSS) 60 / rank group 0 and LSS 61 / rank group 1.
4. Assign the DS8000 volume group to the DS8000 host definition for the VIOS using the **managehostconnect** command.

Example 4-2 shows the output when you create a host attachment and assign DS8000 volumes to VIOS.

*Example 4-2 Creating a host attachment and assigning DS8000 volumes to VIOS*

---

```

dsccli> mkvolgrp -hosttype pSeries vio_1_Residency
CMUC00030I mkvolgrp: Volume group V15 successfully created.

dsccli> mkhostconnect -wwname 10000000C9998F6E -hosttype pSeries -portgrp 60 p770_viol_0
CMUC00012I mkhostconnect: Host connection 0019 successfully created.

dsccli> mkhostconnect -wwname 10000000C9998F6F -hosttype pSeries -portgrp 60 p770_viol_1
CMUC00012I mkhostconnect: Host connection 001A successfully created.

dsccli> mkfbvol -cap 50 -extpool p8 -name p770_viol -volgrp v15 9000-9002
CMUC00025I mkfbvol: FB volume 9000 successfully created.
CMUC00025I mkfbvol: FB volume 9001 successfully created.
CMUC00025I mkfbvol: FB volume 9002 successfully created.

dsccli> managehostconnect -volgrp v15 60
CMUC00016I managehostconnect: Port group number 60 successfully modified.

```

---

To quickly check which volumes are assigned to the VIOS on a DS8000 storage system, use the DS8000 Command Line Interface (DS CLI) with the **lshostconnect** and **lsfbvol** commands, as shown in Example 4-3. The **lshostconnect -login** command even shows the DS8000 I/O ports where the Fibre Channel adapters of the VIOS are logged in.

*Example 4-3 Checking DS8000 volume assignment to VIOS using DSCLI*

---

```

dsccli> lshostconnect -portgrp 60
Name      ID  WWPN                HostType Profile                portgrp volgrpID ESSIOport
=====
p770_viol_0 0019 10000000C9998F6E pSeries  IBM pSeries - AIX          60 V15    all
p770_viol_1 001A 10000000C9998F6F pSeries  IBM pSeries - AIX          60 V15    all

dsccli> lsfbvol -volgrp v15
Name      ID  acctstate  datastate  configstate  deviceMTM  datatype  extpool  cap(2^30B)  cap(10^9B)  cap(blocks)
=====
p770_viol 9000 Online    Normal    Normal      2107-900  FB 512    P8         50.0        -           104857600
p770_viol 9001 Online    Normal    Normal      2107-900  FB 512    P8         50.0        -           104857600
p770_viol 9002 Online    Normal    Normal      2107-900  FB 512    P8         50.0        -           104857600

dsccli> lshostconnect -login
WWNN                WWPN                ESSIOport LoginType Name                ID

```

```
=====
20000000C9998F6E 10000000C9998F6E I0005 SCSI p770_vio1_0 0019
20000000C9998F6E 10000000C9998F6E I0005 SCSI p770_vio1_0 0019
20000000C9998F6F 10000000C9998F6F I0135 SCSI p770_vio1_1 001A
20000000C9998F6F 10000000C9998F6F I0135 SCSI p770_vio1_1 001A
=====
```

---

On the VIOS partition, either the IBM SDD or the Subsystem Device Driver Path Control Module (SDDPCM) file sets must be installed for the DS8000 volumes, as follows:

- ▶ IBM SDD:
  - devices.fcp.disk.ibm.mpio.rte
  - devices.sdd.61.rte (61 stands for AIX6.1 or VIOS 2.x)
- ▶ IBM SDDPCM:
  - devices.fcp.disk.ibm.mpio.rte
  - devices.sddpcm.61.rte (61 stands for AIX6.1 or VIOS 2.x)

The SDD or SDDPCM file sets require separate DS8000 host attachment file sets (ibm.fcp.disk.ibm) as prerequisite. You can have only SDD or SDDPCM file sets installed on a given VIOS, not both. Also note that the SDD or SDDPCM file set to be installed is dependent on the VIOS release (VIOS 1.x based on AIX 5.3 or VIOS 2.x based on AIX 6.1).

For more information about SDD, SDDPCM, and file set downloads, see the support matrix at this website:

<http://www-01.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=DA400&uid=ssg1S7001350>

Also see the readme file of the particular SDD or SDDPCM version for latest updates and prerequisites. New SDD or SDDPCM releases typically require a specific minimum version of the host attachment file set. For supported SDD and SDDPCM releases with IBM System Storage DS8000 series environments and VIOS releases, see the IBM SSIC at this website:

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

The SDD or SDDPCM installation and management on the VIOS is similar to the description given in Chapter 5, “AIX considerations” on page 61.

You can install file sets on the VIOS, by using one of the following commands:

- ▶ The **cfgassist** command within the restricted padmin environment
- ▶ The **oem\_setup \_env** command to gain root privileges, and access the OEM install and setup environment, as on an AIX system

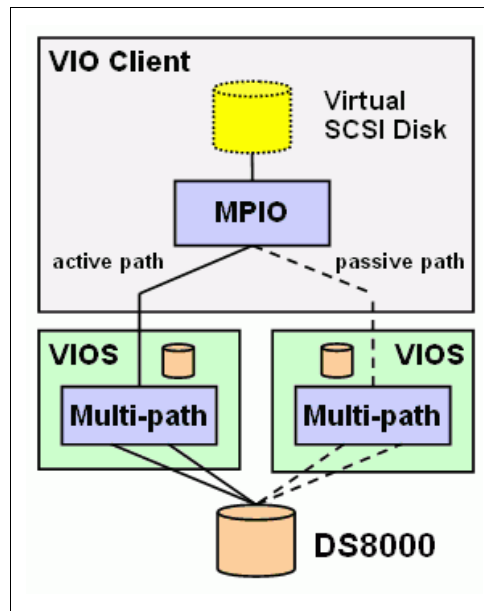
To access the SDD **datapath** command or the SDDPCM **pcmpath** command, access the OEM install and setup environment using **oem\_setup \_env** first. These commands provide detailed and compact information about the devices and their paths, including device statistics, the DS8000 serial number, and the DS8000 volume IDs.

The installed multipathing file sets on the VIOS can be checked, using the `lssw` command as shown in Example 4-4.

*Example 4-4 Checking for the installed SDD or SDDPCM file set on the Virtual I/O Server*

```
$ lssw | grep -iE "sdd|fcp.disk.ibm|FCP Disk"
devices.fcp.disk.ibm.mpio.rte
                                1.0.0.21  C   F   IBM MPIO FCP Disk Device
devices.sddpcm.61.rte           2.6.0.0  C   F   IBM SDD PCM for AIX V61
There is no efix data on this system.
```

For a redundant VIOS setup (Figure 4-3) with two VIOS partitions providing the same DS8000 LUNs to a client partition using multipathing, only SDDPCM is supported on the VIOS.



*Figure 4-3 Dual VIOS setup with DS8000 backing devices and multipath support in client partition*

For more information, see Chapter 4, section 4.14.1 “Supported virtual SCSI configurations,” in the section “Supported scenarios using multiple Virtual I/O Servers”, in the *PowerVM Virtualization on IBM System p: Introduction and Configuration Fourth Edition*, SG24-7940.

IBM i multipath is automatically started on the DS8000 LUNs that are connected through two VIOS partitions in virtual SCSI. There is no need to install any particular driver for multipath in IBM i.

For proper multipath support of virtual SCSI devices in the client partition with a dual VIOS setup, the DS8000 LUN must be presented as a physical device `hdiskX` from the Virtual I/O Servers to the client partition. It is not possible to provide a single DS8000 LUN and then further subdivide it into logical volumes as backing devices at the VIOS level, when intending to use a redundant setup with two VIOS and multipathing in the client partition.

Using SDDPCM, for example, the SDD path control module for AIX MPIO on the VIOS, the DS8000 LUNs is displayed as IBM MPIO FC 2107 devices when properly configured as shown in Example 4-5.

*Example 4-5 VIOS with attached DS8000 volumes using SDDPCM*

---

```
$ lsdev -type disk
name          status      description
hdisk0        Available   IBM MPIO FC 2107
hdisk1        Available   IBM MPIO FC 2107
hdisk2        Available   IBM MPIO FC 2107
```

---

The properties of the DS8000 LUNs using SDDPCM 2.6.0.0 are shown in Example 4-6. You can get the serial number of the DS8000 storage system and the DS8000 volume ID from the output of the `lsdev -dev hdiskX -vpd` command. The Serial Number 75TV1819000 displayed includes the serial number (75TV181) of the storage image on the DS8000 storage system and the DS8000 volume ID (9000).

*Example 4-6 Serial number of DS8000 LUNs in Virtual I/O Server*

---

```
$ lsdev -dev hdisk0 -vpd
hdisk0          U78C0.001.DBJ2743-P2-C4-T1-W500507630A4B429F-L4090400000000000 IBM MPIO FC 2107

Manufacturer.....IBM
Machine Type and Model.....2107900
Serial Number.....75TV1819000
EC Level.....160
Device Specific.(Z0).....10
Device Specific.(Z1).....0135
Device Specific.(Z2).....075
Device Specific.(Z3).....22010
Device Specific.(Z4).....08
Device Specific.(Z5).....00
```

PLATFORM SPECIFIC

```
Name: disk
Node: disk
Device Type: block
```

```
$ lspath -dev hdisk0
status name parent connection

Enabled hdisk0 fscsi0 500507630a4b429f,4090400000000000
Enabled hdisk0 fscsi0 500507630a40429f,4090400000000000
Enabled hdisk0 fscsi1 500507630a4b429f,4090400000000000
Enabled hdisk0 fscsi1 500507630a40429f,4090400000000000
```

---

The `lspath` command provides information about the available paths for a given LUN when using AIX MPIO with SDDPCM. It displays the Fibre Channel adapter port in the server `fscsi0`, the DS8000 I/O port WWPN 500507630a4b429f and the DS8000 volume ID 9000 with third, fourth, sixth and eighth positions in the number 4090400000000000 provided with the output for each path.

By using the DSCLI **lσιοport** command, as shown in Example 4-7, you can identify the I/O ports used on the DS8000 for each path.

*Example 4-7 Using the DSCLI lσιοport command to list the WWPNs of the DS8000 I/O ports*

---

```
$ dscli> lσιοport
Date/Time: 30 de mayo de 2012 10:39:40 AM CEST IBM DSCLI Versio
  IBM.2107-75TV181
ID      WWPN                State Type                topo      portgrp
=====
I0005  500507630A40429F Online Fibre Channel-SW SCSI-FCP 0
I0135  500507630A4B429F Online Fibre Channel-SW SCSI-FCP 0
```

---

These I/O ports must match the ports listed by the **lshostconnect -login** command, as shown in Example 4-3 on page 51.

For a proper assignment of the DS8000 volumes as VSCSI backing devices on the VIOS to the client LPARs running on AIX, IBM i, or Linux on POWER, correctly identify the association of DS8000 volumes with VIOS hard disks. Each VIOS hdisk can be associated with the DS8000 volume ID by using the command shown in Example 4-8.

*Example 4-8 Associate VIOS hdisks to DS8000 volumes*

---

```
$ lsdev -type disk|grep 2107 | while read a b; do echo "$a $(lsdev -dev $a
-vpd|grep Serial)"; done
hdisk0      Serial Number.....75TV1819000
hdisk1      Serial Number.....75TV1819004
hdisk2      Serial Number.....75TV1819005
```

---

In redundant VIOS environments such as dual VIOS environments, it is important to uniquely identify the hdisks by their DS8000 serial number to assign the correct volumes on both VIOS to the client partitions, for proper multipathing. Another alternative to uniquely identify the hard disks on the VIOS, is to use the **chkdev** command, which displays the UDID of each hdisk, as shown in Example 4-9.

*Example 4-9 Identifying DS8000 LUNs on VIOS using UDID*

---

```
$ chkdev | grep -E "NAME|ID"
NAME:          hdisk0
IDENTIFIER:    200B75TV181900007210790003IBMfcp
NAME:          hdisk1
IDENTIFIER:    200B75TV181900407210790003IBMfcp
NAME:          hdisk2
IDENTIFIER:    200B75TV181900507210790003IBMfcp
```

---

In a dual VIOS environment with two virtual I/O servers providing the same DS8000 LUN as backing device to the same client LPAR, the UDID of the DS8000 disk is used by AIX MPIO to properly identify all the paths for a given virtual SCSI disk.

The UDID of the Virtual SCSI disk can also be used on the client partition for identifying the volume ID of the DS8000 volume used as backing device, as shown in 5.2, “Attaching virtual SCSI” on page 65.

The UDID shown in Example 4-9 contains the DS8000 serial number, volume ID and device type. For example, the unit device identifier (UDID) **200B75TV181900007210790003IBMfcp** contains the DS8000 storage image serial number 75TV181, the DS8000 volume number 9000 and the device type 2107-900.

Because virtual SCSI connections operate at memory speed, generally, no added performance is gained by adding multiple adapters between a VIOS and the client partition. However, it is a best practice to use separate virtual SCSI adapter server and client pairs for boot or AIX rootvg disks and data disks, in addition to backing devices from separate storage systems.

In general, for AIX virtual I/O client partitions, each adapter pair can handle up to 85 virtual devices with the default queue depth of three. For IBM i clients, up to 16 virtual disks and 16 optical devices are supported.

**Tip:** The queue depth, on LUNs connected by virtual SCSI to IBM i, is 32.

In situations where virtual devices per partition are expected to exceed that number, or where the queue depth on certain devices might be increased to more than the default, additional adapter slots for the VIOS and the virtual I/O client partition need to be created.

The VSCSI adapters have a fixed queue depth. There are 512 command elements, of which two are used by the adapter, three are reserved for each VSCSI LUN for error recovery, and the rest are used for I/O requests. Thus, with the default queue depth of 3 for VSCSI LUNs, that allows for up to 85 LUNs to use an adapter:

$$(512-2)/(3+3) = 85 \text{ LUNs}$$

If a higher queue depths for the virtual SCSI disk devices in the client partition is set, then the number of LUNs per adapter is reduced. For example, with a default queue depth of 20 for a DS8000 LUN as backing device using SDDPCM (see Example 4-10) you might need to increase the queue depth of the virtual SCSI disk device in the client partition also to 20 for best performance. For an AIX client partition, it allows the following DS8000 LUNs per adapter:

$$(512-2)/(20+3) = 22$$

The default attributes of a DS8000 device on the VIOS with SDDPCM version 2.6.0.0 are shown in Example 4-10.

*Example 4-10 Attributes of DS8000 volumes in the VIOS partition using SDDPCM*

```
$ lsdev -dev hdisk0 -attr
```

attribute	value	description	user_settable
PCM	PCM/friend/sddpcm	PCM	True
PR_key_value	none	Reserve Key	True
algorithm	load_balance	Algorithm	True
clr_q	no	Device CLEARS its Queue on error	True
dist_err_pcmt	0	Distributed Error Percentage	True
dist_tw_width	50	Distributed Error Sample Time	True
hcheck_interval	60	Health Check Interval	True
hcheck_mode	nonactive	Health Check Mode	True
location		Location Label	True
lun_id	0x4090400000000000	Logical Unit Number ID	False
lun_reset_spt	yes	Support SCSI LUN reset	True
max_transfer	0x100000	Maximum TRANSFER Size	True
node_name	0x500507630affc29f	FC Node Name	False
pvid	00f60dbc813b7cbd0000000000000000	Physical volume identifier	False
q_err	yes	Use QERR bit	True
q_type	simple	Queuing TYPE	True
qfull_dly	2	delay in seconds for SCSI TASK SET FULL	True
queue_depth	20	Queue DEPTH	True
reserve_policy	no_reserve	Reserve Policy	True



retry_timeout	120	Retry Timeout	True
rw_timeout	60	READ/WRITE time out value	True
scbsy_dly	20	delay in seconds for SCSI BUSY	True
scsi_id	0x201400	SCSI ID	False
start_timeout	180	START unit time out value	True
unique_id	200B75TV181900007210790003IBMfcp	Device Unique Identification	False
ww_name	0x500507630a4b429f	FC World Wide Name	False

When using a redundant dual VIOS setup with DS8000 devices as virtual SCSI backing devices and native AIX MPIO in the client partition, see the setup advice for virtual SCSI redundancy in *PowerVM Virtualization on IBM System p: Introduction and Configuration Fourth Edition*, SG24-7940. Figure 5-2 on page 67 shows an overview of preferred MPIO settings in the VIOS partition and AIX client partition.

In a dual VIOS setup where the same DS8000 volume is assigned to both VIOS partitions and mapped as backing device to the same client partition, you must set the **reserve\_policy** attribute on the device to `no_reserve`, which is already the default for DS8000 devices with SDDPCM installed. Also, the preferred AIX MPIO device attributes **algorithm** (`load_balance`), **hcheck\_mode** (`nonactive`) and **hcheck\_interval** (`60 seconds`) are already set by default for DS8000 devices with current SDDPCM releases.

For dual VIOS setups, change the default values of the attributes **fc\_err\_recov** and **dyntrk** of the Fibre Channel adapter's child device, `fscsiX`, to the settings shown in Example 4-11 on both virtual I/O servers.

*Example 4-11 Changing the fscsi attributes in a dual VIOS setup*

```
$ chdev -dev fscsi0 -attr fc_err_recov=fast_fail dyntrk=yes -perm
fscsi0 changed
```

Changing the **fc\_err\_recov** attribute to `fast_fail` will fail any I/Os immediately, if the adapter detects a link event, such as a lost link between a storage device and a switch. The **fast\_fail** setting is only advisable for dual VIOS configurations. Setting the **dyntrk** attribute to `yes` enables the VIOS to allow cabling changes in the SAN. Both virtual I/O servers need to be rebooted for these changed attributes to take effect.

**Applying changes:** Changes applied with the `chdev` command and the `-perm` option require a restart to become effective.

## 4.3 Using NPIV with IBM VIOS and DS8000

VIOS 2.1 release with fix pack FP20.1 (or later) introduced virtual Fibre Channel capabilities, NPIV, to IBM Power Systems, considerably extending the virtualization capabilities and enabling innovative solutions with IBM Power Systems and SAN storage.

NPIV is a standardized method for virtualizing a physical Fibre Channel port. The PowerVM implementation of NPIV enables POWER LPAR to have virtual Fibre Channel adapters, each with a dedicated WWPN.

With NPIV and unique WWPNs, a system partition has its own identity in the SAN with direct access to both disk and tape SAN storage devices, as shown in Figure 4-4. It is supported, for example, with the IBM 8 Gbps Fibre Channel HBA #5735.

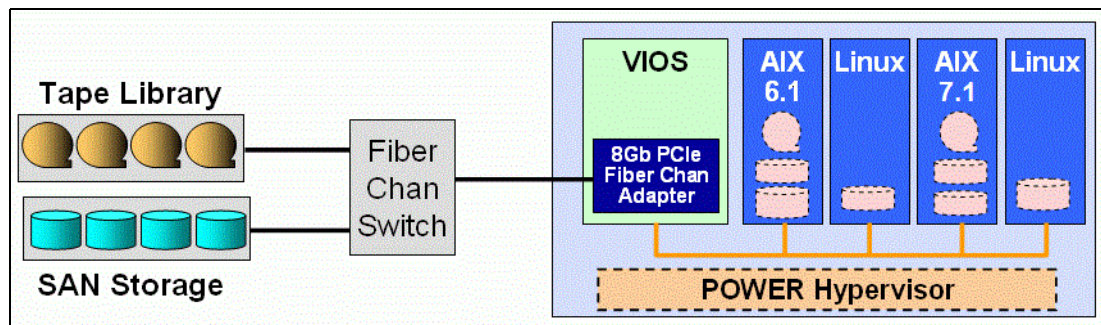


Figure 4-4 Virtual Fibre Channel support using NPIV on IBM Virtual I/O Server

IBM System Storage DS8000 supports VIOS with client partitions using virtual Fibre Channel adapters and NPIV. This establishes techniques for SAN storage management, such as SAN zoning, LUN mapping and masking, combined with NPIV, to be used when provisioning dedicated storage to specific LPARs and applications, without the need to additionally map those volumes as backing devices in the VIOS partition. Furthermore, solutions built around advanced storage system features, such as copy services and backup and restore, work immediately in an application LPAR with virtual Fibre Channel and NPIV.

With virtual Fibre Channel, physical-to-virtual device compatibility issues do not arise, and even SCSI-2 reserve or release, and SCSI-3 persistent reserve methodologies in clustered or distributed environments are available. The specific multipathing software, such as SDDPCM providing a load-balancing path selection algorithm, for the mapped SAN devices must be installed directly on the client LPAR, not on the VIOS. The VIOS partition only acts as a pass-through module. The LUNs mapped to WWPNS using NPIV are not actually seen by the VIOS.

With NPIV, you can configure the managed system so that multiple logical partitions can access independent physical storage through the same physical Fibre Channel adapter, each with their own unique WWPNS. The VIOS partition owns the physical Fibre Channel adapters which must support NPIV, such as the IBM 8 Gbps PCI Express Dual Port Fibre Channel Adapter. The VIOS partition acts as pass-through module to bridge Fibre Channel traffic from the configured virtual Fibre Channel host adapters vfchost to the physical adapter ports.

The client partition owns a virtual Fibre Channel client adapter and directly connects to a virtual Fibre Channel host adapter on the VIO partition, as shown in Figure 4-5.

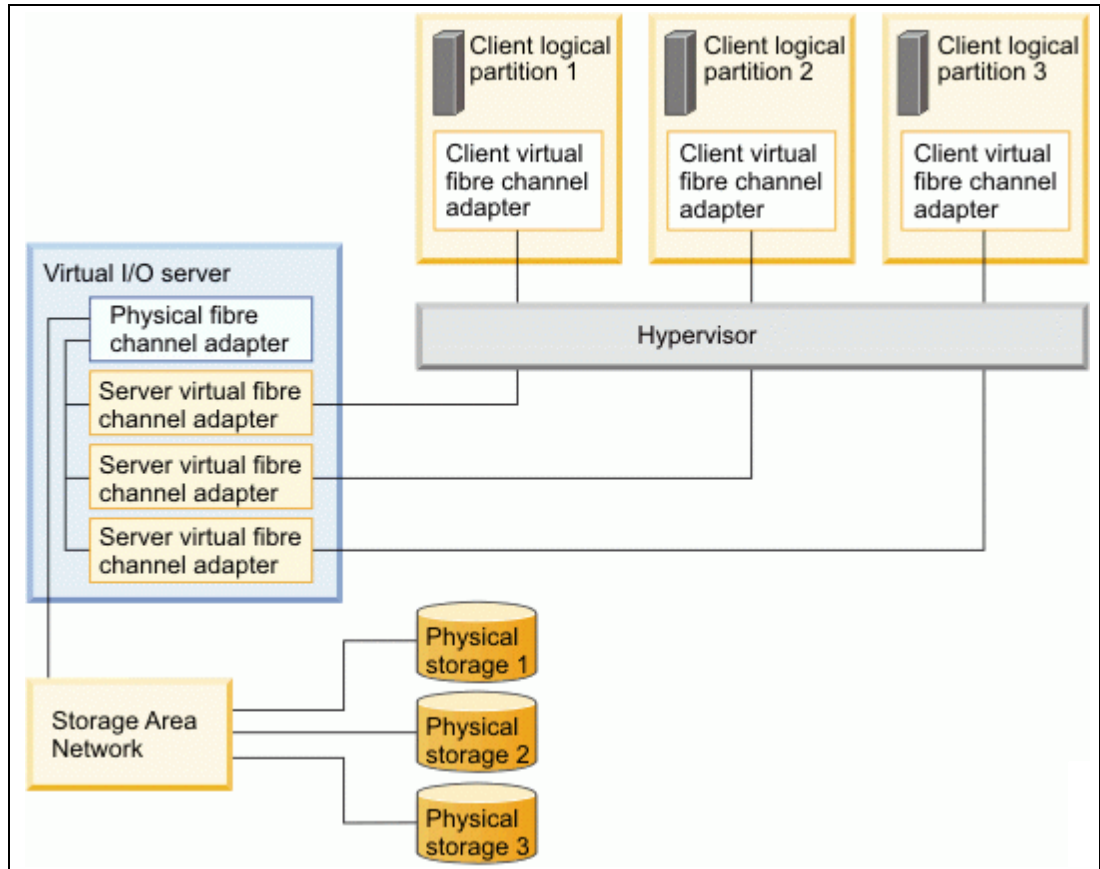


Figure 4-5 Virtual Fibre Channel on a single VIOS managed system

Each virtual Fibre Channel server adapter on the VIOS partition connects to one virtual Fibre Channel client adapter on a virtual I/O client partition. Using their unique WWPNs and the virtual Fibre Channel connections to the physical Fibre Channel adapter, the operating systems that runs in the client logical partitions discover, initialize, and manage their physical storage located in the SAN.

The WWPN for the client LPAR is automatically generated by the system when the virtual Fibre Channel client adapter is created for an active partition. The Hardware Management Console (HMC) generates WWPNs based on the range of names available for use with the prefix from the vital product data within the managed system. When a virtual Fibre Channel adapter is removed from a client logical partition, the hypervisor deletes the WWPNs that are assigned to the virtual Fibre Channel adapter on the client logical partition. Because the HMC does not reuse deleted WWPNs when generating future WWPNs for virtual Fibre Channel adapters, careful attention is required to manage partition profiles with virtual Fibre Channel adapters to maintain WWPNs and not lose the work done for the device mapping on the storage system and SAN zoning.

Continuous availability is achieved with a redundant VIO server setup, such as the following example:

- ▶ Using a minimum of four virtual Fibre Channel adapters in the client partition
- ▶ Connecting to both VIO server partitions
- ▶ Mapping to separate physical Fibre Channel adapters and two SAN fabrics

A redundant VIO server setup (Figure 4-6) provides a reasonable redundancy against a VIOS failure. A VIOS failure can be due to a planned maintenance and a simultaneous failure of one SAN fabric.

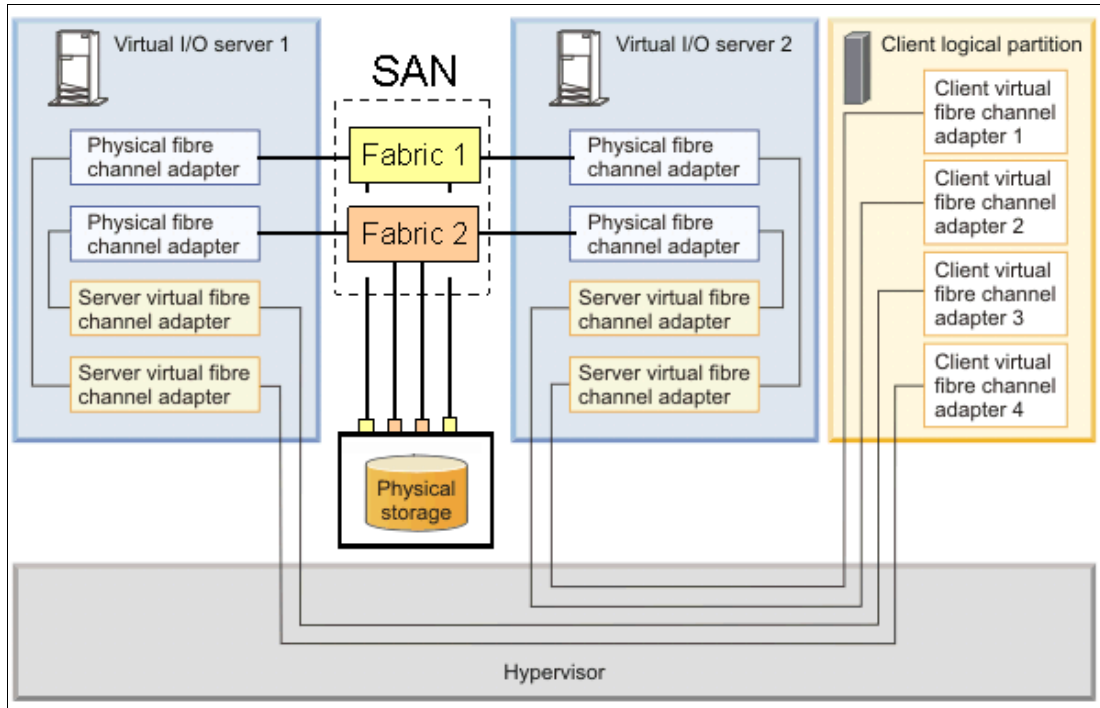


Figure 4-6 Virtual Fibre Channel on a dual VIOS managed system

For more information about virtual Fibre Channel and NPIV, see *IBM PowerVM Virtualization Managing and Monitoring*, SG24-7590.

**LUNs:** IBM i supports DS8000 LUNs attached to VIOS using NPIV. IBM i multipath is automatically started on the DS8000 LUNs that are connected through two VIOS partitions in NPIV. Up to 64 LUNs can be assigned to a virtual port in IBM i in NPIV, the queue depth is 6 operations per LUN.

On IBM Power Systems with PowerVM, a pair of WWPNs is generated for a virtual Fibre Channel client adapter with subsequent WWPNs, as shown in Figure 5-1 on page 64. Only one WWPN is active on a managed system at a time. The second WWPN is used for the PowerVM Live Partition Mobility feature and becomes active on the secondary managed system when a partition with virtual Fibre Channel adapters is migrated to this managed system. In order to maintain access to the mapped volumes on DS8000 storage systems with PowerVM Live Partition Mobility, create additional host connections and SAN zoning also for these secondary WWPNs.

For more information about IBM PowerVM Live Partition Mobility, see *IBM PowerVM Live Partition Mobility*, SG24-7460.



## AIX considerations

This chapter provides information about the IBM AIX operating system. It is not intended to repeat general information that is in other publications. For general information, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917 at the following website:

<http://www-304.ibm.com/support/docview.wss?uid=ssg1S7001161>

When attaching an AIX system to the DS8000 series, you can distinguish between direct attachment using physical or virtual Fibre Channel adapters in the AIX partition, or virtual attachment using virtual SCSI disks, which are backed by DS8000 volumes in a VIOS partition. Both approaches are described in this chapter.

The following topics are covered:

- ▶ Attaching native Fibre Channel
- ▶ Attaching virtual SCSI
- ▶ Important additional considerations
- ▶ Multipathing with AIX
- ▶ Configuring LVM
- ▶ Using AIX access methods for I/O
- ▶ Expanding dynamic volume with AIX
- ▶ SAN boot support

## 5.1 Attaching native Fibre Channel

To allocate DS8000 disks to AIX, the WWPN of each HBA on the host server must be registered in the DS8000. The WWPNs of physical or virtual Fibre Channel adapters can be seen in the output of the `lscfg` command as shown in Example 5-1.

*Example 5-1 List WWPNs of the Fibre Channel adapters in an AIX partition*

---

```
# lsdev -SA -l "fcs*"
[p7-770-02v5:root:/:] lsdev -SA -l "fcs*"
fcs0 Available 00-00 8Gb PCI Express Dual Port FC Adapter (df1000f114108a03)
fcs1 Available 00-01 8Gb PCI Express Dual Port FC Adapter (df1000f114108a03)

# lsdev -SA -l "fcs*" | while read a b; do echo $a $(lscfg -v1 $a | grep Net); done
fcs0 Network Address.....10000000C9808618
fcs1 Network Address.....10000000C9808619
```

---

Example 5-1 shows two physical `fcs0`, `fcs1` Fibre Channel adapters in the AIX partition.

Alternatively, if you have SDD or SDDPCM already installed on your machine (see 5.4, “Multipathing with AIX” on page 72 for details), you can use either the `datapath` or the `pcmpath` command to get the list of WWPNs. Both commands use the same syntax. Example 5-2 shows a list of WWPNs after using the `pcmpath` command.

*Example 5-2 SDD WWPN query*

---

```
# pcmpath query wwpn
Adapter Name      PortWWN
fscsi0            10000000C9808618
fscsi1            10000000C9808619
```

---

### 5.1.1 Assigning volumes

Complete the following steps on the DS8000 storage system to directly assign volumes to an AIX partition using Fibre Channel adapters:

1. Create a DS8000 volume group for the volumes accessed by the AIX server using the `mkvolgrp` command with `hosttype pSeries`.
2. Create DS8000 host connections for the Fibre Channel adapters in the AIX server using the `mkhostconnect` command with `hosttype pSeries` and an unused port group (for example, `port group 45`) to group multiple host connections of the same host or hosts sharing the same volumes. It simplifies connection management later. Also, you can assign a volume group to a host connection definition at this time, or do it later using the `managehostconnect` command.
3. Create and assign DS8000 volumes to DS8000 volume group using the `mkfbvol` command or later using `chvolgrp` command. Example 5-3 shows these examples: extent pools P8 and P9 as storage pools and volume IDs from LSS of 60 per rank group 0, and LSS 61 per rank group 1.
4. Assign DS8000 volume group to DS8000 host definition for the AIX server using the `managehostconnect` command if not already done with `mkhostconnect` command.

Example 5-3 illustrates steps 2 through 4 on page 62.

*Example 5-3 Creating a host attachment and assigning DS8000 volumes to an AIX server*

```

dsccli> mkvolgrp -hosttype pseries LPAR_1_Residency
CMUC00030I mkvolgrp: Volume group V16 successfully created.
dsccli> mkhostconnect -wwname 1000000C9808618 -hosttype pseries -portgrp 45
p770_02_lpar1_fc0
CMUC00012I mkhostconnect: Host connection 001B successfully created.
dsccli> mkhostconnect -wwname 1000000C9808619 -hosttype pseries -portgrp 45
p770_02_lpar1_fc1
CMUC00012I mkhostconnect: Host connection 001C successfully created.
dsccli> mkfbvol -cap 50 -extpool p8 -name p770_vol -volgrp v45 6200-6201
CMUC00025I mkfbvol: FB volume 6200 successfully created.
CMUC00025I mkfbvol: FB volume 6201 successfully created.
dsccli> managehostconnect -volgrp v45 45
CMUC00016I managehostconnect: Port group number 45 successfully modified.

```

To quickly check which volumes are assigned to a given AIX server on a DS8000 storage system, use the DS8000 Command Line Interface (DSCLI) commands **lshostconnect** and **lsfbvol**, as shown in Example 5-4. The **lshostconnect -login** command on DSCLI also shows the DS8000 I/O ports where the Fibre Channel adapters of the AIX server are logged in.

*Example 5-4 Checking DS8000 volume assignment to an AIX server using DSCLI*

```

dsccli> lshostconnect -portgrp 45
Name                ID   WWPN                HostType Profile                portgrp volgr pID ESSIOport
=====
p770_02_lpar1_fc0 005C 1000000C9808618 pSeries  IBM pSeries - AIX          45 V45 all
p770_02_lpar1_fc1 005D 1000000C9808619 pSeries  IBM pSeries - AIX          45 V45 all

dsccli> lsfbvol -volgrp v45
Name                ID   accstate  datastate  configstate  deviceMTM  datatype  extpool  cap (2^30B)  cap (10^9B)  cap
(blocks)
=====
p770_vol            6200 Online    Normal     Normal      2107-900  FB 512    P8 50.0    -            104857600
p770_vol            6201 Online    Normal     Normal      2107-900  FB 512    P8 50.0    -            104857600
p770_vol            6202 Online    Normal     Normal      2107-900  FB 512    P8 50.0    -            104857600
boot_LPAR1_9002    9002 Online    Normal     Normal      2107-900  FB 512    P8 50.0    -            104857600

dsccli> lshostconnect -login
WWNN                WWPN                ESSIOport LoginType Name                ID
=====
20000000C9808618 10000000C9808618 I0005     SCSI      p770_02_lpar1_fc0 005C
20000000C9808619 10000000C9808619 I0005     SCSI      p770_02_lpar1_fc1 005D

```

## 5.1.2 Using node port ID virtualization (NPIV)

When using virtual Fibre Channel adapters, two virtual WWPNs are assigned to a given virtual Fibre Channel adapter by the IBM Power System. Figure 5-1 shows the properties for a virtual Fibre Channel adapter on a Power HMC GUI.

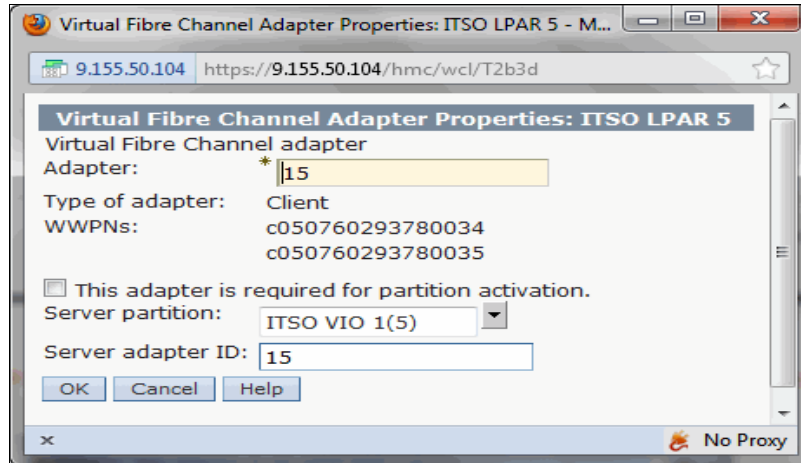


Figure 5-1 WWPNs of a virtual Fibre Channel adapter

Multipath drivers such as SDD or SDDPCM provide commands such as the **pcmpath** command for SDDPCM and the **datapath** command for SDD. These commands can be used to manage and list DS8000 devices and their paths on AIX and to associate them with their DS8000 serial numbers.



## 5.2 Attaching virtual SCSI

When using virtual SCSI emulation on the VIOS with DS8000 storage system volumes, all the DS8000 volumes that are used as backing devices for the client partitions need to be mapped to the VIOS partition using the WWPNs of the physical Fibre Channel adapters in the VIOS partition, as described in “Using VSCSI with IBM VIOS and DS8000” on page 49.

The virtual SCSI disks that are backed by DS8000 volumes on the VIOS appear as generic virtual SCSI disk devices on the AIX client partition without any storage system specific attributes, as shown in Example 5-5.

*Example 5-5 DS8000 backed virtual SCSI disks on an AIX client partition*

---

```
# lsdev -Cc disk
hdisk0 Available Virtual SCSI Disk Drive

# lscfg -v1 hdisk2
hdisk0          U9117.MMB.100DBCP-V9-C31-T1-L8100000000000000 Virtual SCSI Disk Drive

# lsattr -El hdisk2
PCM             PCM/friend/vscsi          Path Control Module      False
algorithm       fail_over                  Algorithm                 True
hcheck_cmd      test_unit_rdy             Health Check Command     True
hcheck_interval 20                       Health Check Interval    True
hcheck_mode     nonactive                  Health Check Mode        True
max_transfer    0x40000                   Maximum TRANSFER Size    True
pvid            00f60dbc823d03050000000000000000 Physical volume identifier False
queue_depth     3                          Queue DEPTH              True
reserve_policy  no_reserve                 Reserve Policy           True
```

---

However, the UDID of the virtual SCSI disk devices can be used to properly identify which DS8000 LUN is used on the VIOS as backing device. The UDID is stored in the AIX Object Data Manager (ODM) and can be listed using the `odmget -q attribute=unique_id CuAt` command, as shown in Example 5-6.

*Example 5-6 Listing the UDID for DS8000 backed Virtual SCSI Disk devices in AIX client partition*

---

```
# odmget -q attribute=unique_id CuAt
CuAt:
    name = "hdisk0"
    attribute = "unique_id"
    value = "3520200B75TV181900407210790003IBMfcp05VDASD03AIXvscsi"
    type = "R"
    generic = ""
    rep = "n"
    nls_index = 0
```

---

The UDID in Example 5-6 contains the original UDID of the DS8000 LUN, as shown on the VIOS, using the `chkdev` command in Example 4-9 on page 55. In Example 5-6, the UDID value of `3520200B75TV181900407210790003IBMfcp05VDASD03AIXvscsi` shown for `hdisk0` contains the serial number for the DS8000 storage image 75TV181, the DS8000 volume ID 9004 and the device type 2107-900. Without viewing the virtual device mapping on the VIOS, you can see which DS8000 LUN is associated with each virtual SCSI disk device on an AIX client partition.

The default AIX MPIO path control module (PCM) is used on the AIX client partition, even though SDD or SDDPCM is used as multipath driver on the VIOS partition with a default multipath algorithm of load balancing for the DS8000 LUN. The only available multipath algorithm here is failover as shown in Example 5-7. So, for any given Virtual SCSI Disk device only one path to one VIOS is used for I/O at a given time even when both paths are enabled. AIX MPIO will switch to the secondary path only in the case of a failure of the primary path, for example, during a shutdown of the connected VIOS.

*Example 5-7 Multipath algorithms for Virtual SCSI Disk devices with default MPIO PCM in AIX*

---

```
# lsattr -Rl hdisk0 -a algorithm
fail_over
```

---

In a redundant dual VIOS environment, the virtual SCSI disk device is provided by two virtual I/O servers to the client partition. The I/O can be balanced manually between both VIOS partitions by setting the AIX path priority accordingly on the AIX client partition for each virtual SCSI device, using the **chpath** command as shown in Example 5-8.

The default path priority is 1 which is the highest priority. By default every path has priority 1 and the first one will be used for I/O in failover mode only. By changing this value for selected paths to 2 or even a higher value, the path priority is lowered and only the highest priority path is going to be used for I/O in failover mode, as long as it is available. Example 5-8 shows the set the priority of the second path through the adapter vscsi1, connected to VIOS#2, of hdisk2 to 2, so that the first path through the adapter vscsi0, connected to VIOS#1, with a priority of 1 remains the preferred path for I/O. By applying this pattern for each of the available virtual SCSI disks devices in the partition in a load-balanced rotating method, I/O can manually be balanced across both VIOS partitions.

*Example 5-8 Using the AIX chpath command to balance I/O across paths in a dual VIOS setup*

---

```
# lspath -l hdisk0
Enabled hdisk0 vscsi0
Enabled hdisk0 vscsi1

# lspath -AHE -l hdisk0 -p vscsi0
attribute value description user_settable
priority 1 Priority True

# lspath -AHE -l hdisk0 -p vscsi1
attribute value description user_settable
priority 1 Priority True

# chpath -l hdisk0 -p vscsi1 -a priority=2
path Changed

# lspath -AHE -l hdisk0 -p vscsi1
attribute value description user_settable
priority 2 Priority True
```

---

For AIX MPIO multipath support of virtual SCSI devices in the AIX client partition, the DS8000 LUN must be presented as a physical device (hdiskX) from the VIOS to the client partition. It is not possible to provide a single DS8000 LUN and then further subdivide it into logical volumes as backing devices at the VIOS level when intending to use two virtual I/O servers and client AIX MPIO multipathing.

When using a redundant dual VIOS setup with DS8000 devices as virtual SCSI backing devices and native AIX MPIO in the client partition, follow the setup advice for the MPIO in the client partition in *IBM PowerVM Virtualization Introduction and Configuration*, SG24-7940.

Figure 5-2 provides an example of MPIO in the client partition.

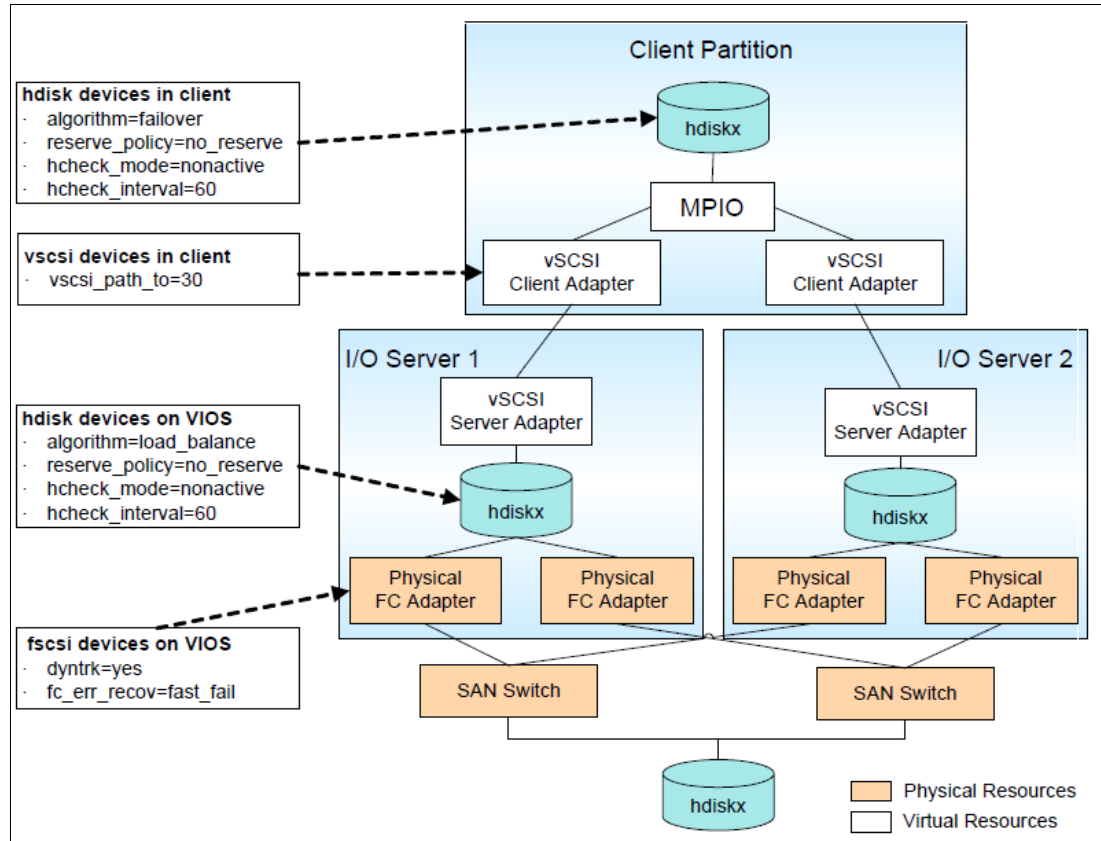


Figure 5-2 Preferred MPIO attributes for DS8000 devices in a dual VIOS setup

Using the default attributes for MPIO multipathing of the virtual SCSI disk devices in the AIX client partition, as shown in Example 5-5 on page 65, set the `hcheck_interval` from 0 (disabled) to the preferred value of 60 seconds on the client partition in a dual VIOS setup, as shown in Example 5-9, so that the path status is updated automatically. The `hcheck_interval` attribute defines how often the health check is performed. The attribute `hcheck_mode` is set to the preferred value of `nonactive` by default, which means that health check commands are sent down all paths that have no active I/O, including paths with a state of failed. In that case, after one VIOS fails and then comes back, manually set the path as available. Changing `hcheck_interval` is advisable for all virtual SCSI hdisk devices, as shown in Example 5-9.

*Example 5-9 Changing the `hcheck_interval` to 60 seconds for a virtual SCSI disk*

```
# chdev -l hdisk2 -a hcheck_interval=60 -P
hdisk2 changed
```

The `hcheck_interval` must never be set lower than the read or write time-out value of the underlying physical disk on the VIOS, which is 60 seconds for a DS8000 volume being used as backing device with SDDPCM (see Example 4-10 on page 56). Otherwise, if the Fibre Channel adapter has an error, there will be new health check requests sent before the running ones time out and it can lead to an adapter failure condition.

**The `hcheck_interval` attribute:** Set the `hcheck_interval` attribute value to 60.

Also, as shown in Example 5-5 on page 65, the default queue depth of a virtual SCSI disk device is set to 3. When a DS8000 volume is used as backing device on the VIOS with a default queue depth of 20, as shown in Example 4-10 on page 56, with SDDPCM, consider increasing the queue depth of the virtual SCSI disk device to the same value shown in Example 5-10 for best performance.

With 512 command elements on the virtual SCSI client adapter (`vscsiX`) of which 2 are used by the adapter, 3 are reserved for each VSCSI LUN for error recovery, and the rest are used for I/O requests, you cannot configure more than 22 DS8000 LUNs as backing devices for a given VSCSI adapter pair, when using a queue depth of 20 for the virtual SCSI disk device.

*Example 5-10 Changing the queue depth to 20 for a Virtual SCSI Disk backed by a DS8000 volume*

---

```
# chdev -l hdisk2 -a queue_depth=20 -P
hdisk2 changed
```

---

On the `vscsiX`, configure the dual VIOS setups for the virtual SCSI adapter path time-out feature (`vscsi_path_to`). It enables the client adapter to check the health of VIOS servicing a particular adapter and detect if a VIOS is not responding to I/O requests. In such a case, the client will failover to an alternate path, if the virtual SCSI adapter path time-out is configured. This feature is configured using the `chdev` command, as shown in Example 5-11:

*Example 5-11 Changing the virtual SCSI client adapter path time-out value in dual VIOS setups*

---

```
# chdev -l vscsi0 -a vscsi_path_to=30 -P
vscsi0 changed
```

---

**Tip:** Changes applied with the `chdev` command and the `-P` option require a reboot to become effective.

## 5.3 Important additional considerations

This section describes some additional considerations to be taken when attaching a host running an AIX system to a DS8000 storage system.

### 5.3.1 Queue depth tuning

Figure 5-3 represents the order in which software comes into play over time as the I/Os traverse the stack.

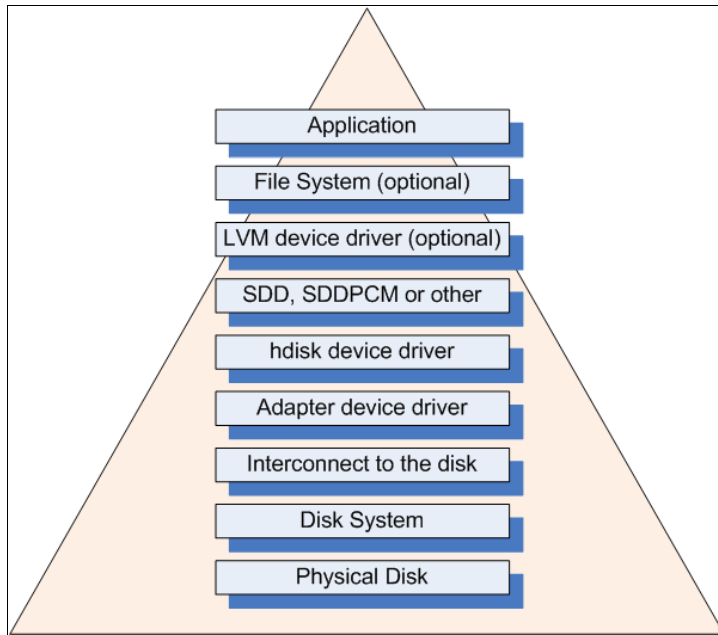


Figure 5-3 I/O stack from the application to the disk

The reason to submit more than one I/O to a disk is performance. Only submitting a single I/O at a time can give very good I/O service time, but very poor throughput. The IOPS for a disk is limited by  $\text{queue\_depth} / \text{average I/O service time}$ .

The queue depth is the number of I/O operations that can run in parallel on a device. The service time is the time needed to physically process the I/O. Assuming a `queue_depth` of 3, and an average I/O service time of 10 ms, this yields a maximum throughput of 300 IOPS for the `hdisk`. And, for many applications, it might not be enough throughput.

As I/Os traverse the I/O stack, AIX needs to keep track of them at each layer. So I/Os are essentially queued at each layer. Generally, some number of in flight I/Os can be issued at each layer and if the number of I/O requests exceeds that number, they reside in a wait queue until the required resource becomes available. So there is essentially an “in process” queue and a “wait” queue at each layer (SDD and SDDPCM are a little more complicated).

At the file system layer, file system buffers limit the maximum number of in flight I/Os for each file system. At the LVM device driver layer, `hdisk` buffers limit the number of in flight I/Os. At the SDD layer, I/Os are queued if the data path optimizer (`dpo`) device's attribute, `qdepth_enable`, is set to `yes` (which it is by default). The data path optimizer is a pseudo device, which is the pseudo parent of the `vpaths` that allows you to control I/O flow to SDD path devices. Example 5-12 shows how to list the parameters for the `dpo` device's attribute.

*Example 5-12 Listing the `qdepth_enable` parameter for the `dpo` device's attribute*

```
#lsattr -El dpo
2062_max_luns 512 Maximum LUNS allowed for 2062 False
2105_max_luns 1200 Maximum LUNS allowed for 2105 True
2145_max_luns 512 Maximum LUNS allowed for 2145 False
persistent_resv yes Subsystem Supports Persistent Reserve Command False
qdepth_enable yes Queue Depth Control True
```

With certain database applications, such as an application running with an IBM DB2® database, IBM Lotus® Notes®, IBM Informix® database, the software might generate many threads, which can send heavy I/O to a relatively small number of devices.

Executing queue depth logic to control I/O flow can cause performance degradation, or even a system hang. If you are executing queue depth logic, use the `qdepth_enable` attribute to disable this queue depth logic on I/O flow control. This removes the limit on the amount of I/O sent to vpath devices. Some releases of SDD do not queue I/Os so it depends on the release of SDD.

SDDPCM, on the other hand, does not queue IOs before sending them to the disk device driver. The hdisks have a maximum number of in flight I/Os that's specified by its `queue_depth` attribute. And FC adapters also have a maximum number of in flight I/Os specified by `num_cmd_elems`. The disk subsystems themselves queue IOs and individual physical disks can accept multiple I/O requests but only service one at a time.

For more information about how to tune the `queue_depth` at the different levels, see the following website:

<http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/TD105745>

### 5.3.2 timeout\_policy attribute

Starting from SDDPCM v2.6.3.0, the path selection and management algorithm has enhanced to reduce this I/O performance degradation as described before. A new device attribute “`timeout_policy`” is introduced. This attribute provides several options to manage a path connected to an unstable FC port in different ways, to avoid being toggled between Open and FAILED states frequently.

This feature is designed to reduce the I/O performance degradation symptom, caused by intermittently failing links, which can result in paths cycling between the online and offline states. Under this failing link condition, I/O fails with TIMEOUT error, and the path was put to FAILED state, then subsequently brought back online following the successful completion of a health check command. The recovered path being selected for IO will fail again a short time later, due to intermittent link failure condition. Furthermore, the TIMEOUT error triggers FC recovery commands, and the recovery commands also experience TIMEOUT error. This TIMEOUT/Recovery storm degrades I/O performance severely. In the worst case, it might cause application failure.

A path (not the last path) will be set to FAILED state if I/O timed out. To recover a failed path due to TIMEOUT error, the algorithm is different based on the `timeout_policy` setting:

- ▶ `retry_path`: The algorithm works the same way as previous version, when a health check command succeeds, a failed path due to TIMEOUT will be recovered immediately.
- ▶ `fail_path`: This setting requires two consecutive successful health check commands to recover a failed path due to TIMEOUT.
- ▶ `disable_path`: If a TIMEOUT failed path continuously experiences health check command TIMEOUT within a certain period of time, the failed path due to TIMEOUT will be set to DISABLED (OFFLINE) state. It will stay in the DISABLED state, until user manually recovers it.

**Tip:** The default setting of “`timeout_policy`” is “`fail_path`”

#### Considerations for fail\_over:

- ▶ The “fail\_path” and “disable\_path” options are not supported under the “fail\_over” algorithm. Therefore, the user who wants to change the device algorithm to “fail\_over” must change the “timeout\_policy” to “retry\_path” first; otherwise, the changing device algorithm command will fail.
- ▶ In case the device’s algorithm is set to “fail\_over” with SDDPCM version before 2.6.3.0, and the user wants to upgrade SDDPCM to v2.6.3.x, then additional steps are required to properly configure these devices. For detailed information, see the following SDDPCM flash:

[http://www-01.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=D600&uid=s5g1S1004072&loc=en\\_US&cs=utf-8&lang=en](http://www-01.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=D600&uid=s5g1S1004072&loc=en_US&cs=utf-8&lang=en)

### 5.3.3 max\_xfer\_size attribute

There are some cases where there is too much I/O and the adapter cannot handle all of the requests at once. It is common to find errors indicating that the host adapter is unable to activate an I/O request on the first attempt. The most likely cause of these errors is that the host is low on resources.

To reduce the incidence of these errors, follow these examples to increase the resources by modifying the maximum transfer size attribute for the adapter. The default max\_xfer size value is 0x1000000.

It is possible to check the values as shown in Example 5-13.

*Example 5-13 Checking the max\_xfer\_size value*

---

```
# lsattr -El fcs0 |grep max_xfer_size
max_xfer_size 0x100000    Maximum Transfer Size                True
```

---

**Tip:** To view the range of allowed values for the attribute, type:

```
lsattr -Rl adapter_name -a max_xfer_size
```

Example 5-14 shows how to increase the size of the setting.

*Example 5-14 Changing the max\_xfer\_size parameter*

---

```
#chdev -l fcs0 -P -a max_xfer_size=0x400000
fcs0 changed
```

---

**Tip:** After changing the setting, it will then require a host reboot to take effect.

### 5.3.4 Storage unit with multiple IBM Power Systems hosts running AIX

On a host with non-MPIO SDD installed, a reserve bit is placed on all disks that are part of the volume group when that volume group is varied online. But with MPIO SDDPCM, the no\_reserve policy in AIX is changed to no\_reserve. It means that if the rootvg volume is shared with another host, the data might be accidentally damaged.

## 5.4 Multipathing with AIX

Information in this section provides configurations and settings for AIX hosts with DS8000. The DS8000 supports two methods of attaching AIX hosts:

- ▶ Subsystem Device Driver (SDD)
- ▶ AIX multipath I/O (MPIO) with a DS8000-specific Path Control Module (SDDPCM)

SDD and SDDPCM cannot be installed on the same AIX server.

On an AIX server, you need to have installed either the SDD or SDDPCM file sets for the DS8000 volumes:

- ▶ Filesets for SDD
  - devices.fcp.disk.ibm.rte
  - devices.sdd.61.rte (here 61 stands for AIX6.1x)
- ▶ Filesets for SDDPCM
  - devices.fcp.disk.ibm.mpio.rte
  - devices.sddpcm.71.rte (here 71 stands for AIX7.1)

The SDD or SDDPCM file sets require separate DS8000 host attachment file sets (ibm.fcp.disk.ibm) installed as a prerequisite. You can only have SDD or SDDPCM file sets installed on a given AIX server, not both. Also notice that the separate SDD or SDDPCM file sets are dependent on the AIX release.

See the SDD or SDDPCM support matrix at the following website for further information and file set downloads:

<http://www.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=DA400&uid=ssg1S7001350>

Also see the readme file of your SDD or SDDPCM version for latest updates and prerequisites. New SDD or SDDPCM releases typically require a specific minimum version of the host attachment file set. For supported SDD and SDDPCM releases with IBM System Storage DS8000 series environments on various AIX releases, see the IBM SSIC website at this website:

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

For the management of MPIO devices in AIX 6.1, see the following online documentation:

[http://publib.boulder.ibm.com/infocenter/aix/v6r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/dm\\_mpio.htm](http://publib.boulder.ibm.com/infocenter/aix/v6r1/index.jsp?topic=/com.ibm.aix.baseadm/doc/baseadmndita/dm_mpio.htm)

### 5.4.1 SDD for AIX

**SDD to SDDPCM migration:** AIX Version 7.1 does not support the IBM Subsystem Device Driver (SDD) for IBM TotalStorage IBM Enterprise Storage Server®, the IBM TotalStorage DS family, and the IBM System Storage SAN Volume Controller. If you are using SDD, you must transition to Subsystem Device Driver Path Control Module (SDDPCM) or AIX Path Control Module (PCM) for the multipath support on AIX for IBM SAN storage. SDD to SDDPCM migration scripts are available to help you with the transition.



Subsystem Device Driver for AIX does not use native AIX MPIO. Every path to each disk or volume is represented by a logical hdisk device in a system. On top of the paths, the new logical device vpath is created, which must be used as a logical device instead of hdisk, as shown in Example 5-15.

*Example 5-15 lsvpcfg command*

---

```
# lsvpcfg
vpath0 (Avail pv vg_2200) 75TV1812200 = hdisk2 (Avail ) hdisk6 (Avail ) hdisk10 (Avail ) hdisk14 (Avail )
vpath1 (Avail pv vg_2201) 75TV1812201 = hdisk3 (Avail ) hdisk7 (Avail ) hdisk11 (Avail ) hdisk15 (Avail )
vpath2 (Avail pv vg_2300) 75TV1812300 = hdisk4 (Avail ) hdisk8 (Avail ) hdisk12 (Avail ) hdisk16 (Avail )
vpath3 (Avail pv vg_2301) 75TV1812301 = hdisk5 (Avail ) hdisk9 (Avail ) hdisk13 (Avail ) hdisk17 (Avail )
```

---

**Tip:** To create a volume group using SDD, always use **mkvg4vp**, instead of the AIX default **mkvg** command. The vpath devices have Port VLAN ID (PVID), but the disks do not.

As shown in Example 5-16, paths are identified as an IBM FC 2107 device, while vpath devices are identified as Data Path Optimizer Pseudo Device Driver.

*Example 5-16 List disk devices on an AIX system using the lsdev command*

---

```
# lsdev -Cc disk
hdisk2 Available 30-T1-01 IBM FC 2107
hdisk3 Available 30-T1-01 IBM FC 2107
hdisk4 Available 30-T1-01 IBM FC 2107
hdisk5 Available 30-T1-01 IBM FC 2107
hdisk6 Available 30-T1-01 IBM FC 2107
hdisk7 Available 30-T1-01 IBM FC 2107
hdisk8 Available 30-T1-01 IBM FC 2107
hdisk9 Available 30-T1-01 IBM FC 2107
hdisk10 Available 31-T1-01 IBM FC 2107
hdisk11 Available 31-T1-01 IBM FC 2107
hdisk12 Available 31-T1-01 IBM FC 2107
hdisk13 Available 31-T1-01 IBM FC 2107
hdisk14 Available 31-T1-01 IBM FC 2107
hdisk15 Available 31-T1-01 IBM FC 2107
hdisk16 Available 31-T1-01 IBM FC 2107
hdisk17 Available 31-T1-01 IBM FC 2107
vpath0 Available Data Path Optimizer Pseudo Device Driver
vpath1 Available Data Path Optimizer Pseudo Device Driver
vpath2 Available Data Path Optimizer Pseudo Device Driver
vpath3 Available Data Path Optimizer Pseudo Device Driver
```

---

Management of SDD devices requires more steps, in comparison to SDDPCM devices. To remove a specific volume from the system, remove all hdisks that represent paths to a volume, and then the vpath.

To identify, which hdisk identifies which path, use the **datapath query essmap** command, or the **datapath query device** command, which provides additional device information. In Example 5-17, device information is provided for devices 0 and 1.

*Example 5-17 datapath query device*

---

```
# datapath query device 0 1
DEV#: 0 DEVICE NAME: vpath0 TYPE: 2107900 POLICY: Optimized
SERIAL: 75TV1812200
=====
```

Path#	Adapter/Hard Disk	State	Mode	Select	Errors
0	fscsi0/hdisk2	OPEN	NORMAL	1297	0
1	fscsi0/hdisk6	OPEN	NORMAL	1336	0
2	fscsi1/hdisk10	OPEN	NORMAL	1284	0
3	fscsi1/hdisk14	OPEN	NORMAL	1347	0

DEV#: 1 DEVICE NAME: vpath1 TYPE: 2107900 POLICY: Optimized  
SERIAL: 75TV1812201

---

Path#	Adapter/Hard Disk	State	Mode	Select	Errors
0	fscsi0/hdisk3	OPEN	NORMAL	927	0
1	fscsi0/hdisk7	OPEN	NORMAL	985	0
2	fscsi1/hdisk11	OPEN	NORMAL	940	0
3	fscsi1/hdisk15	OPEN	NORMAL	942	0

Here are some additional features of SDD for AIX:

- ▶ Enhanced SDD configuration methods and migration:

SDD has a feature in the configuration method to read the PVID from the physical disks and convert the PVID from hdisks to vpaths during the SDD vpath configuration. With this feature, you can skip the process of converting the PVID from hdisks to vpaths after configuring SDD devices. Furthermore, SDD migration can skip the PVID conversion process. This tremendously reduces the SDD migration time, especially with a large number of SDD devices and in an logical volume manager (LVM) configuration environment.

- ▶ Migration option for large device configuration:

SDD offers an environment variable SKIP\_SDD\_MIGRATION to customize the SDD migration or upgrade to maximize performance. The SKIP\_SDD\_MIGRATION environment variable is an option available to permit the bypass of the SDD automated migration process backup, restoration, and recovery of LVM configurations and SDD device configurations. This variable can help decrease the SDD upgrade time if you reboot the system after upgrading SDD.

## 5.4.2 SDDPCM for AIX

The base functionality of MPIO is limited, but provides an interface for vendor-specific PCMs that allow for implementation of advanced algorithms. IBM provides a PCM for DS8000 and other devices, that enhances MPIO with all the features of the original SDD.

### Benefits of MPIO

There are several benefits of MPIO with SDDPCM, compared to traditional SDD:

- ▶ Performance improvements due to direct integration with AIX
- ▶ Better integration if separate storage systems are attached
- ▶ Easier administration

### Requirements and considerations

The following requirements and considerations apply for MPIO:

- ▶ Default MPIO is not supported on DS8000, SDDPCM is required.
- ▶ Both SDDPCM and SDD cannot be installed on the same AIX server.
- ▶ If you use MPIO with SDDPCM instead of SDD, remove the regular DS8000 host attachment script and install the MPIO version. The MPIO version identifies the DS8000 volumes to the operating system as MPIO manageable.

In comparison to SDD, where each path is represented by `hdisk` and a special `vpath` device is created, SDDPCM does not create a special device for every path. See Example 5-18.

*Example 5-18 The `pcmpath` query device*

```
# pcmpath query device 12 13

DEV#: 12 DEVICE NAME: hdisk12 TYPE: 2107900 ALGORITHM: Load Balance
SERIAL: 75TV1816200
=====
Path#      Adapter/Path Name      State   Mode   Select   Errors
  0         fscsi0/path0          CLOSE  NORMAL     0         0
  1         fscsi0/path1          CLOSE  NORMAL     0         0
  2         fscsi1/path2          CLOSE  NORMAL     0         0
  3         fscsi1/path3          CLOSE  NORMAL     0         0

DEV#: 13 DEVICE NAME: hdisk13 TYPE: 2107900 ALGORITHM: Load Balance
SERIAL: 75TV1816201
=====
Path#      Adapter/Path Name      State   Mode   Select   Errors
  0         fscsi0/path0          CLOSE  NORMAL     0         0
  1         fscsi0/path1          CLOSE  NORMAL     0         0
  2         fscsi1/path2          CLOSE  NORMAL     0         0
  3         fscsi1/path3          CLOSE  NORMAL     0         0
```

This method significantly reduces number of devices in the system (see Example 5-19) and helps with managing devices. For example, when removing a physical volume from a system, using `rmdev -d1 hdiskX` is sufficient.

*Example 5-19 `lsdev` command*

```
# lsdev -Cc disk
[...]
hdisk12 Available 00-08-02 IBM MPIO FC 2107
hdisk13 Available 00-08-02 IBM MPIO FC 2107
hdisk14 Available 00-08-02 IBM MPIO FC 2107
hdisk15 Available 00-08-02 IBM MPIO FC 2107
hdisk16 Available 00-08-02 IBM MPIO FC 2107
hdisk17 Available 00-08-02 IBM MPIO FC 2107
hdisk18 Available 00-08-02 IBM MPIO FC 2107
hdisk19 Available 00-08-02 IBM MPIO FC 2107
hdisk20 Available 00-08-02 IBM MPIO FC 2107
hdisk21 Available 00-08-02 IBM MPIO FC 2107
```

For path management, you can use AIX built-in commands such as `chpath`, `rmpath`, or you might find the SDDPCM `pcmpath` command more flexible.

Starting from SDDPCM version 2.4.0.3, a new device attribute “`retry_timeout`” is added for ESS/DS6K/DS8K/SVC devices. This attribute allows the user to set the time-out value for I/O retry on the last path. The default value of this attribute is 120 seconds and it is user-changeable with the valid range of 30 to 600 seconds. The `pcmpath` command provides a new CLI command to dynamically change this device `retry_timeout` attribute.

Here is the syntax of this command:

```
pcmpath set device <device number> retry_timeout <time>
```

When only num is specified for the device number, then the command applies to the `hdisk` specified by that num.

This feature enables the user to adjust the time-out value to control how long SDDPCM will retry the I/O on the last path before it fails to the application.

In the situation where a device loss of access is only temporary, the `retry_timeout` value might need to be set to a higher value.

### SDDPCM server daemon

The SDDPCM server (also referred to as `pcmsrv`) is an integrated component of SDDPCM 2.1.0.0 (or later). This component consists of a UNIX application daemon that is installed in addition to the SDDPCM path control module. The SDDPCM server daemon provides a path-recovery function for SDDPCM devices and the First Time Data Capture function. After you have installed SDDPCM and restarted the system, verify if the SDDPCM server (`pcmsrv`) has automatically started as shown in Example 5-20.

*Example 5-20 verifying if SDDPCM has automatically started*

---

```
# lssrc -s pcmsrv
Subsystem      Group          PID           Status
pcmsrv         pcmsrv         4718636      active
```

---

It is possible to start the SDDPCM server manually as follows:

- ▶ For SDDPCM 3.0.0.0 or later releases, the command to start the server is:  
**startpcmsrv**
- ▶ For SDDPCM 2.6.0.1 or prior releases, the command to start the server is:  
**startsrv -s pcmsrv -e XPG\_SUS\_ENV=0N**

It is possible as well to stop the SDDPCM server manually as follows:

- ▶ For SDDPCM 3.0.0.0 or later releases, the command to stop the server is:  
**stoppcmsrv**
- ▶ For SDDPCM 2.6.0.1 or prior releases, the command to stop the server is:  
**stopsrc -s pcmsrv**

For more details about the SDDPCM multipath driver, see the *Multipath Subsystem Device Driver User's Guide*, GC52-1309-03.

## 5.5 Configuring LVM

This section provides information and considerations when configuring LVM on AIX servers. In AIX, all storage is managed by the AIX LVM. It virtualizes physical disks to dynamically create, delete, resize, and move logical volumes for application use. With AIX, DS8000 logical volumes appear as physical SCSI disks.

### 5.5.1 LVM striping

*Striping* is a technique for spreading the data in a logical volume across several physical disks in such a way that all disks are used in parallel to access data on one logical volume. The primary objective of striping is to increase the performance of a logical volume beyond that of a single physical disk.

LVM striping can be used to distribute data across more than one array or rank, in case of single rank extent pools.

## 5.5.2 Inter-physical volume allocation policy

Inter-physical volume allocation is one of the simplest and most advisable methods to spread the workload accesses across physical resources. Most LVMs offer inter-disk allocation policies for logical volumes. This method can also be recognized when the term *physical partition spreading* is used.

With AIX LVM, one or more volume groups can be created using the physical disks, which are logical volumes on the DS8000. LVM organizes volume group space in physical partitions. Use physical partition allocation for the logical volumes in rotating order, known as round-robin. With this method, the first free extent is allocated from the first available physical volume. The next free extent is allocated from the next available physical volume, and so on. If the physical volumes have the same size, optimal I/O load distribution among the available physical volumes can be achieved.

## 5.5.3 LVM mirroring

LVM has the capability to mirror logical volumes across several physical disks. This improves availability, because when a disk fails, there is another disk with the same data. When creating mirrored copies of logical volumes, ensure that the copies are distributed across separate disks.

With the introduction of SAN technology, LVM mirroring can help provide protection against a site failure. Using longwave Fibre Channel connections, a mirror can be stretched up to a 10 km (6.2 mi.) distance.

## 5.5.4 Impact of DS8000 storage pool striping

Starting with DS8000 licensed machine code 5.3.xx.xx, it is possible to stripe the extents of a DS8000 logical volume across multiple RAID arrays. DS8000 storage pool striping can improve throughput for certain workloads. It is performed on a 1 GB granularity, so it will generally benefit random workloads more than consecutive workloads.

If you are already using a host-based striping method, for example, LVM striping or DB2 database container striping, you do not need to use storage pool striping, but it is possible. If you use a host-based striping method and a storage pool striping method, then combine the wide stripes on DS8000 with small granularity stripes on the host. The preferred size for these is usually between 8 and 64 MB. If large stripes on both DS8000 and attached host interfere with each other, I/O performance might be affected.

## 5.6 Using AIX access methods for I/O

AIX provides several modes to access data in a file system. It can be important for performance to select the right access method. Use the information provided in this section to help determine your access method.

### 5.6.1 Synchronous I/O

Synchronous I/O occurs while you wait. An application's processing cannot continue until the I/O operation is complete. It is a more secure and traditional way of handling data. It helps ensure consistency at all times, but can affect performance. It also does not allow the operating system to take full advantage of functions of modern storage devices, such as queuing, command reordering, and so on.

## 5.6.2 Asynchronous I/O

Asynchronous I/O operations run in the background and do not block user applications. This improves performance, because I/O and application processing run simultaneously. Many applications, such as databases and file servers, take advantage of the ability to overlap processing and I/O; although, they need to take measures to ensure data consistency. You can configure, remove, and change asynchronous I/O for each device by using the **chdev** command or System Management Interface Tool (SMIT).

**Tip:** If the number of asynchronous I/O requests is high, then increase `maxservers` to approximately the number of simultaneous I/Os that there might be. In most cases, it can be better to leave the `minservers` parameter at the default value, because the asynchronous I/O kernel extension will generate additional servers, if needed. When viewing the CPU utilization of the asynchronous I/O servers, if the utilization is even across all servers, all servers are being used and you might want to try increasing their asynchronous I/O number. Running `psstat -a`, helps you view the asynchronous I/O servers by name, and running `ps -k` shows them to you as the name `kproc`.

## 5.6.3 Concurrent I/O

In 2003, IBM introduced a new file system feature, called *concurrent I/O* (CIO), for the second generation of the journaled file system (JFS2). CIO includes all of the advantages of direct I/O and relieves the serialization of write accesses. It helps improve performance for many environments, particularly commercial relational databases. In many cases, the database performance achieved using CIO with JFS2 is comparable to that obtained by using raw logical volumes.

A method for enabling the concurrent I/O mode is to use the `mount -o cio` command when mounting a file system.

## 5.6.4 Direct I/O

An alternative I/O technique, called *direct I/O*, bypasses the virtual memory manager (VMM) altogether and transfers data directly from the user's buffer to the disk and from the disk to the user's buffer. The concept behind it is similar to raw I/O in the sense that they both bypass caching at the file system level. This reduces the CPU's processing impact and helps provide more memory available to the database instance, which can make more efficient use of it for its own purposes.

Direct I/O is provided as a file system option in JFS2. It can be used either by mounting the corresponding file system with the `mount -o dio` command, or by opening a file with the `O_DIRECT` flag specified in the `open()` system call. When a file system is mounted with the `-o dio` option, all files in the file system use direct I/O by default.

Direct I/O benefits applications that have their own caching algorithms by eliminating the impact of copying data twice, first between the disk and the operating system buffer cache, and then from the buffer cache to the application's memory. For applications that benefit from the operating system cache, do not use direct I/O, because all I/O operations are synchronous. Direct I/O also bypasses the JFS2 read-ahead. Read-ahead can provide a significant performance increase for consecutively accessed files.

## 5.7 Expanding dynamic volume with AIX

Starting with IBM DS8000 licensed machine code 5.3.xx.xx, it is possible to expand a logical volume in size, without taking the volume offline. Additional actions are required on the attached host to make use of the extra space. This section describes the required AIX logical volume manager (LVM) tasks.

After the DS8000 logical volume is expanded, use the **chvg** command with the **-g** option to examine all the disks in the volume group to see if they have grown in size. If the disks have grown in size, then the **chvg** command attempts to add additional physical partitions to the AIX physical volume that corresponds to the expanded DS8000 logical volume.

Example 5-21 shows an AIX file system that was created on a single DS8000 logical volume. The DSCLI is used to display the characteristics of the DS8000 logical volumes; AIX LVM commands show the definitions of volume group, logical volume, and file systems. The available space for the file system is almost gone.

*Example 5-21 DS8000 logical volume and AIX file system before dynamic volume expansion*

```
dsccli> lsfbvol 4700
Name          ID  acstate  datastate  configstate  deviceMTM  datatype  extpool  cap (2^30B)
cap (10^9B)  cap (blocks)
=====
ITS0_p770_1_4700 4700 Online   Normal    Normal     2107-900  FB 512   P53      18.0
-          37748736

# lsvg -p dvevg
dvevg:
PV_NAME      PV STATE      TOTAL PPs    FREE PPs     FREE DISTRIBUTION
hdisk0       active        286          5            00..00..00..00..05
# lsvg -l dvevg
dvevg:
LV NAME      TYPE          LPs  PPs  PVs  LV STATE    MOUNT POINT
dvelv       jfs2         280  280  1   open/syncd  /dvefs
loglv00     jfs2log      1    1    1   open/syncd  N/A
# lsfs /dvefs
Name          Nodename  Mount Pt          VFS  Size  Options  Auto Accounting
/dev/dvelv    --        /dvefs            jfs2 36700160 rw      yes  no
```

If more space is required in this file system, two options are available with the AIX operating system: either add another DS8000 logical volume (which is a physical volume on AIX) to the AIX volume group, or extend the DS8000 logical volume and subsequently adjust the AIX LVM definitions. The second option is demonstrated in Example 5-22. The DSCLI is used to extend the DS8000 logical volume. On the attached AIX host, the configuration is changed with the **cfgmgr** and **chvg** AIX commands. Afterwards, the file system is expanded online and the results are displayed.

*Example 5-22 Dynamic volume expansion of DS8000 logical volume and AIX file system*

```
dsccli> chfbvol -cap 24 4700
CMUC00332W chfbvol: Some host operating systems do not support changing the volume
size. Are you sure that you want to resize the volume? [y/n]: y
CMUC00026I chfbvol: FB volume 4700 successfully modified.
# cfgmgr
# chvg -g dvevg
# lsvg -p dvevg
dvevg:
PV_NAME      PV STATE      TOTAL PPs    FREE PPs     FREE DISTRIBUTION
```

```

hdisk0          active          382          101          00..00..00..24..77
# chfs -a size=4500000 /dvefs
Filesystem size changed to 45088768
# lsvg -p dvevg
dvevg:
PV_NAME          PV STATE          TOTAL PPs   FREE PPs   FREE DISTRIBUTION
hdisk0          active            382         37         00..00..00..00..37
# lsvg -l dvevg
dvevg:
LV NAME          TYPE              LPs   PPs   PVs  LV STATE      MOUNT POINT
dvelv            jfs2              344   344   1    open/syncd    /dvefs
loglv00          jfs2log           1     1     1    open/syncd    N/A
# lsfs /dvefs
Name             Nodename  Mount Pt          VFS  Size  Options  Auto
Accounting
/dev/dvelv       --        /dvefs            jfs2 45088768 rw      yes
no

```

---

For the online size extension of the AIX volume group, you might need to deactivate and then reactivate the AIX volume group for LVM to see the size change on the disks. See the AIX documentation for more information at this website:

<http://publib16.boulder.ibm.com/pseries/index.htm>

**Tip:** The `-g` option in the `chvg` command was not valid for concurrent vgs or for rootvg. That restriction was removed on AIX release 6.1 TL3 or later.

## 5.8 SAN boot support

The DS8000 is supported as a boot device on Power Systems servers that have Fibre Channel boot capability. For additional information, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, GC27-2298-02





## Linux considerations

This chapter presents the specifics of attaching IBM System Storage DS8000 systems to host systems running Linux.

The following topics are covered:

- ▶ Working with Linux and DS8000
- ▶ Attaching to a basic host
- ▶ Resizing DS8000 volumes dynamically
- ▶ Using FlashCopy and remote replication targets
- ▶ Troubleshooting and monitoring

Additionally, this chapter provides information about the following hardware architectures that are supported for DS8000 attachment:

- ▶ Intel x86 and x86\_64
- ▶ IBM Power Systems
- ▶ IBM System z®

Although older Linux versions can work with the DS8000, this chapter covers the most recent enterprise level distributions, such as these products:

- ▶ Novell SUSE Linux Enterprise Server 11, Service Pack 1 (SLES11 SP1)
- ▶ Red Hat Enterprise Linux (RHEL) 5, Update 5 (Red Hat Enterprise Linux 5U5), Red Hat Enterprise Linux 6.0, and Red Hat Enterprise 6.1.

## 6.1 Working with Linux and DS8000

Linux is an open source, UNIX-like operating system that uses the Linux kernel. This chapter presents information for working with the DS8000 and the Linux operating system. The Linux kernel, along with the tools and software needed to run an operating system, are maintained by a loosely organized community of thousands of mostly volunteer programmers.

### 6.1.1 How Linux differs from other operating systems

Linux differs from the other proprietary operating systems in many ways, such as the following factors:

- ▶ No one person or organization can be held responsible or called for support.
- ▶ Depending on the target group, the distributions differ in the kind of support that is available.
- ▶ Linux is available for almost all computer architectures.
- ▶ Linux is rapidly evolving.

These factors present challenges for generic support of Linux. Therefore, the product integration support approach by IBM helps limit the uncertainty and the amount of testing.

IBM supports the following Linux distributions that are targeted at enterprise clients:

- ▶ Red Hat Enterprise Linux
- ▶ SUSE Linux Enterprise Server

These distributions have major release cycles of about 18 months. These release cycles are maintained for five years and require you to sign a support contract with the distributor. They also have a schedule for regular updates. These factors help reduce the amount of issues due to the lack of support listed previously. Also, the limited number of supported distributions enables IBM to work closely with the vendors to ensure interoperability and support. Details about the supported Linux distributions can be found in the IBM SSIC matrix at the following website:

<http://www.ibm.com/systems/support/storage/ssic/interoperability.wss>

### 6.1.2 Attaching Linux server to DS8000 resources

Use the information provided in the following sections to help attach Linux server to a DS8000 storage subsystem.

#### **The DS8000 Host Systems Attachment Guide**

The *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917 provides information about preparing an Intel IA-32-based machine for DS8000 attachment, including the following instructions:

- ▶ Working with the unique Linux SCSI subsystem
- ▶ Installing and configuring the Fibre Channel HBA
- ▶ Discovering DS8000 volumes
- ▶ Setting up multipathing
- ▶ Preparing a system that boots from the DS8000

For more information, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917, at the following website:

<http://www.ibm.com/support/docview.wss?uid=ssg1S7001161>

This publication does not cover other hardware platforms, such as the IBM Power Systems or IBM System z.

## Online Storage Reconfiguration Guide

The *Online Storage Reconfiguration Guide* is part of the documentation provided by Red Hat for Red Hat Enterprise Linux 5 and 6. Although written specifically for Red Hat Enterprise Linux 5, most of the general information in this guide is valid for Linux 6 as well. This guide covers the following topics for Fibre Channel and iSCSI attached devices:

- ▶ Persistent device naming
- ▶ Dynamically adding and removing storage devices
- ▶ Dynamically resizing storage devices
- ▶ Low level configuration and troubleshooting

This guide is available at the following websites:

- ▶ Red Hat 5:

[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/html/Online\\_Storage\\_Reconfiguration\\_Guide/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/Online_Storage_Reconfiguration_Guide/index.html)

- ▶ Red Hat 6:

[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/pdf/Storage\\_Administration\\_Guide/Red\\_Hat\\_Enterprise\\_Linux-6-Storage\\_Administration\\_Guide-en-US.pdf](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/pdf/Storage_Administration_Guide/Red_Hat_Enterprise_Linux-6-Storage_Administration_Guide-en-US.pdf)

## DM Multipath Configuration and Administration

The Red Hat document, *DM Multipath Configuration and Administration*, is also part of the information provided for Red Hat Enterprise Linux 5. In this document, you can find general information for Red Hat Enterprise Linux 5, in addition to other Linux operating systems, for device-mapper multipathing (DM Multipath), such as the following topics:

- ▶ How DM Multipath works
- ▶ How to set up and configure DM Multipath within Red Hat Enterprise Linux 5
- ▶ Troubleshooting DM Multipath

This documentation is available at the following website:

[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/5/html/DM\\_Multipath/index.html](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/5/html/DM_Multipath/index.html)

For Red Hat Enterprise 6 DM Multipath documentation, see the following website:

[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/pdf/DM\\_Multipath/Red\\_Hat\\_Enterprise\\_Linux-6-DM\\_Multipath-en-US.pdf](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/pdf/DM_Multipath/Red_Hat_Enterprise_Linux-6-DM_Multipath-en-US.pdf)

## SLES 11 SP2: Storage Administration Guide

The *SLES 11 SP2: Storage Administration Guide* is part of the documentation for Novell SUSE Linux Enterprise Server 11, Service Pack 2. Although written specifically for SUSE Linux Enterprise Server, this guide contains useful storage related information for Linux, such as the following topics:

- ▶ Setting up and configuring multipath I/O
- ▶ Setting up a system to boot from multipath devices
- ▶ Combining multipathing with LVM and Linux software RAID

This guide can be found at the following website:

[http://www.suse.com/documentation/sles11/stor\\_admin/?page=documentation/sles11/stor\\_admin/data/bookinfo.html](http://www.suse.com/documentation/sles11/stor_admin/?page=documentation/sles11/stor_admin/data/bookinfo.html)

## IBM wiki for Linux on the Power Architecture system

IBM maintains a wiki that contains information such as a discussion forum, announcements, and technical articles about Linux on IBM Power Architecture® systems.

This wiki can be found at the following website:

<https://www.ibm.com/developerworks/wikis/display/LinuxP/Home>

## Fibre Channel Protocol for Linux and z/VM on IBM System z

The IBM Redpaper™ publication, *Getting Started with zSeries Fibre Channel Protocol*, REDP-0205, provides comprehensive information about storage attachment with Fibre Channel for IBM z/VM®, and Linux on z/VM. It covers the following topics:

- ▶ General Fibre Channel Protocol (FCP) concepts
- ▶ Setting up and using FCP with z/VM and Linux
- ▶ FCP naming and addressing schemes
- ▶ FCP devices in the 2.6 Linux kernel
- ▶ N-Port ID virtualization
- ▶ FCP security topics

You can find this Redpaper publication at the following website:

<http://www.redbooks.ibm.com/abstracts/sg247266.html>

## Other sources of information

Additional information from IBM and Linux distributors, about working with the Linux platform, is provided in the following sections.

The Linux distributors' documentation pages are good starting points for installation, configuration, and administration of Linux servers, especially when working with server platform specific issues. This documentation can be found at the following websites:

- ▶ Novell SUSE Linux Enterprise Server:  
<http://www.novell.com/documentation/suse.html>
- ▶ Red Hat Enterprise Linux:  
<http://www.redhat.com/docs/manuals/enterprise/>

For more information about storage attachment using FCP, see the System z I/O Connectivity page, at the following website:

<http://www.ibm.com/systems/z/connectivity/products/>

The *IBM System z Connectivity Handbook*, SG24-5444, contains information about connectivity options available for use within and beyond the data center for IBM System z servers. This handbook has a section for Fibre Channel attachment. You can access this book at the following website:

<http://www.redbooks.ibm.com/redbooks.nsf/RedbookAbstracts/sg245444.html>

### 6.1.3 Understanding storage related improvements to Linux

This section provides a summary of storage related improvements that were introduced to Linux in recent years, and details about usage and configuration.

#### Limitations and issues of older Linux versions

The following list provides limitations and issues documented with older Linux versions. These limitations are not valid with newer releases:

- ▶ Limited number of devices that can be attached
- ▶ Gaps in LUN sequence leading to incomplete device discovery
- ▶ Limited dynamic attachment of devices
- ▶ Non-persistent device naming that can lead to re-ordering
- ▶ No native multipathing

#### Dynamic generation of device nodes

Linux uses special files, also called *device nodes* or *special device files*, for access to devices. In earlier versions, these files were created statically during installation. The creators of a Linux distribution tried to anticipate all devices that can be used for a system and created the nodes for them. This often led to a confusing number of existing nodes or missing nodes.

In recent versions of Linux, two new subsystems were introduced, *hotplug* and *udev*. Hotplug has the task to detect and register newly attached devices without user intervention and *udev* dynamically creates the required device nodes for them, according to predefined rules. In addition, the range of major and minor numbers, the representatives of devices in the kernel space, was increased and they are now dynamically assigned.

With these improvements, the required device nodes exist immediately after a device is detected, and only the device nodes needed are defined.

#### Persistent device naming

As mentioned earlier, *udev* follows predefined rules when it creates the device nodes for new devices. These rules are used to define device node names that relate to certain device characteristics. In the case of a disk drive, or SAN-attached volume, this name contains a string that uniquely identifies the volume. This string ensures that every time this volume is attached to the system, it gets the same name.

Therefore, the issue where devices received separate names, depending on the order they were discovered, is no longer valid with newer Linux releases.

#### Multipathing

Linux now has its own built in multipathing solution, DM Multipath. It is based on the *device mapper*, a block device virtualization layer in the Linux kernel. The device mapper is also used for other virtualization tasks, such as the logical volume manager, data encryption, snapshots, and software RAID.

DM Multipath overcomes the issues when only proprietary multipathing solutions existed:

- ▶ The proprietary multipathing solutions were only supported for certain kernel versions. Therefore, systems followed the distributions' update schedule.
- ▶ Proprietary multipathing was often binary only and not supported by the Linux vendors because they were unable to debug the issues.
- ▶ A mix of separate vendors storage systems on the same server, or even separate types of the same vendor, usually was not possible, because the multipathing solutions cannot coexist.

Today, DM Multipath is the only multipathing solution fully supported by both Red Hat and Novell for the enterprise Linux distributions. It is available on all hardware platforms and supports all block devices that can have more than one path. IBM has also adopted a strategy to support DM Multipath wherever possible.

**Tip:** The SDD, the IBM proprietary multipathing solution, is no longer updated for Linux.

### **Adding and removing volumes online**

With the new hotplug and udev subsystems, it is now possible to easily add and remove disks from Linux. SAN-attached volumes are usually not detected automatically, because adding a volume to a DS8000 volume group does not create a hotplug trigger event, such as inserting a USB storage device. SAN-attached volumes are discovered during the user initiated device scans and then automatically integrated into the system, including multipathing.

To remove a disk device before you physically detach it, make sure it is not in use and then remove it logically from the system.

### **Dynamic LUN resizing**

Recently, improvements were introduced to the SCSI layer and DM Multipath that allow resizing of SAN-attached volumes, when they are in use. Capabilities are still limited to certain cases.

## **6.2 Attaching to a basic host**

In this section, information is provided for making DS8000 volumes available to your Linux host. Additionally, separate methods of attaching storage for separate hardware architectures are explained, in addition to configuring the Fibre Channel HBA driver, setting up multipathing, and any required special settings.

### **6.2.1 Platform-specific information**

The most popular hardware platform for Linux, the Intel x86 (32 or 64 bit) architecture, only allows direct mapping of DS8000 volumes to the host through Fibre Channel fabrics and HBAs. The other platforms, such as IBM System z and IBM Power Systems, provide additional mapping methods to enable better exploitation of their much more advanced virtualization capabilities.

#### **IBM Power Systems**

Linux, running in a logical partition on an IBM Power System, can get storage from a DS8000 either directly through an exclusively assigned Fibre Channel HBA, or through a VIOS running on the system.

Direct attachment works the same way as documented previously with the other platforms. However, VIOS attachment requires specific considerations. For other details about the way VIOS works and how it is configured, see Chapter 4, “Virtual I/O Server considerations” on page 47.

For other publications that cover VIOS attachment on Power Systems, see these books:

- ▶ *PowerVM Virtualization on IBM System p: Introduction and Configuration Fourth Edition*, SG24-7940
- ▶ *IBM PowerVM Virtualization Managing and Monitoring*, SG24-7590

### **Virtual vscsi disks through VIOS**

Linux on IBM Power distributions contain a kernel module or driver for a virtual SCSI HBA, which attaches the virtual disks provided by the VIOS to the Linux system. This driver is called *ibmvscsi*. Example 6-1 shows how the devices are viewed by the Linux system.

*Example 6-1 Virtual SCSI disks*

---

```
p6-570-lpar13:~ # lsscsi
[0:0:1:0]   disk    AIX      VDASD          0001  /dev/sda
[0:0:2:0]   disk    AIX      VDASD          0001  /dev/sdb
```

---

The SCSI vendor ID is AIX, and the device model is VDASD. Apart from that, they are treated as any other SCSI disk. If you run a redundant VIOS setup on a machine, the virtual disks can be attached through both servers. The virtual disks will then show up twice and must be managed by DM Multipath to ensure data integrity and proper path handling.

### **Virtual Fibre Channel adapters through NPIV**

IBM PowerVM, the hypervisor of the IBM Power machine, can use the NPIV capabilities of modern SANs and Fibre Channel HBAs to provide virtual HBAs for the LPARs. The mapping of these to the LPARs is again done by the VIOS.

Virtual HBAs register to the SAN with their own WWPNs. To the DS8000, they look exactly like physical HBAs. You can create host connections for them and map volumes. This can result in easier, more streamlined storage management, and better isolation of the logical partition (LPAR) in an IBM Power machine.

Linux on IBM Power distributions come with a kernel module for the virtual HBA, which is called *ibmvfc*. After loaded, it presents the virtual HBA to the Linux operating system as though it were a real Fibre Channel HBA. DS8000 volumes that are attached to the virtual HBA appear exactly the same way as though they were connected through a physical adapter. Example 6-2 lists volumes mapped through NPIV virtual HBAs.

*Example 6-2 Volumes mapped through NPIV virtual HBAs*

---

```
p6-570-lpar13:~ # lsscsi
[1:0:0:0]   disk    IBM      2107900        3.23  /dev/sdc
[1:0:0:1]   disk    IBM      2107900        3.23  /dev/sdd
[1:0:0:2]   disk    IBM      2107900        3.23  /dev/sde
[2:0:0:0]   disk    IBM      2107900        3.23  /dev/sdm
[2:0:0:1]   disk    IBM      2107900        3.23  /dev/sdn
[2:0:0:2]   disk    IBM      2107900        3.23  /dev/sdo
```

---

To maintain redundancy, use more than one virtual HBA, each one running on a separate real HBA. In this case, DS8000 volumes will show up once per path, and will need to be managed by a DM Multipath.

## System z

For Linux running on an IBM System z server, there are even more storage attachment choices and therefore potential confusion.

### ***Linux running natively in a System z LPAR***

To run Linux directly on a System z LPAR, two methods can be used to attach disk storage:

- ▶ IBM FICON® channel-attached count key data (CKD) devices:

Linux on System z supports channel attached CKD devices. These type of devices are used in traditional mainframe environments like z/OS and are called *direct access storage devices* (DASDs). DASDs are connected to the Linux LPAR using the regular System z methods with the I/O definition file (IODF), and hardware configuration definition (HCD). By design, Linux cannot use CKD devices. Its I/O access methods are all designed for *fixed block* (FB) storage, such as SCSI attached disks, as it is used in all open systems implementations. To overcome this, Linux on System z provides a kernel module for DASD. It creates a fixed block address range within a data set that resides on a CKD device. This emulated FB storage appears to the Linux system as though it were a SCSI disk.

For redundancy, DASDs usually are connected through more than one channel. The multipathing is managed in the I/O subsystem of the System z machine. Therefore, the Linux system only sees one device.

Linux also supports *parallel access volume* (PAV) with DASDs. This increases the throughput by enabling several I/O requests to the same device in parallel, such as the System z method of command queuing. In recent Linux distributions, SLES11, Red Hat Enterprise Linux 6, *dynamic PAV* is supported and managed by the DASD driver completely without user intervention. In older Linux versions, only *static PAV* is supported and must be configured manually. Additional details about PAV for Linux on System z from SHARE Inc., can be found at the following website:

<http://share.confex.com/share/115/webprogram/Session7154.html>

- ▶ FCP attached SCSI devices:

The FICON channel adapter cards in a System z machine can also operate in *Fibre Channel Protocol* (FCP) mode. FCP is the protocol that transports SCSI commands over the Fibre Channel interface. It is used in all open systems implementations for SAN-attached storage. Certain operating systems that run on a System z mainframe can use this FCP capability and connect directly to FB storage devices. Linux on System z provides the kernel module *zfc* to operate the FICON adapter in FCP mode. A channel card can only run either in FCP or FICON mode. Also, in FCP mode, it must be dedicated to a single LPAR and cannot be shared.

To maintain redundancy, use more than one FCP adapter to connect to the DS8000 volumes. Linux will see a separate disk device for each path and requires DM Multipath to manage them.

### ***Linux on System z running in a virtual machine under z/VM***

Running several of virtual Linux instances in a z/VM environment is much more common. The IBM z/VM operating system provides granular and flexible assignment of resources to the virtual machines (VMs), and also enables sharing of resources between VMs.



The z/VM offers several ways to connect storage to its VMs, such as the following methods:

- ▶ Channel attached DASD device:

The z/VM system can assign DASD devices directly to a virtual machine. If a VM runs Linux, it can access the device with the DASD driver. Multipathing is handled in the I/O subsystem; therefore, DM Multipath is not required. PAV is supported in the same method as for Linux running directly in an LPAR.

- ▶ z/VM minidisk:

A *minidisk* is a virtual DASD device that z/VM models from a storage pool it has assigned for its use. It can be attached to a VM. Within the VM, it is treated as though it were a real device. Linux uses the DASD driver for access.

- ▶ Fibre Channel (FCP) attached SCSI devices:

The z/VM system can also assign a Fibre Channel card running in FCP mode to a VM. A Linux instance running in this VM can operate the card using the *zfc* driver and access the attached DS8000 FB volumes.

To maximize the utilization of the FCP adapters, share them between more than one VM. However, z/VM cannot assign FCP attached volumes individually to virtual machines. Each VM can theoretically access all volumes that are attached to the shared FCP adapter. The Linux instances running in the VMs must ensure that each VM only uses the volumes that it is designated to use.

- ▶ FCP attachment of SCSI devices through NPIV:

NPIV was introduced for System z, z/VM, and Linux on System z, to enable multiple virtual Fibre Channel HBAs running on a single physical HBA. These virtual HBAs are assigned individually to virtual machines. They log on to the SAN with their own WWPNs. To the DS8000, they look exactly like physical HBAs. You can create host connections for them and map volumes. This method provides the ability to assign DS8000 volumes directly to the Linux virtual machine. No other instance can access these, even if it uses the same physical adapter card.

## 6.2.2 Configuring Fibre Channel attachment

In the following sections, information about how Linux is configured to access DS8000 volumes is presented. This section focuses on the Intel x86 platform, but also presents information where other platforms differ. Most examples shown are command line based.

### Loading the Linux Fibre Channel drivers

The following are four main brands of *Fibre Channel Host Bus Adapters* (FC HBAs):

- ▶ QLogic. These adapters are the most used HBAs for Linux on the Intel X86 platform. There is a unified driver for all types of QLogic FC HBAs. The name of the kernel module is *qla2xxx* and is included in the enterprise Linux distributions. The shipped version is supported for DS8000 attachment.
- ▶ Emulex. These adapters are sometimes used in Intel x86 servers and, rebranded by IBM, the standard HBA for the Power Systems platform. There also is a unified driver that works with all Emulex FC HBAs. The kernel module name is *lpfc*. A supported version is also included in the enterprise Linux distributions, for both Intel x86 and Power Systems.
- ▶ Brocade. *Converged network adapters* (CNA) that operate as FC and Ethernet adapters and are relatively new to the market. They are supported on the Intel x86 platform for FC attachment to the DS8000. The kernel module version provided with the current enterprise Linux distributions is not supported. You must download the supported version from the Brocade website. The driver package comes with an installation script that compiles and installs the module.

There might be support issues with the Linux distributor because of the modifications done to the kernel. The FC kernel module for the CNAs is called *bfa*. You can download the driver from this website:

<http://www.brocade.com/services-support/drivers-downloads/adapters/index.page>

- ▶ IBM FICON Express: The HBAs for the System z platform. They can either operate in FICON for traditional CKD devices, or FCP mode for FB devices. Linux deals with them directly only in FCP mode. The driver is part of the enterprise Linux distributions for System z and is called *zfcp*.

Kernel modules or drivers are loaded with the **modprobe** command. They can also be removed as long as they are not in use, as shown in Example 6-3.

*Example 6-3 Load and unload a Linux Fibre Channel HBA kernel module*

---

```
x36501ab9:~ # modprobe qla2xxx
x36501ab9:~ # modprobe -r qla2xxx
```

---

Upon loading, the FC HBA driver examines the FC fabric, detects attached volumes, and registers them in the operating system.

To see whether a driver is loaded and the dependencies for it, use the **lsmod** command (Example 6-4).

*Example 6-4 Filter list of running modules for a specific name*

---

```
x36501ab9:~ #lsmod | tee >(head -n 1) >(grep qla) > /dev/null
Module                Size  Used by
qla2xxx                293455  0
scsi_transport_fc      54752  1 qla2xxx
scsi_mod               183796  10 qla2xxx,scsi_transport_fc,scsi_tgt,st,ses, ....
```

---

With the **modinfo** command, you see detailed information about the kernel module itself, such as the version number, what options it supports, and so on. You can see a partial output in Example 6-5.

*Example 6-5 Detailed information about a specific kernel module*

---

```
x36501ab9:~ # modinfo qla2xxx
filename:
/lib/modules/2.6.32.12-0.7-default/kernel/drivers/scsi/qla2xxx/qla2xxx.ko
...
version:      8.03.01.06.11.1-k8
license:      GPL
description:  QLogic Fibre Channel HBA Driver
author:       QLogic Corporation
...
depends:       scsi_mod,scsi_transport_fc
supported:    yes
vermagic:     2.6.32.12-0.7-default SMP mod_unload modversions
parm:         ql2xlogintimeout:Login timeout value in seconds. (int)
parm:         qlport_down_retry:Maximum number of command retries to a port ...
parm:         ql2xplogiabsentdevice:Option to enable PLOGI to devices that ...
...
```

---

**Tip:** The *zfc* driver for Linux on System z automatically scans and registers the attached volumes only in the most recent Linux distributions and only if NPIV is used. Otherwise, you must configure the volumes for access. The reason is that the Linux virtual machine might not use all volumes that are attached to the HBA. See “Linux on System z running in a virtual machine under z/VM” on page 88, and 6.2.7, “Adding DS8000 volumes to Linux on System z” on page 96, for more information.

## Using the FC HBA driver during installation

You can use DS8000 volumes attached to a Linux system during installation. This provides the ability to install all or part of the system to the SAN-attached volumes. The Linux installers detect the FC HBAs, load the necessary kernel modules, scan for volumes, and offer them in the installation options.

When you have an unsupported driver version included with your Linux distribution, either replace it immediately after installation or, if it does not work at all, use a driver disk during the installation. This issue currently exists for Brocade HBAs. A driver disk image is available for download from the Brocade website, see “Loading the Linux Fibre Channel drivers” on page 89.

### Important considerations:

- ▶ Installing a Linux system on a SAN-attached disk does not mean that it will be able to start from it. You need to take additional steps to configure the boot loader or boot program.
- ▶ Take special precautions when configuring multipathing, if you are running Linux on SAN-attached disks.

See 6.5.4, “Booting Linux from DS8000 volumes” on page 116, for details.

## Initial Fibre Channel driver availability

If the SAN-attached DS8000 volumes are needed early in the Linux boot process, for example, if all or part of the system is located on these volumes, it is necessary to include the HBA driver in the *initial RAM filesystem* (initRAMFS) image. With the initRAMFS method, the Linux boot process provides certain system resources before the real system disk is set up.

The Linux distributions contain a script called `mkinitrd` that creates the initRAMFS image automatically. Also, the Linux distributions will automatically include the HBA driver if you already use a SAN-attached disk during installation. If not, you need to include it manually.

**Tip:** The *initRAMFS* was introduced many years ago and replaced the *initial RAM disk* (initrd). The `initrd` reference is still being used; however, `initRAMFS` replaced `initrd`.

## SUSE Linux Enterprise Server

Kernel modules that must be included in the initRAMFS are listed in the `/etc/sysconfig/kernel` file, in the line that starts with `INITRD_MODULES`. The order in which they are shown in this line is the order in which they are loaded at system startup (Example 6-6).

*Example 6-6* Configuring SLES to include a kernel module in the initRAMFS

```
x36501ab9:~ # cat /etc/sysconfig/kernel
...
# This variable contains the list of modules to be added to the initial
```

```
# ramdisk by calling the script "mkinitrd"
# (like drivers for scsi-controllers, for lvm or reiserfs)
#
INITRD_MODULES="thermal aacraid ata_piix ... processor fan jbd ext3 edd qla2xxx"
...
```

---

After adding the HBA driver module name to the configuration file, rebuild the initRAMFS with the **mkinitrd** command. This command creates and installs the image file with standard settings and to standard locations, as shown in Example 6-7.

*Example 6-7 Creating the initRAMFS*

---

```
x3650lab9:~ # mkinitrd

Kernel image:  /boot/vmlinuz-2.6.32.12-0.7-default
Initrd image:  /boot/initrd-2.6.32.12-0.7-default
Root device:   /dev/disk/by-id/scsi-SServeRA_Drive_1_2D0DE908-part1 (/dev/sda1)..
Resume device: /dev/disk/by-id/scsi-SServeRA_Drive_1_2D0DE908-part3 (/dev/sda3)
Kernel Modules: hwmon thermal_sys ... scsi_transport_fc qla2xxx ...
(module qla2xxx.ko firmware /lib/firmware/ql2500_fw.bin) (module qla2xxx.ko ...
Features:      block usb resume.userspace resume.kernel
Bootsplash:    SLES (800x600)
30015 blocks
```

---

If you need nonstandard settings, for example a separate image name, use parameters for the **mkinitrd** command. For additional information, see the man page for the **mkinitrd** command on your Linux system.

### **Red Hat Enterprise Linux**

Kernel modules that must be included in the initRAMFS are listed in the file `/etc/modprobe.conf`. The order that they appear in the file, as shown in Example 6-8, is the order they will be loaded at system startup.

*Example 6-8 Configuring Red Hat Enterprise Linux to include a kernel module in the initRAMFS*

---

```
[root@x3650lab9 ~]# cat /etc/modprobe.conf
alias eth0 bnx2
alias eth1 bnx2
alias eth2 e1000e
alias eth3 e1000e
alias scsi_hostadapter aacraid
alias scsi_hostadapter1 ata_piix
alias scsi_hostadapter2 qla2xxx
alias scsi_hostadapter3 usb-storage
```

---

After adding the HBA driver module to the configuration file, rebuild the initRAMFS with the **mkinitrd** command. The Red Hat version of **mkinitrd** requires parameters, such as the name and location of the image file to create, and the kernel version it is built for, as shown in Example 6-9.

*Example 6-9 Creating the initRAMFS*

---

```
[root@x3650lab9 ~]# mkinitrd /boot/initrd-2.6.18-194.e15.img 2.6.18-194.e15
```

---

If an image file with the specified name already exists, use the **-f** option to force **mkinitrd** to overwrite the existing one. For a more detailed output, use the **-v** option.

Determine the kernel version that is currently running on the system with the `uname` command, as shown in Example 6-10.

*Example 6-10 Determining the Kernel version*

```
[root@x36501ab9 ~]# uname -r
2.6.18-194.el5
```

## 6.2.3 Determining the WWPN of installed HBAs

To create a host connection on the DS8000 that provides the ability to map volumes to a certain HBA, you need the HBA's WWPN. The WWPN, along with additional information about the HBA is provided in the `sysfs` file. The `sysfs` file is a Linux pseudo file system that reflects the installed hardware and its configuration. Example 6-11 shows how to determine which SCSI host instances are assigned to the installed FC HBAs, and their WWPNs.

*Example 6-11 Determining the WWPNs of the FC HBAs*

```
[root@x36501ab9 ~]# ls /sys/class/fc_host/
host3 host4
[root@x36501ab9 ~]# cat /sys/class/fc_host/host3/port_name
0x2100001b32095bad
[root@x36501ab9 ~]# cat /sys/class/fc_host/host4/port_name
0x2100001b320969b6
```

**Tip:** The `sysfs` file contains additional information and is used to modify the hardware configuration. Additional information is provided for `sysfs` in Example 6-40 on page 108.

## 6.2.4 Checking attached volumes

The easiest method to check for DS8000 volumes attached to the Linux system is to use the `lsscsi` command, as shown in Example 6-12.

*Example 6-12 Listing attached SCSI devices*

```
x36501ab9:~ # lsscsi
[0:0:0:0] disk ServeRA Drive 1 V1.0 /dev/sda
[2:0:0:0] cd/dvd MATSHITA UJDA770 DVD/CDRW 1.24 /dev/sr0
[3:0:0:0] disk IBM 2107900 .288 /dev/sdb
[3:0:0:1] disk IBM 2107900 .288 /dev/sdc
[3:0:0:2] disk IBM 2107900 .288 /dev/sdd
[3:0:0:3] disk IBM 2107900 .288 /dev/sde
[3:0:1:0] disk IBM 2107900 .288 /dev/sdf
[3:0:1:1] disk IBM 2107900 .288 /dev/sdg
[3:0:1:2] disk IBM 2107900 .288 /dev/sdh
[3:0:1:3] disk IBM 2107900 .288 /dev/sdi
[4:0:0:0] disk IBM 2107900 .288 /dev/sdj
[4:0:0:1] disk IBM 2107900 .288 /dev/sdk
[4:0:0:2] disk IBM 2107900 .288 /dev/sdl
[4:0:0:3] disk IBM 2107900 .288 /dev/sdm
[4:0:1:0] disk IBM 2107900 .288 /dev/sdn
[4:0:1:1] disk IBM 2107900 .288 /dev/sdo
[4:0:1:2] disk IBM 2107900 .288 /dev/sdp
[4:0:1:3] disk IBM 2107900 .288 /dev/sdq
```

Example 6-12 on page 93 shows that Linux recognized 16 DS8000 devices. By viewing the SCSI addresses in the first column, you can determine that there actually are four DS8000 volumes, each connected through four paths. Linux creates a SCSI disk device for each of the paths.

**Red Hat Enterprise Linux installer:** The Red Hat Enterprise Linux installer does not install `lsscsi` by default. It is shipped with the distribution, but must be selected for installation.

## 6.2.5 Linux SCSI addressing

The quadruple in the first column of the `lsscsi` output is the internal Linux SCSI address. It is, for historical reasons, constructed like a traditional parallel SCSI address. It consists of four fields:

- ▶ HBA ID. Each HBA in the system, whether parallel SCSI, FC, or even a SCSI emulator, gets a host adapter instance when it is initiated.
- ▶ Channel ID. This ID is always zero. It was formerly used as an identifier for the channel in multiplexed parallel SCSI HBAs.
- ▶ Target ID. For parallel SCSI, it is the real target ID; the one set through a jumper on the disk drive. For Fibre Channel, it represents a remote port that is connected to the HBA. With it, you can distinguish between multiple paths and between multiple storage systems.
- ▶ LUN. This item is rarely used in parallel SCSI. In Fibre Channel, it is used to represent a single volume that a storage system offers to the host. The LUN is assigned by the storage system.

Figure 6-1 illustrates how the SCSI addresses are generated.

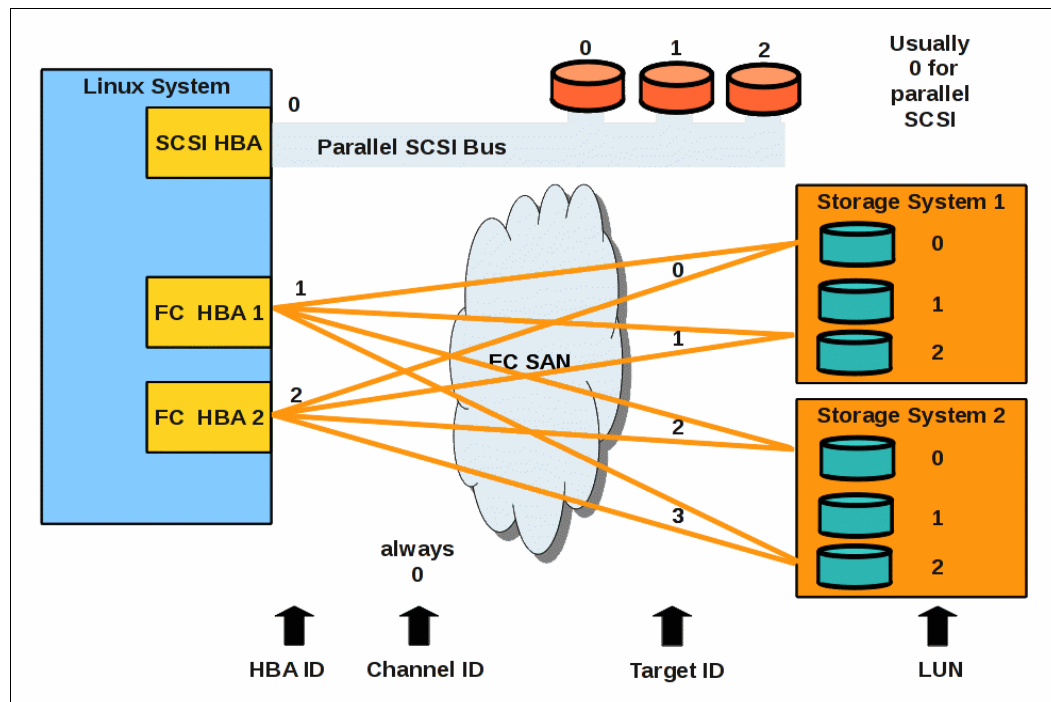


Figure 6-1 Understanding the composition of Linux internal SCSI addresses

## 6.2.6 Identifying DS8000 devices

The udev subsystem creates device nodes for all attached devices. In the case of disk drives, it not only sets up the traditional `/dev/sdx` nodes, but also other representatives, such as the ones that can be found in `/dev/disk/by-id` and `/dev/disk/by-path`.

The nodes for DS8000 volumes in `/dev/disk/by-id` provide a unique identifier that is composed of the WWNN of the DS8000 system and the DS8000 volume ID, as shown in Example 6-13.

*Example 6-13 The `/dev/disk/by-id` device nodes*

---

```
x36501ab9:~ # ls -l /dev/disk/by-id/ | cut -c 44-
...
scsi-3600507630affc29f0000000000004690 -> ../../sdj
scsi-3600507630affc29f0000000000004691 -> ../../sdk
scsi-3600507630affc29f0000000000004790 -> ../../sdl
scsi-3600507630affc29f0000000000004791 -> ../../sdq
...
```

---

The udev subsystem already recognizes that more than one path exists to each DS8000 volume. It creates only one node for each volume, instead of four.

**Important:** The device nodes in `/dev/disk/by-id` are persistent, whereas the `/dev/sdx` nodes are not. They can change, when the hardware configuration changes. Do not use `/dev/sdx` device nodes to mount file systems or specify system disks.

In `/dev/disk/by-path`, there are nodes for all paths to all DS8000 volumes. Here, you can see the physical connection to the volumes, starting with the Peripheral Component Interconnect (PCI) identifier of the HBAs, through the remote port represented by the DS8000 WWPN, to the LUN of the volumes, as shown in Example 6-14.

*Example 6-14 The `/dev/disk/by-path` device nodes*

---

```
x36501ab9:~ # ls -l /dev/disk/by-path/ | cut -c 44-
...
pci-0000:1c:00.0-fc-0x500507630a00029f:0x0000000000000000 -> ../../sdb
pci-0000:1c:00.0-fc-0x500507630a00029f:0x0001000000000000 -> ../../sdc
pci-0000:1c:00.0-fc-0x500507630a00029f:0x0002000000000000 -> ../../sdd
pci-0000:1c:00.0-fc-0x500507630a00029f:0x0003000000000000 -> ../../sde
pci-0000:1c:00.0-fc-0x500507630a0b029f:0x0000000000000000 -> ../../sdf
pci-0000:1c:00.0-fc-0x500507630a0b029f:0x0001000000000000 -> ../../sdg
pci-0000:1c:00.0-fc-0x500507630a0b029f:0x0002000000000000 -> ../../sdh
pci-0000:1c:00.0-fc-0x500507630a0b029f:0x0003000000000000 -> ../../sdi
pci-0000:24:00.0-fc-0x500507630a13029f:0x0000000000000000 -> ../../sdj
pci-0000:24:00.0-fc-0x500507630a13029f:0x0001000000000000 -> ../../sdk
pci-0000:24:00.0-fc-0x500507630a13029f:0x0002000000000000 -> ../../sdl
pci-0000:24:00.0-fc-0x500507630a13029f:0x0003000000000000 -> ../../sdm
pci-0000:24:00.0-fc-0x500507630a1b029f:0x0000000000000000 -> ../../sdn
pci-0000:24:00.0-fc-0x500507630a1b029f:0x0001000000000000 -> ../../sdo
pci-0000:24:00.0-fc-0x500507630a1b029f:0x0002000000000000 -> ../../sdp
pci-0000:24:00.0-fc-0x500507630a1b029f:0x0003000000000000 -> ../../sdq
```

---

## 6.2.7 Adding DS8000 volumes to Linux on System z

Only in the most recent Linux distributions for System z does the `zfc` driver automatically scan for connected volumes. This section provides information about how to configure the system, so that the driver automatically makes specified volumes available when it starts. Volumes and their path information, the local HBA and DS8000 ports, are defined in configuration files.

**Tip:** SLES10 SP3 was used for the production of the examples in this section. Procedures, commands and configuration files of other distributions can differ.

Linux on System z was used to produce Example 6-15 with two FC HBAs assigned through `z/VM`. In `z/VM` you can determine the device numbers of these adapters.

*Example 6-15 FCP HBA device numbers in z/VM*

---

```
#CP QUERY VIRTUAL FCP
FCP 0500 ON FCP 5A00 CHPID 8A SUBCHANNEL = 0000
...
FCP 0600 ON FCP 5B00 CHPID 91 SUBCHANNEL = 0001
...
```

---

The Linux on System z tool to list the FC HBAs is `lszfc`. It shows the enabled adapters only. Adapters that are not listed correctly can be enabled using the `chccwdev` command, as shown in Example 6-16.

*Example 6-16 List and enable the Linux on System z FCP adapters*

---

```
lnxmnt01:~ # lszfc
0.0.0500 host0

lnxmnt01:~ # chccwdev -e 600
Setting device 0.0.0600 online
Done

lnxmnt01:~ # lszfc
0.0.0500 host0
0.0.0600 host1
```

---

For SUSE Linux Enterprise Server 10, the volume configuration files reside in the `/etc/sysconfig/hardware` directory. There must be one configuration file for each HBA. Example 6-17 provides their naming scheme.

*Example 6-17 HBA configuration files*

---

```
lnxmnt01:~ # ls /etc/sysconfig/hardware/ | grep zfc
hwcfg-zfc-bus-ccw-0.0.0500
hwcfg-zfc-bus-ccw-0.0.0600
```

---

**Using YAST:** The type of configuration file described in Example 6-17 are based on SLES9 and SLES10. SLES11 uses `udev` rules, that are automatically created by the *Yet Another Setup Tool* (YAST) when you use it to determine and configure SAN-attached volumes. The `udev` rules can be complicated and are not well documented. With SLES11, use YAST for easier configurations.



The configuration files contain a remote DS8000 port, a LUN pair for each path to each volume. Example 6-18 demonstrates two DS8000 volumes to the HBA 0.0.0500, going through two separate DS8000 host ports.

*Example 6-18 HBA configuration file*

---

```
lnxmnt01:~ # cat /etc/sysconfig/hardware/hwcfg-zfcp-bus-ccw-0.0.0500
#!/bin/sh
#
# hwcfg-zfcp-bus-ccw-0.0.0500
#
# Configuration for the zfcp adapter at CCW ID 0.0.0500
#
...
# Configured zfcp disks
ZFCP_LUNS="
0x500507630a00029f:0x4049400000000000
0x500507630a00029f:0x4049400100000000
0x500507630a0b029f:0x4049400000000000
0x500507630a0b029f:0x4049400100000000"

```

---

The `ZFCP_LUNS="..."` statement in the file defines the remote port volume relations or paths that the `zfcp` driver sets up when it starts. The first term in each pair is the WWPN of the DS8000 host port, the second term, after the colon, is the LUN of the DS8000 volume.

**Tip:** The LUN provided in Example 6-18 is the same LUN in the DS8000 LUN map, as shown in Example 6-19. This LUN is padded with zeroes, such that it reaches a length of eight bytes.

DS8000 LUNs for the host type zLinux are created from the DS8000 volume ID, as demonstrated in Example 6-19.

*Example 6-19 DS8000 LUN format for zLinux*

---

```
dscli> showvolgrp -lunmap V18
Name ITS0_zLinux
ID V18
Type SCSI Mask
Vols 4800 4900 4901 4902
=====LUN Mapping=====
vol lun
=====
4800 40484000
4900 40494000
4901 40494001
4902 40494002

```

---

**Using the `lsscsi` command:** The `lsscsi` command produces odd LUNs for DS8000 volumes in Linux on System z. These LUNs are decimal representations of word-swapped DS8000 LUNs. For example, the `lsscsi` command shows LUN **1073758281**, which is **0x40004049** hexadecimal and corresponds to DS8000 LUN **0x40494000**.

Red Hat Enterprise Linux uses the `/etc/zfcp.conf` file to configure SAN-attached volumes. The format of this file is shown in Example 6-20. The three bottom lines in this example are comments that explain the format. Comments are not required in the file.

Example 6-20 Format of the `/etc/zfcp.conf` file for Red Hat Enterprise Linux

```

lrxmnt01:~ # cat /etc/zfcp.conf
0x0500 0x500507630a00029f 0x4049400000000000
0x0500 0x500507630a00029f 0x4049400100000000
0x0500 0x500507630a0b029f 0x4049400000000000
0x0500 0x500507630a0b029f 0x4049400100000000
0x0600 0x500507630a13029f 0x4049400000000000
0x0600 0x500507630a13029f 0x4049400100000000
0x0600 0x500507630a1b029f 0x4049400000000000
0x0600 0x500507630a1b029f 0x4049400100000000
# |
#FCP HBA | LUN
# Remote (DS8000) Port

```

## 6.2.8 Setting up device mapper multipathing

To gain redundancy and optimize performance, connect a server to a storage system through more than one HBA, fabric and storage port. This server connection results in multiple paths from the server to each attached volume. Linux detects such volumes more than once and creates a device node for every instance. To use the path redundancy and increased I/O bandwidth, and at the same time maintain data integrity, you need an additional layer in the Linux storage stack to recombine the multiple disk instances into one device.

Linux provides its own native multipathing solution. It is based on the *device mapper*, a block device virtualization layer in the Linux kernel. Therefore it is called DM Multipath. Figure 6-2 illustrates how DM Multipath is integrated into the Linux storage stack.

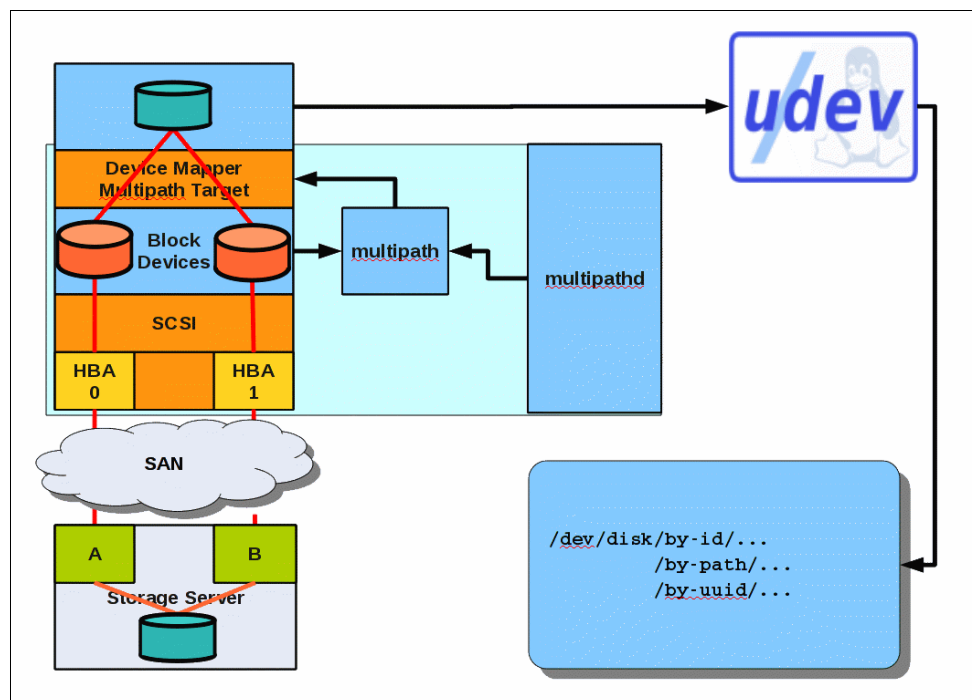


Figure 6-2 Device mapper multipathing in the Linux storage stack

The device mapper is also used for other virtualization tasks, such as the logical volume manager, data encryption, snapshots, and software RAID. DM Multipath is able to manage path failover, failback, and load balancing for various separate storage architectures. In simplified terms, DM Multipath consists of four main components:

- ▶ The **dm-multipath** kernel module takes the I/O requests that go to the multipath device and passes them to the individual devices representing the paths.
- ▶ The **multipath** tool scans the device or path configuration and builds the instructions for the device mapper. These include the composition of the multipath devices, failover, failback patterns, and load balancing behavior. Future functionality of the tool will be moved to the multipath background daemon and will not be available in future releases.
- ▶ The multipath background daemon, **multipathd**, constantly monitors the state of the multipath devices and the paths. In case of events, it triggers failover and failback activities in the DM-Multipath module. It also provides a user interface for online reconfiguration of the multipathing. In the future, it will take over all configuration and setup tasks.
- ▶ A set of rules that tell udev what device nodes to create, so that multipath devices can be accessed and are persistent.

## Configuring DM Multipath

Use the `/etc/multipath.conf` file to configure DM Multipath to:

- ▶ Define new storage device types
- ▶ Exclude certain devices or device types
- ▶ Set names for multipath devices
- ▶ Change error recovery behavior

For more information about `/etc/multipath.conf`, see 6.1.2, “Attaching Linux server to DS8000 resources” on page 82. In 6.2.9, “Attaching DS8000: Considerations” on page 104 also provides settings that are preferred specifically for DS8000 attachment.

You can configure DM Multipath to generate user friendly device names by specifying this option in `/etc/multipath.conf`, as shown in Example 6-21.

*Example 6-21 Specify user friendly names in `/etc/multipath.conf`*

---

```
defaults {  
    ...  
    user_friendly_names yes  
    ...  
}
```

---

The names generated using this method are persistent, meaning that they do not change, even if the device configuration changes. If a volume is removed, its former DM Multipath name will not be used again for a new one. If it is re-attached, it will get its old name. The mappings between unique device identifiers and DM Multipath user friendly names are stored in the file `/var/lib/multipath/bindings`.

**User friendly names:** The user friendly names are separate for SLES 11 and Red Hat Enterprise Linux 5. No new changes were found for the user friendly set up between Red Hat Enterprise Linux 5 and 6, however there are some minor difference for enabling multipathing. For more information about working with user friendly names, see “Accessing DM Multipath devices in SLES 11” on page 102 and “Accessing DM Multipath devices in Red Hat Enterprise Linux 5” on page 103.

## Enabling multipathing for SLES 11

You can start device mapper multipathing by running two start scripts that have already been prepared, as shown in Example 6-22.

### Example 6-22 Starting DM Multipath in SLES 11

---

```
x3650lab9:~ # /etc/init.d/boot.multipath start
Creating multipath target                               done
x3650lab9:~ # /etc/init.d/multipathd start
Starting multipathd                                   done
```

---

In order to have DM Multipath start automatically at each system start, add the start scripts to the SLES 11 system start process, as shown in Example 6-23.

### Example 6-23 Configuring automatic start of DM Multipath in SLES 11

---

```
x3650lab9:~ # inserv boot.multipath
x3650lab9:~ # inserv multipathd
```

---

## Enabling multipathing for Red Hat Enterprise Linux 5

Red Hat Enterprise Linux comes with the `/etc/multipath.conf` default file. This file contains a section that blacklists all device types. To start DM Multipath, remove or comment out these lines.

A pound (#) sign in front of the comment lines will mark them as comments, so they will be ignored the next time DM Multipath scans for devices, as shown in Example 6-24.

### Example 6-24 Disabling blacklisting all devices in `/etc/multipath.conf`

---

```
...
# Blacklist all devices by default. Remove this to enable multipathing
# on the default devices.
#blacklist {
#devnode "*"
#}
...
```

---

Start DM Multipath, as shown in Example 6-25.

### Example 6-25 Starting DM Multipath in Red Hat Enterprise Linux 5

---

```
[root@x3650lab9 ~]# /etc/init.d/multipathd start
Starting multipathd daemon:                               [ OK ]
```

---

In order to have DM Multipath start automatically at each system start, add the start script to the Red Hat Enterprise Linux 5 system start process, as shown in Example 6-26.

### Example 6-26 Configuring automatic start of DM Multipath in Red Hat Enterprise Linux 5

---

```
[root@x3650lab9 ~]# chkconfig --add multipathd
[root@x3650lab9 ~]# chkconfig --levels 35 multipathd on
[root@x3650lab9 ~]# chkconfig --list multipathd
multipathd      0:off  1:off  2:off  3:on   4:off  5:on   6:off
```

---

## Enabling multipathing for Red Hat Enterprise Linux 6

There are some new features that are related to the DS8000 storage in Red Hat 6. There are no major differences in setting up the Red Hat 6 from the set up outline in 5. However, a good practice for this setup is to follow Chapter 2, “Storage Consideration,” in the documentation from the official Red Hat 6 website:

[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/pdf/Storage\\_Administration\\_Guide/Red\\_Hat\\_Enterprise\\_Linux-6-Storage\\_Administration\\_Guide-en-US.pdf](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/pdf/Storage_Administration_Guide/Red_Hat_Enterprise_Linux-6-Storage_Administration_Guide-en-US.pdf)

It is important to have previously installed the device-mapper-multipath package, 32 or 64 version, from the Red Hat Enterprise 6 CD. Create the storage volumes and volume groups, and create the multipath.conf file as stated in the Storage Administration Guide mentioned.

**Tip:** When you create the multipath.conf file, under the /etc directory, make sure to copy and paste the script from the RHEL 6 Storage Administration Guide without spaces, because it can give a keyboard error when starting the multipath server. See Figure 6-2 on page 98 for an example of a multipath set up.

Red Hat 6 has new features that can improve workloads with the DS8000 storage units:

- ▶ **File System Caching:** This feature allows the use of local storage data for cash over the network which can lead to decreased traffic network.
- ▶ **I/O Limit Processing:** This new feature allows I/O optimization for the devices that provide this feature.
- ▶ **exT4 Support:** Red Hat now supports an unlimited number of subdirectories, granular time stamping, journeying quota, extended attribute support and other small features. You can use this new feature for creating the file path on the system during the installation because it is now supported.
- ▶ **Network Block Storage:** Now Red Hat 6 has support for Fibre Channel over Ethernet for 10 Gb/s usage with the same protocols.

To review all the Red Hat 6 features and how to configure the ones listed before, see the *Red Hat Storage Administration Manual*:

[http://docs.redhat.com/docs/en-US/Red\\_Hat\\_Enterprise\\_Linux/6/pdf/Storage\\_Administration\\_Guide/Red\\_Hat\\_Enterprise\\_Linux-6-Storage\\_Administration\\_Guide-en-US.pdf](http://docs.redhat.com/docs/en-US/Red_Hat_Enterprise_Linux/6/pdf/Storage_Administration_Guide/Red_Hat_Enterprise_Linux-6-Storage_Administration_Guide-en-US.pdf)

## Working with the DM Multipath configuration

The multipath background daemon provides a user interface to print and modify the DM Multipath configuration. It can be started as an interactive session with the `multipathd -k` command. Within this session, a variety of options are available. Use the `help` command to get a list of available options. The most significant options are demonstrated in the following examples, and in 6.2.10, “Understanding non-disruptive actions on attached hosts” on page 106.

The `show topology` command prints out a detailed view of the current DM Multipath configuration, including the state of all available paths, as shown in Example 6-27.

*Example 6-27 Show multipath topology*

```
x3650lab9:~ # multipathd -k
multipathd> show topology
3600507630affc29f0000000000004690 dm-0 IBM,2107900
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
   |- 3:0:0:0 sdb 8:16 active ready running
```

```

|- 3:0:1:0 sdf 8:80 active ready running
|- 4:0:0:0 sdj 8:144 active ready running
~- 4:0:1:0 sdn 8:208 active ready running
3600507630affc29f0000000000004790 dm-1 IBM,2107900
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
~+~ policy='round-robin 0' prio=1 status=active
|- 3:0:0:2 sdd 8:48 active ready running
|- 3:0:1:2 sdh 8:112 active ready running
|- 4:0:0:2 sdl 8:176 active ready running
~- 4:0:1:2 sdp 8:240 active ready running
3600507630affc29f0000000000004791 dm-2 IBM,2107900
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
~+~ policy='round-robin 0' prio=1 status=active
|- 3:0:0:3 sde 8:64 active ready running
|- 3:0:1:3 sdi 8:128 active ready running
|- 4:0:0:3 sdm 8:192 active ready running
~- 4:0:1:3 sdq 65:0 active ready running
3600507630affc29f0000000000004691 dm-3 IBM,2107900
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
~+~ policy='round-robin 0' prio=1 status=active
|- 3:0:1:1 sdg 8:96 active ready running
|- 3:0:0:1 sdc 8:32 active ready running
|- 4:0:0:1 sdk 8:160 active ready running
~- 4:0:1:1 sdo 8:224 active ready running

```

---

Use **reconfigure**, as shown in Example 6-28, to configure DM Multipath to update the topology after scanning the paths and configuration files. You can also use it to add new multipath devices after adding new DS8000 volumes. For more information, see “Adding and removing DS8000 volumes dynamically on Linux” on page 106.

*Example 6-28 Reconfiguring DM Multipath*

---

```

multipathd> reconfigure
ok

```

---

**Exit:** The **multipathd -k** command prompt of SLES11 SP1 supports the **quit** and **exit** commands to terminate. Press CTRL+D to terminate or quit the session in Red Hat Enterprise Linux 5U5.

**Fast path:** You can also issue commands in a *one-shot-mode* by enclosing them in double quotes and typing them directly, without space, behind the **multipath -k** command, for example, **multipathd -k"show paths"**.

**Print:** Although the **multipath -l** and **multipath -ll** commands can be used to print the current DM Multipath configuration, use the **multipathd -k** interface. The **multipath** tool will be removed from DM Multipath and all further development and improvements go into **multipathd**.

## Accessing DM Multipath devices in SLES 11

The device nodes used to access DM Multipath devices are created by udev in the **/dev/mapper** directory. If you do not change any settings, SLES 11 uses the unique identifier of a volume as device name, as shown in Example 6-29.

*Example 6-29 Multipath devices in SLES 11 in /dev/mapper*

---

```
x36501ab9:~ # ls -l /dev/mapper | cut -c 48-
```

```
3600507630affc29f0000000000004690
3600507630affc29f0000000000004691
3600507630affc29f0000000000004790
3600507630affc29f0000000000004791
```

---

**Important:** The device mapper itself creates its default device nodes in the /dev directory. These nodes are named /dev/dm-0, /dev/dm-1, and so on. However, these nodes are not persistent, can change with configuration changes, and cannot be used for device access.

SLES 11 creates an additional set of device nodes for multipath devices. It overlays the former single path device nodes in the /dev/disk/by-id directory. It means that any device mapping, for example mounting a file system, configured with one of these nodes works exactly the same as before starting DM Multipath. Instead of using the SCSI disk device to start, it uses the DM Multipath device, as shown in Example 6-30.

*Example 6-30 SLES 11 DM Multipath device nodes in /dev/disk/by-id*

---

```
x36501ab9:~ # ls -l /dev/disk/by-id/ | cut -c 44-
```

```
...
scsi-3600507630affc29f0000000000004690 -> ../../dm-0
scsi-3600507630affc29f0000000000004691 -> ../../dm-3
scsi-3600507630affc29f0000000000004790 -> ../../dm-1
scsi-3600507630affc29f0000000000004791 -> ../../dm-2
...
```

---

If you set the `user_friendly_names` option in the /etc/multipath.conf file, SLES 11 will create DM Multipath devices with the names `mpatha`, `mpathb`, and so on, in the /dev/mapper directory. The DM Multipath device nodes in the /dev/disk/by-id directory are not changed. These nodes still exist and have the volumes' unique IDs in their names.

## Accessing DM Multipath devices in Red Hat Enterprise Linux 5

Red Hat Enterprise Linux sets the `user_friendly_names` option in its default /etc/multipath.conf file. The devices it creates in the /dev/mapper directory is shown in Example 6-31.

*Example 6-31 Multipath devices in Red Hat Enterprise Linux 5 in /dev/mapper*

---

```
[root@x36501ab9 ~]# ls -l /dev/mapper/ | cut -c 45-
```

```
mpath1
mpath2
mpath3
mpath4
```

---

A second set of device nodes contain the unique IDs of the volumes in their name, whether user friendly names are specified or not. These nodes are in the /dev/mpath directory.

## Using multipath devices

Use the device nodes that are created for multipath devices just like any other block device:

- ▶ Create a file system and mount it
- ▶ Use LVM on the device nodes
- ▶ Build software RAID devices

You can also partition a DM Multipath device using the `fdisk` command, or any other partitioning tool.

To make new partitions on DM Multipath devices available, use the `partprobe` command. It triggers udev to set up new block device nodes for the partitions, as shown in Example 6-32.

*Example 6-32 Using the `partprobe` command to register newly created partitions*

---

```
x36501ab9:~ # fdisk /dev/mapper/3600507630affc29f000000000004691
...
<all steps to create a partition and write the new partition table>
...
x36501ab9:~ # ls -l /dev/mapper/ | cut -c 48-
3600507630affc29f000000000004690
3600507630affc29f000000000004691
3600507630affc29f000000000004790
3600507630affc29f000000000004791
x36501ab9:~ # partprobe
x36501ab9:~ # ls -l /dev/mapper/ | cut -c 48-
3600507630affc29f000000000004690
3600507630affc29f000000000004691
3600507630affc29f000000000004691_part1
3600507630affc29f000000000004790
3600507630affc29f000000000004791
```

---

Example 6-32 was created with SLESS 11. The method works as well for Red Hat Enterprise Linux 5, but the partition names might be separate.

**Older Linux versions:** One of the issues with older Linux versions is that LVM, by default, will not work with DM Multipath devices; it does not exist in recent Linux versions.

## 6.2.9 Attaching DS8000: Considerations

This section provides additional considerations that specifically apply to the DS8000.

### Configuring DM multipath for DS8000

IBM provides a `multipath.conf` file that contains the preferred settings for DS8000 attachment at the following website:

<http://www.ibm.com/support/docview.wss?rs=540&context=ST52G7&dc=D430&uid=ssg1S4000107#DM>

The relevant settings for DS8000 are shown in Example 6-33.

*Example 6-33 DM Multipath settings for DS8000*

---

```
defaults {
    polling_interval    30
    failback            immediate
    no_path_retry       5
    path_checker        tur
    user_friendly_names yes
    rr_min_io           100
}
devices {
```



```

device {
    vendor          "IBM"
    product         "2107900"
    path_grouping_policy  group_by_serial
}

```

**User friendly setting:** Settings for the `user_friendly_names` parameter are provided in Example 6-21 on page 99. You can leave it as it is or remove it from the file.

The values for `polling_interval`, `failback`, `no_path_retry`, and `path_checker`, control the behavior of DM Multipath in case of path failures. For most instances, these values should not be changed. If you need to modify these parameters, see 6.1.2, “Attaching Linux server to DS8000 resources” on page 82, for more information.

The `rr_min_io` setting specifies the number of I/O requests that are sent to one path before switching to the next one. In certain cases, you can improve performance by lowering this value from 100 to 32.

**Important:** In SLES 11 SP1, `rr_min_io` does not work in the `defaults` section. Move it to the `devices` section, for example:

```

...
devices {
    device {
        vendor          "IBM"
        product         "2107900"
        path_grouping_policy  group_by_serial
        rr_min_io       32
    }
}
...

```

## System z multipath settings

For Linux on System z multipathing, set the `dev_loss_tmo` parameter to 90 seconds and the `fast_io_fail_tmo` parameter to 5 seconds.

Modify the `/etc/multipath.conf` file to specify the values, as shown in Example 6-34.

*Example 6-34 System z multipath settings*

```

...
defaults {
...
    dev_loss_tmo      90
    fast_io_fail_tmo  5
...
}
...

```

Make the changes effective by using the `reconfigure` command in the interactive `multipathd -k` prompt.

## Disabling QLogic failover

The QLogic HBA kernel modules have limited built in multipathing capabilities. Because multipathing is managed by DM Multipath, ensure that the QLogic failover support is disabled. Use the `modinfo qla2xxx` command to verify, as shown in Example 6-35.

*Example 6-35 Verify QLogic failover is disabled*

---

```
x36501ab9:~ # modinfo qla2xxx | grep version
version:      8.03.01.04.05.05-k
srcversion:   A2023F2884100228981F34F
```

---

If the version string ends with `-fo`, the failover capabilities are turned on and must be disabled. To do this, add a line to the `/etc/modprobe.conf` file of your Linux system, as shown in Example 6-36.

*Example 6-36 Disabling QLogic failover*

---

```
x36501ab9:~ # cat /etc/modprobe.conf
...
options qla2xxx ql2xfailover=0
...
```

---

After modifying this file, run the `depmod -a` command to refresh the kernel driver dependencies. Then, reload the `qla2xxx` module to make the change effective. If you have included the `qla2xxx` module in the `InitRAMFS`, you must create a new `qla2xxx` module.

## 6.2.10 Understanding non-disruptive actions on attached hosts

This section describes some non-disruptive actions that can be taken on attached hosts:

- ▶ Adding and removing DS8000 volumes dynamically on Linux
- ▶ Adding and removing DS8000 volumes in Linux on System z
- ▶ Adding new DS8000 host ports to Linux on System z
- ▶ Resizing DS8000 volumes dynamically
- ▶ Using FlashCopy and remote replication targets

### Adding and removing DS8000 volumes dynamically on Linux

Unloading and reloading the Fibre Channel HBA adapter was the typical method to discover newly attached DS8000 volumes. However, this action is disruptive to all applications that use Fibre Channel-attached disks on this particular host.

With recent Linux versions, you can add newly attached LUNs without unloading the FC HBA driver. To do this, obtain the SCSI instances that your FC HBAs have, then scan for new Fibre Channel attached devices by using the command `syfs`, as shown in Example 6-37.

*Example 6-37 Scanning for new Fibre Channel attached devices*

---

```
x36501ab9:~ # ls /sys/class/fc_host/
host3 host4
x36501ab9:~ # echo "- - -" > /sys/class/scsi_host/host3/scan
x36501ab9:~ # echo "- - -" > /sys/class/scsi_host/host4/scan
```

---

The triple dashes “- - -” represent the channel, target, and LUN combination to scan. A dash causes a scan through all possible values. A number will limit the scan to the given value. New disk devices that are discovered with this method, automatically get device nodes, and are added to DM Multipath.

**Tip:** For certain older Linux versions, you must force the FC HBA to perform a port login to recognize newly added devices, by using the following command:

```
echo 1 > /sys/class/fc_host/host<ID>/issue_lip
```

You need to run this command for all FC HBAs.

## Removing a disk device or volume from Linux

To remove a disk device or volume from Linux, and avoid system suspending due to incomplete I/O requests, follow this sequence:

1. Stop all applications that use the device and ensure all updates or writes are completed.
2. Un-mount the file systems that use the device.
3. If the device is part of an LVM configuration, remove it from all logical volumes and volume groups.
4. Remove all paths to the device from the system, as shown in Example 6-38.

*Example 6-38 Removing all paths to a disk device*

---

```
x36501ab9:~ # echo 1 > /sys/class/scsi_disk/3\0\0\3/device/delete
x36501ab9:~ # echo 1 > /sys/class/scsi_disk/3\0\1\3/device/delete
x36501ab9:~ # echo 1 > /sys/class/scsi_disk/4\0\0\3/device/delete
x36501ab9:~ # echo 1 > /sys/class/scsi_disk/4\0\1\3/device/delete
```

---

The device paths or disk devices are represented by their Linux SCSI address, see 6.2.5, “Linux SCSI addressing” on page 94. Run the `multipathd -k"show topology"` command after removing each path to monitor the progress.

DM Multipath and udev recognize the removal automatically and delete all corresponding disk and multipath device nodes. Make sure you remove all paths that exist to the device. After all paths are removed, you can detach the device on the storage system level.

**The watch command:** Run the `watch` command periodically to monitor the multipath topology. To monitor the multipath topology with a period of one second, see the following command:

```
watch -n 1 'multipathd -k"show top"'
```

## Adding and removing DS8000 volumes in Linux on System z

The mechanisms to scan and attach new volumes provided in “Adding and removing DS8000 volumes dynamically on Linux” on page 106, do not work in the same manner for Linux on System z. Commands are available that discover and show the devices connected to the FC HBAs, but they do not process the logical attachment to the operating system automatically. In SLES10 SP3, use the `zfcpsan_disc` command for discovery.

Example 6-39 shows how to discover and list the connected volumes for one remote port or path, using the `zfcpsan_disc` command. Run this command for all available remote ports.

*Example 6-39 List LUNs connected through a specific remote port*

---

```
lnxmnt01:~ # zfcpsan_disc -L -p 0x500507630a00029f -b 0.0.0500
0x4048400000000000
0x4049400000000000
0x4049400100000000
0x4049400200000000
```

---

**Tip:** In recent Linux distributions, the `zfcpsan_disc` command is not available. Remote ports are automatically discovered. The attached volumes can be listed using the `ls1uns` script.

After discovering the connected volumes, do the logical attachment using `sysfs` interfaces. Remote ports or device paths are represented in `sysfs`.

There is a directory for each local or remote port combination, or path. This directory contains a representative of each attached volume and various meta files, as interfaces for action.

Example 6-40 shows a `sysfs` structure for a specific DS8000 remote port.

*Example 6-40 sysfs structure for a remote port*

```
lnxmnt01:~ # ls -l /sys/bus/ccw/devices/0.0.0500/0x500507630a00029f/
total 0
drwxr-xr-x 2 root root    0 2010-11-18 11:03 0x4049400000000000
...
--w----- 1 root root    0 2010-11-18 17:27 unit_add
--w----- 1 root root    0 2010-11-18 17:55 unit_remove
```

In Example 6-41, LUN 0x4049400100000000 is added to all available paths, using the `unit_add` metafile.

*Example 6-41 Adding a volume to all existing remote ports*

```
lnxmnt01:~ # echo 0x4049400100000000 > /sys/.../0.0.0500/0x500507630a00029f/unit_add
lnxmnt01:~ # echo 0x4049400100000000 > /sys/.../0.0.0500/0x500507630a0b029f/unit_add
lnxmnt01:~ # echo 0x4049400100000000 > /sys/.../0.0.0600/0x500507630a03029f/unit_add
lnxmnt01:~ # echo 0x4049400100000000 > /sys/.../0.0.0600/0x500507630a08029f/unit_add
```

**Important:** Perform discovery, using `zfcpsan_disc`, when new devices, such as remote ports or volumes are attached. Otherwise, the system does not recognize them, even if you do the logical configuration.

New disk devices attached in this method will automatically get device nodes and are added to DM Multipath.

## Removing a volume from Linux on System z

To remove a volume from Linux on System z and avoid system suspending due to incomplete I/O requests, follow the same sequence used in “Removing a disk device or volume from Linux” on page 107. Volumes can then be removed logically, using a method similar to the attachment. Write the LUN of the volume into the `unit_remove` metafile for each remote port in `sysfs`.

**Tip:** For newly added devices to be persistent, use the methods provided in 6.2.7, “Adding DS8000 volumes to Linux on System z” on page 96, to create the configuration files to be used at the next system start.

### 6.2.11 Adding new DS8000 host ports to Linux on System z

If you connect new DS8000 ports or a new DS8000 system to Linux on System z, complete the following steps to logically attach the new remote ports:

1. Discover DS8000 ports that are connected to the HBAs, as shown in Example 6-42.

*Example 6-42 Discovering connected remote ports*

---

```
lnxmnt01:~ # zfcplib_san_disc -W -b 0.0.0500
0x500507630a00029f
0x500507630a0b029f
lnxmnt01:~ # zfcplib_san_disc -W -b 0.0.0600
0x500507630a03029f
0x500507630a08029f
```

---

2. Attach the DS8000 ports logically to the HBAs. As Example 6-43 shows, there already is a remote port attached to HBA **0.0.0500**. It is the one path already available to access the DS8000 volume.

*Example 6-43 Listing and attaching remote ports*

---

```
lnxmnt01:~ # ls /sys/bus/ccw/devices/0.0.0500/ | grep 0x
0x500507630a00029f
```

```
lnxmnt01:~ # echo 0x500507630a0b029f > /sys/bus/ccw/devices/0.0.0500/port_add
```

```
lnxmnt01:~ # ls /sys/bus/ccw/devices/0.0.0500/ | grep 0x
0x500507630a00029f
0x500507630a0b029f
```

---

3. Add the second connected DS8000 port to the HBA. Example 6-44 shows how to add two DS8000 ports to the second HBA.

*Example 6-44 Attaching remote ports to the second HBA*

---

```
lnxmnt01:~ # echo 0x500507630a08029f > /sys/bus/ccw/devices/0.0.0600/port_add
lnxmnt01:~ # echo 0x500507630a03029f > /sys/bus/ccw/devices/0.0.0600/port_add
lnxmnt01:~ # ls /sys/bus/ccw/devices/0.0.0600/ | grep 0x
0x500507630a03029f
0x500507630a08029f
```

---

## 6.3 Resizing DS8000 volumes dynamically

Currently, only SLES11 SP1 and RHEL 6 are capable of utilizing the additional capacity of dynamically enlarged DS8000 volumes. Reducing the size is not supported for versions before RHEL 5.

To resize DS8000 volumes dynamically, complete the following steps:

1. Create an **ext3** file system on one of the DS8000 multipath devices and mount it.

The **df** command demonstrated in Example 6-45 shows the available capacity.

*Example 6-45 Checking the size and available space on a mounted file system*

---

```
x3650lab9:~ # df -h /mnt/itso_4791
Filesystem      Size Used Avail Use% Mounted on
/dev/mapper/3600507630affc29f0000000000004791
                50G  180M  47G   1% /mnt/itso_4791
```

---

2. Use the **chfbvol** command of the DSCLI to increase the capacity of the volume from 50 to 100 GB, as shown in Example 6-46.

*Example 6-46 Increasing the size of a DS8000 volume*

---

```
dscli> chfbvol -cap 100 4791
Date/Time: October 12, 2010 4:48:59 PM CEST IBM DSCLI Version: ...
CMUC00332W chfbvol: Some host operating systems do not support changing the
volume size. Are you sure that you want to resize the volume? [y/n]: y
CMUC00026I chfbvol: FB volume 4791 successfully modified.
```

---

3. Rescan all disk devices or paths of a DS8000 volume. The Linux SCSI layer picks up the new capacity when a rescan is initiated of each SCSI disk device or path through `sysfs`, as shown in Example 6-47.

*Example 6-47 Rescan all disk devices or paths of a DS8000 volume*

---

```
x3650lab9:~ # echo 1 > /sys/class/scsi_disk/3\0\0\3/device/rescan
x3650lab9:~ # echo 1 > /sys/class/scsi_disk/3\0\1\3/device/rescan
x3650lab9:~ # echo 1 > /sys/class/scsi_disk/4\0\0\3/device/rescan
x3650lab9:~ # echo 1 > /sys/class/scsi_disk/4\0\1\3/device/rescan
```

---

Example 6-48 shows the message log that indicates the change in capacity.

*Example 6-48 Linux message log indicating the capacity change of a SCSI device*

---

```
x3650lab9:~ # tail /var/log/messages
...
Oct 13 16:52:25 x3650lab9 kernel: [ 9927.105262] sd 3:0:0:3: [sde] 209715200
512-byte logical blocks: (107 GB/100 GiB)
Oct 13 16:52:25 x3650lab9 kernel: [ 9927.105902] sde: detected capacity change
from 53687091200 to 107374182400
```

---

4. Indicate the device change to DM Multipath using the `resize_map` command of `multipathd`, as shown in Example 6-49. Afterwards, you can see the updated capacity in the output of `show topology`.

*Example 6-49 Resizing a multipath device*

---

```
x3650lab9:~ # multipathd -k"resize map 3600507630affc29f0000000000004791"
ok
x3650lab9:~ # multipathd -k"show top map 3600507630affc29f0000000000004791"
3600507630affc29f0000000000004791 dm-4 IBM,2107900
size=100G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
  |- 3:0:0:3 sde 8:64 active ready running
  |- 3:0:1:3 sdi 8:128 active ready running
  |- 4:0:1:3 sdq 65:0 active ready running
  `-- 4:0:0:3 sdm 8:192 active ready running
```

---

5. Resize the file system and check the new capacity, as shown in Example 6-50.

*Example 6-50 Resizing the file system and check capacity*

---

```
x3650lab9:~ # resize2fs /dev/mapper/3600507630affc29f0000000000004791
resize2fs 1.41.9 (22-Aug-2009)
Filesystem at /dev/mapper/3600507630affc29f0000000000004791 is mounted on
/mnt/itso_4791; on-line resizing required
old desc_blocks = 4, new_desc_blocks = 7
Performing an on-line resize of /dev/mapper/3600507630affc29f0000000000004791
to 26214400 (4k) blocks.
```

The filesystem on /dev/mapper/3600507630affc29f000000000004791 is now **26214400** blocks long.

```
x36501ab9:~ # df -h /mnt/itso_4791/
Filesystem                Size  Used Avail Use% Mounted on
/dev/mapper/3600507630affc29f000000000004791
                          99G  7.9G   86G   9% /mnt/itso_4791
```

**Dynamic volume increase:** Currently, the dynamic volume increase process has the following restrictions:

- ▶ From the supported Linux distributions, only SLES11 SP1 and RHEL 6 have this capability.
- ▶ The sequence works only with unpartitioned volumes. The file system must be created directly on the DM Multipath device.
- ▶ Only the modern file systems can be resized when they are mounted. The ext2 file system cannot.

## 6.4 Using FlashCopy and remote replication targets

DS8000 FlashCopy and remote replication solutions, such as IBM Metro Mirror or IBM Global Mirror, create bitwise identical copies of the source volumes. The target volume has a separate unique identifier, which is generated from the DS8000 WWNN and volume ID. Any metadata that is stored on the target volume, such as the file system identifier or LVM signature, is identical to that of the source volume. This can lead to confusion and data integrity problems, if you plan to use the target volume on the same Linux system as the source volume. Information presented in this section describe methods that avoid integrity issues when you use the target volume on the same Linux system as the source volume.

### 6.4.1 Using a file system residing on a DS8000 volume

The copy of a file system created directly on a SCSI disk device or single path, or a DM Multipath device, without an additional virtualization layer, such as RAID or LVM, can be used on the same host as the source without modification.

To use this copy of a file system, complete the following steps:

1. Mount the original file system by using a device node that is bound to the volume's unique identifier and not to any metadata that is stored on the device itself (Example 6-51).

*Example 6-51 Mounting the source volume*

```
x36501ab9:/mnt # mount /dev/mapper/3600507630affc29f000000000004791
/mnt/itso_4791/
x36501ab9:/mnt # mount
...
/dev/mapper/3600507630affc29f000000000004791 on /mnt/itso_4791 type ext3 (rw)
```

2. Verify that the data on the source volume is consistent, for example, by running the **sync command**.
3. Create the FlashCopy on the DS8000 and map the target volume to the Linux host. In this example the FlashCopy source has the volume ID 4791 and the target volume has ID 47b1.

4. Initiate a device scan on the Linux host. For more information, see “Adding and removing DS8000 volumes dynamically on Linux” on page 106. DM Multipath will automatically integrate the FlashCopy target, as shown in Example 6-52.

*Example 6-52 Checking DM Multipath topology for target volume*

---

```
x3650lab9:/mnt # multipathd -k"show topology"
...
3600507630affc29f0000000000004791 dm-4 IBM,2107900
size=100G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
  |- 3:0:0:3 sde 8:64 active ready running
  |- 3:0:1:3 sdi 8:128 active ready running
  |- 4:0:1:3 sdq 65:0 active ready running
  `-- 4:0:0:3 sdm 8:192 active ready running
3600507630affc29f00000000000047b1 dm-5 IBM,2107900
size=100G features='1 queue_if_no_path' hwhandler='0' wp=rw
`-+- policy='round-robin 0' prio=1 status=active
  |- 3:0:1:4 sds 65:32 active ready running
  |- 3:0:0:4 sdr 65:16 active ready running
  |- 4:0:1:4 sdu 65:64 active ready running
  `-- 4:0:0:4 sdt 65:48 active ready running
```

---

5. As shown in Example 6-53, mount the target volume to a separate mount point, using a device node that is created from the unique identifier of the volume.

*Example 6-53 Mounting the target volume*

---

```
x3650lab9:/mnt # mount /dev/mapper/3600507630affc29f0000000000047b1
/mnt/itso_fc/
x3650lab9:/mnt # mount
...
/dev/mapper/3600507630affc29f0000000000004791 on /mnt/itso_4791 type ext3 (rw)
/dev/mapper/3600507630affc29f0000000000047b1 on /mnt/itso_fc type ext3 (rw)
```

---

Now you can access both the original volume and the point-in-time copy through their respective mount points. This demonstration used an ext3 file system on a DM Multipath device, which is replicated with FlashCopy.

**Attention:** The device manager, udev, also creates device nodes that relate to the file system unique identifier (UUID) or label. These IDs are stored in the data area of the volume and are identical on both the source and target. Such device nodes are ambiguous, if the source and the target are mapped to the host, at the same time. Using them in this situation can result in data loss.

## 6.4.2 Using a file system residing in a logical volume managed by LVM

The Linux LVM uses metadata that is written to the data area of the disk device to identify and address its objects. If you want to access a set of replicated volumes that are under LVM control, this metadata must be modified and made unique to ensure data integrity. Otherwise, LVM can mix volumes from the source and the target sets.

A publicly available script called `vgimportclone.sh` automates the modification of the metadata and thus supports the import of LVM volume groups that reside on a set of replicated disk devices.



You can download this script from the “CVS log for LVM2/scripts/vgimportclone.sh” site at the following website:

<http://sources.redhat.com/cgi-bin/cvsweb.cgi/LVM2/scripts/vgimportclone.sh?cvsroot=lvm2>

For an online copy of the script, go to the UNIX Manual Page at the following website:

<http://www.cl.cam.ac.uk/cgi-bin/manpage?8+vgimportclone>

**Tip:** The `vgimportclone` script is part of the standard LVM tools for Red Hat Enterprise Linux 5. The SLES 11 distribution does not contain the script by default.

To ensure consistent data on the target volumes and avoid mixing up the source and target volume, use a volume group containing a logical volume that is striped over two DS8000 volumes, and FlashCopy to create a point in time copy of both volumes. Then, make both the original logical volume and the cloned volume available to the Linux system. The DS8000 volume IDs of the source volumes are 4690 and 4790, the IDs of the target volumes are 46a0 and 47a0.

The following steps demonstrate this procedure in detail:

1. Mount the original file system using the LVM logical volume device (Example 6-54).

*Example 6-54 Mounting the source volume*

---

```
x36501ab9:~ # mount /dev/mapper/vg_ds8000-lv_itso /mnt/lv_itso/
x36501ab9:~ # mount
...
/dev/mapper/vg_ds8000-lv_itso on /mnt/lv_itso type ext3 (rw)
```

---

2. Make sure the data on the source volume is consistent, for example, by running the `sync` command.
3. Create the FlashCopy on the DS8000 and map the target volumes 46a0 and 47a0 to the Linux host.
4. Initiate a device scan on the Linux host, see “Adding and removing DS8000 volumes dynamically on Linux” on page 106 for details. DM Multipath automatically integrates the FlashCopy targets as shown in Example 6-55.

*Example 6-55 Checking DM Multipath topology for target volume*

---

```
x36501ab9:~ # multipathd -k"show topology"
...
3600507630affc29f0000000000046a0 dm-6 IBM,2107900
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
^-+- policy='round-robin 0' prio=1 status=active
  |- 3:0:0:4 sdf 8:80 active ready running
  |- 3:0:1:4 sdl 8:176 active ready running
  |- 4:0:0:4 sdr 65:16 active ready running
  ^- 4:0:1:4 sdx 65:112 active ready running
3600507630affc29f0000000000047a0 dm-7 IBM,2107900
size=50G features='1 queue_if_no_path' hwhandler='0' wp=rw
^-+- policy='round-robin 0' prio=1 status=active
  |- 3:0:0:5 sdg 8:96 active ready running
  |- 3:0:1:5 sdm 8:192 active undef running
  |- 4:0:1:5 sdy 65:128 active ready running
  ^- 4:0:0:5 sds 65:32 active ready running
```

---

**Important:** To avoid data integrity issues, it is important that LVM configuration commands are not issued until step 5 is complete.

5. Run the `vgimportclone.sh` script against the target volumes, providing a new volume group name, as shown in Example 6-56.

*Example 6-56 Adjusting the target volumes' LVM metadata*

```
x3650lab9:~ # ./vgimportclone.sh -n vg_itso_fc
/dev/mapper/3600507630affc29f00000000000046a0
/dev/mapper/3600507630affc29f00000000000047a0
WARNING: Activation disabled. No device-mapper interaction will be attempted.
Physical volume "/tmp/snap.ptY8xojo/vgimport1" changed
1 physical volume changed / 0 physical volumes not changed
WARNING: Activation disabled. No device-mapper interaction will be attempted.
Physical volume "/tmp/snap.ptY8xojo/vgimport0" changed
1 physical volume changed / 0 physical volumes not changed
WARNING: Activation disabled. No device-mapper interaction will be attempted.
Volume group "vg_ds8000" successfully changed
Volume group "vg_ds8000" successfully renamed to "vg_itso_fc"
Reading all physical volumes. This may take a while...
Found volume group "vg_itso_fc" using metadata type lvm2
Found volume group "vg_ds8000" using metadata type lvm2
```

6. Activate the volume group on the target devices and mount the logical volume, as shown in Example 6-57.

*Example 6-57 Activating volume group on the target device and mount the logical volume*

```
x3650lab9:~ # vgchange -a y vg_itso_fc
1 logical volume(s) in volume group "vg_itso_fc" now active
x3650lab9:~ # mount /dev/vg_itso_fc/lv_itso /mnt/lv_fc_itso/
x3650lab9:~ # mount
...
/dev/mapper/vg_ds8000-lv_itso on /mnt/lv_itso type ext3 (rw)
/dev/mapper/vg_itso_fc-lv_itso on /mnt/lv_fc_itso type ext3 (rw)
```

## 6.5 Troubleshooting and monitoring

This section provides information about the following areas of troubleshooting and monitoring:

- ▶ Checking SCSI devices alternate methods
- ▶ Monitoring performance with the `iostat` command
- ▶ Working with generic SCSI tools
- ▶ Booting Linux from DS8000 volumes
- ▶ Configuring the QLogic BIOS to boot from a DS8000 volume
- ▶ Understanding the OS loader considerations for other platforms
- ▶ Installing SLES11 SP1 on a DS8000 volume

## 6.5.1 Checking SCSI devices alternate methods

The Linux kernel maintains a list of all attached SCSI devices in the `/proc` pseudo file system. The file `/proc/scsi/scsi` contains basically the same information, apart from the device node, as the `ls SCSI` output. It is always available, even if `ls SCSI` is not installed. To view the list, as shown in Example 6-58.

*Example 6-58 An alternative list of attached SCSI devices*

---

```
x3650lab9:~ # cat /proc/scsi/scsi
Attached devices:
Host: scsi0 Channel: 00 Id: 00 Lun: 00
  Vendor: ServeRA Model: Drive 1 Rev: V1.0
  Type: Direct-Access ANSI SCSI revision: 05
Host: scsi2 Channel: 00 Id: 00 Lun: 00
  Vendor: MATSHITA Model: UJDA770 DVD/CDRW Rev: 1.24
  Type: CD-ROM ANSI SCSI revision: 05
Host: scsi3 Channel: 00 Id: 00 Lun: 00
  Vendor: IBM Model: 2107900 Rev: .288
  Type: Direct-Access ANSI SCSI revision: 05
Host: scsi3 Channel: 00 Id: 00 Lun: 01
  Vendor: IBM Model: 2107900 Rev: .288
  Type: Direct-Access ANSI SCSI revision: 05
Host: scsi3 Channel: 00 Id: 00 Lun: 02
  Vendor: IBM Model: 2107900 Rev: .288
  Type: Direct-Access ANSI SCSI revision: 05
Host: scsi3 Channel: 00 Id: 00 Lun: 03
  Vendor: IBM Model: 2107900 Rev: .288
  Type: Direct-Access ANSI SCSI revision: 05
Host: scsi3 Channel: 00 Id: 01 Lun: 00
  Vendor: IBM Model: 2107900 Rev: .288
  Type: Direct-Access ANSI SCSI revision: 05
...
```

---

The `fdisk -l` command can be used to list all block devices, including their partition information and capacity, but it does not contain the SCSI address, vendor, and model information. To view the list, see Example 6-59.

*Example 6-59 Output of the fdisk -l command*

---

```
x3650lab9:~ # fdisk -l

Disk /dev/sda: 146.7 GB, 146695782400 bytes
255 heads, 63 sectors/track, 17834 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0x5bbe5807

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1  *           1          5221    41937651   83  Linux
/dev/sda2                5222        10442    41937682+   83  Linux
/dev/sda3           10443         11486     8385930    82  Linux swap / Solaris

Disk /dev/sdb: 53.7 GB, 53687091200 bytes
64 heads, 32 sectors/track, 51200 cylinders
Units = cylinders of 2048 * 512 = 1048576 bytes
Disk identifier: 0x00000000
```

Disk /dev/sdb doesn't contain a valid partition table

Disk /dev/sdd: 53.7 GB, 53687091200 bytes  
64 heads, 32 sectors/track, 51200 cylinders  
Units = cylinders of 2048 \* 512 = 1048576 bytes  
Disk identifier: 0x00000000

Disk /dev/sdd doesn't contain a valid partition table  
...

---

## 6.5.2 Monitoring performance with the iostat command

Use the **iostat** command to monitor the performance of all attached disks. It is part of the **sysstat** package that ships with every major Linux distribution, but is not necessarily installed by default. The **iostat** command reads data provided by the kernel in `/proc/stats` and prints it in readable format. For more details, see the man page for **iostat**.

## 6.5.3 Working with generic SCSI tools

For Linux, there is a set of tools that allow low-level access to SCSI devices. These tools are called the *sg\_tools*. They communicate with SCSI devices through the generic SCSI layer, which is represented by the `/dev/sg0`, `/dev/sg1`, and so on, special device files. In recent Linux versions, the *sg\_tools* can also access the block devices `/dev/sda`, `/dev/sdb`, or any other device node that represents a SCSI device directly.

The following *sg\_tools* are useful:

- ▶ **sg\_inq /dev/sgx** prints SCSI inquiry data, such as the volume serial number.
- ▶ **sg\_scan** prints the SCSI host, channel, target, and LUN mapping for all SCSI devices.
- ▶ **sg\_map** prints the `/dev/sdx` to `/dev/sgy` mapping for all SCSI devices.
- ▶ **sg\_readcap /dev/sgx** prints the block size and capacity in blocks of the device.
- ▶ **sginfo /dev/sgx** prints SCSI inquiry and mode page data. It also helps you manipulate the mode pages.

## 6.5.4 Booting Linux from DS8000 volumes

You can configure a system to load the Linux kernel and operating system from a SAN-attached DS8000 volume (boot from SAN). This section explains this process, using SLES11 SP1 on an x86 server with QLogic FC HBAs.

### The Linux boot process

In order to understand the configuration steps required to boot a Linux system from SAN-attached DS8000 volumes, you need a basic understanding of the Linux boot process. Therefore, this process is briefly summarized in the following sections for a Linux system, until the point in the process where the well known login prompt is presented.

### OS loader

The system firmware provides functions for rudimentary I/O operations, for example, the BIOS of x86 servers. When a system is turned on, it first performs the *power-on self-test* (POST) to check the hardware available and verify that everything is working. Then it runs the operating system loader (OS loader), which uses basic I/O routines to read a specific location on the defined system disk and executes the code it contains. This code either is part of the boot loader of the operating system, or it branches to the location where the boot loader resides.

If you want to boot from a SAN-attached disk, ensure that the OS loader can access this disk. FC HBAs provide an extension to the system firmware for this purpose. In many cases, it must be specifically activated.

On x86 systems, this location is called the *master boot record* (MBR).

**Tip:** For Linux under z/VM, the OS loader is not part of the firmware but the z/VM `ipl` program.

## The boot loader

The purpose of the boot loader is to start the operating system kernel. To do this, it must recognize the physical location of the kernel image on the system disk, read it in, unpack it (if it is compressed), and start it. All of it is done using the basic I/O routines provided by the firmware. The boot loader can also pass configuration options and the location of the **InitRAMFS** to the kernel. The following most common Linux boot loaders are:

- ▶ Grand unified boot loader (GRUB) for x86 systems
- ▶ `zipl` boot loader for System z
- ▶ `yaboot` for Power Systems

## The kernel and the initRAMFS

After the kernel is unpacked and running, it takes control over the system hardware. It starts and sets up memory management, interrupt handling, and the built in hardware drivers for the hardware that is common on all systems, such as the memory management unit, clock, and so on. It reads and unpacks the initRAMFS image, using the same basic I/O routines.

The initRAMFS contains additional drivers and programs that are needed to set up the Linux file system tree, or root file system. To be able to boot from a SAN-attached disk, the standard initRAMFS must be extended with the FC HBA driver and the multipathing software. In recent Linux distributions, it is done automatically by the tools that create the initRAMFS image.

After the root file system is accessible, the kernel starts the `init()` process.

## The init() process

The `init()` process brings up the operating system itself, such as networking, services, user interfaces, and so on. At this point, the hardware is already completely abstracted. Therefore, `init()` is not platform dependent, and there are not any SAN-boot specifics.

A detailed description of the Linux boot process for x86 based systems can be found on IBM developerWorks® at this website:

<http://www.ibm.com/developerworks/linux/library/l-linuxboot/>

## 6.5.5 Configuring the QLogic BIOS to boot from a DS8000 volume

To configure the HBA to load a BIOS extension that provides the basic I/O capabilities for a SAN-attached disk, see 2.2, “Using the DS8000 as a boot device” on page 8.

**Tip:** For IBM branded QLogic HBAs, the BIOS setup utility is called *FAST!Util*.

## 6.5.6 Understanding the OS loader considerations for other platforms

The BIOS is the method that an x86 system uses to start loading the operating system. You can find information and considerations on how it is done on other supported platforms in this section.

### IBM Power Systems

When you install Linux on an IBM Power System server or LPAR, the Linux installer sets the boot device in the firmware to the drive on which you are installing. You do not need to take any special precautions, regardless of whether you install on a local disk, a SAN-attached DS8000 volume, or a virtual disk provided by the VIOS.

### IBM System z

Linux on System z can be loaded from traditional CKD devices or from Fibre Channel attached fixed block (SCSI) devices. To load from SCSI disks, the SCSI feature (FC 9904) named *initial program load* (IPL) must be installed and activated on the System z server. SCSI IPL is generally available on recent IBM System z machines (IBM z10™ and later).

**Attention:** Activating the SCSI IPL feature is disruptive. It requires a *power-on reset* (POR) of the whole system.

Linux on System z can run in the following two separate configurations:

- ▶ Linux on System z running natively in a System z LPAR:

After installing Linux on System z, you need to provide the device from where the LPAR runs the IPL in the LPAR start dialog on the System z *support element*. After it is registered there, the IPL device entry is permanent until changed.

- ▶ Linux on System z running under z/VM:

Within z/VM, start an operating system with the IPL command. With the IPL command, you must provide the z/VM device address where the Linux boot loader and kernel is installed.

When booting from SCSI disk, the disk does not have a z/VM device address. For more information, see “System z” on page 88. You must provide the information for the LUN that the machine loader uses to start the operating system separately. The z/VM system provides the **set loaddev** and **query loaddev cp** commands, for this purpose, as shown in Example 6-60.

*Example 6-60 Set and query SCSI IPL device in z/VM*

---

```
SET LOADDEV PORTNAME 50050763 0a00029f LUN 40494000 00000000
```

```
QUERY LOADDEV
PORTNAME 50050763 0A00029F      LUN  40494000 00000000      BOOTPROG 0
BR_LBA   00000000 00000000
```

---

The port name provided is the DS8000 host port that is used to access the boot volume. After the load device is set, use the IPL program with the device number of the Fibre Channel HBA that connects to the DS8000 port and LUN to boot from. Automate the IPL by adding the required commands to the z/VM profile of the virtual machine.

## 6.5.7 Installing SLES11 SP1 on a DS8000 volume

With recent Linux distributions, the installation on a DS8000 volume is as easy as the installation on a local disk. For installation, keep in mind these additional considerations:

- ▶ Identifying the right DS8000 volumes to install on
- ▶ Enabling multipathing during installation

**Tip:** After the SLES11 installation program (YAST) is running, the installation is mostly hardware platform independent. It works the same, regardless of running on an x86, IBM Power System, or System z server.

Start the installation process by booting from an installation DVD and follow the installation configuration panels, until you reach the Installation Settings panel (Figure 6-3).

**Using the GUI:** The Linux on System z installer does not automatically list the available disks for installation. You will see a Configure Disks panel before you get to the Installation Settings, where you can discover and attach the disks that are needed to install the system using a GUI. At least one disk device is required to perform the installation.

To install SLES11 SP1 on a DS8000 volume, follow these steps:

1. In the Installation Settings panel (Figure 6-3), click **Partitioning** to perform the configuration steps required to define the DS8000 volume as system disk.



Figure 6-3 SLES11 SP1 installation settings

2. In the Preparing Hard Disk: Step 1 panel (Figure 6-4), select the **Custom Partitioning (for experts)** button. It does not matter which disk device is selected in the Available Disks field. Then click **Next**.

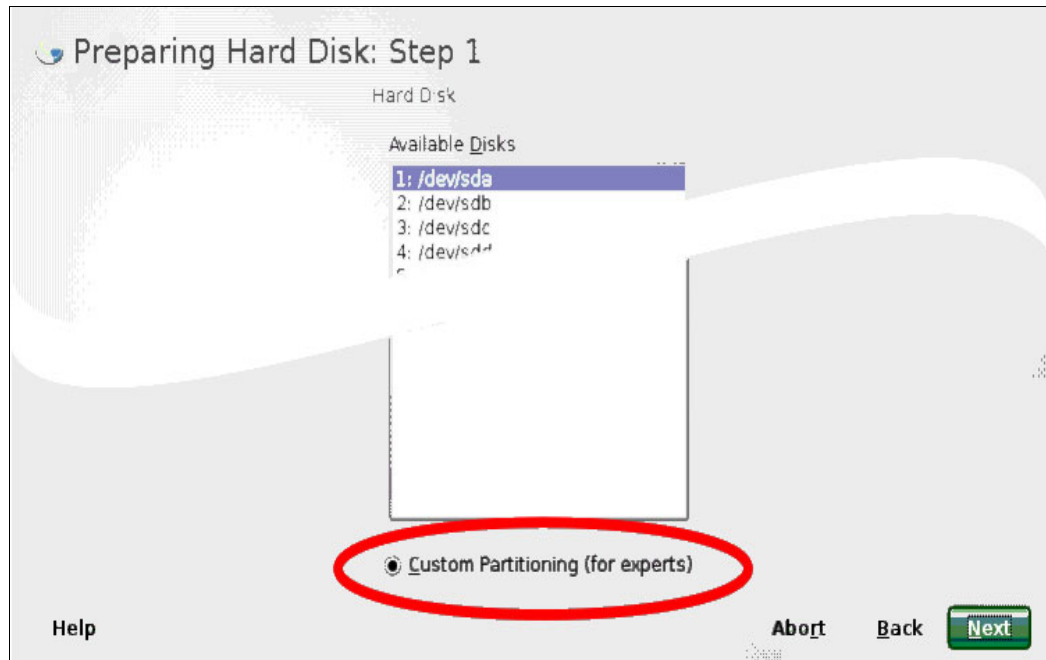


Figure 6-4 Preparing Hard Disk: Step 1

3. In the Expert Partitioner panel (Figure 6-5), enable multipathing:
  - a. Select **Harddisks** in the navigation section on the left side.
  - b. Click the **Configure** button in the bottom right corner of the main panel.
  - c. Select **Configure Multipath**.

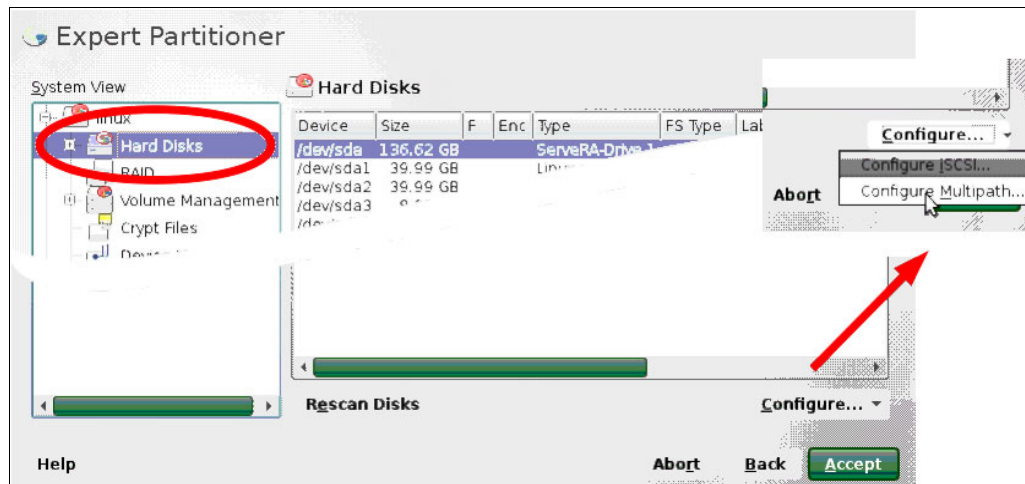


Figure 6-5 Enabling multipath in partitioner



- Click **Yes** when the tool asks for confirmation. The tool rescans the disk devices and provides an updated list of hard disks that also shows the multipath devices it has found, as you can see in Figure 6-6.

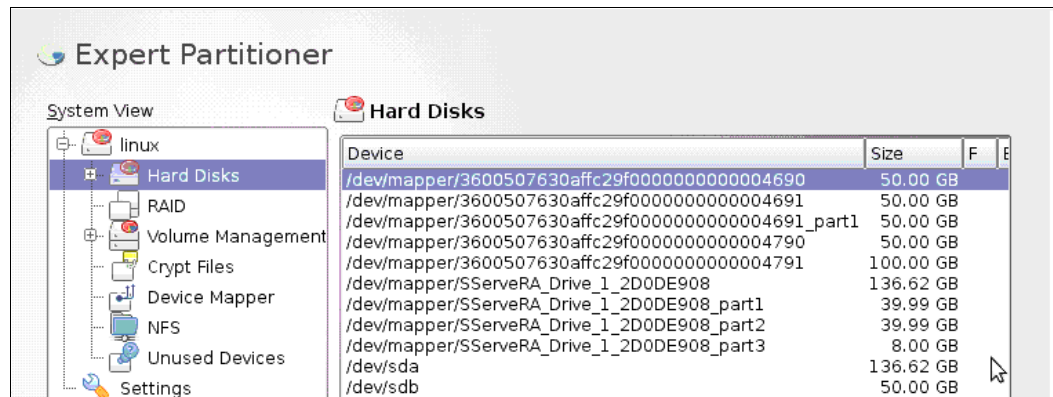


Figure 6-6 Selecting multipath device for installation

- Select the multipath device (DS8000 volume) to install to and click **Accept**.
- On the Partitioner panel, create and configure the required partitions for your system in the method you use on a local disk.

### 6.5.8 Installing with YAST

You can also use the automatic partitioning capabilities of YAST, after the multipath devices were detected. Follow these steps:

- Click the **Back** button until you see the Preparing Hard Disk: Step 1 panel (Figure 6-4 on page 120).
- When you see the multipath devices instead of the disks, select the multipath device you want to install on and click **Next**.
- Select the partitioning scheme.

**Important:** All supported platforms can boot Linux from multipath devices. In certain cases, however, the tools that install the boot loader can only write to simple disk devices. You will need to install the boot loader with multipathing deactivated. SLES10 and SLES11 provide this capability with the parameter `multipath=off` to the boot command in the boot loader.

The boot loader for IBM Power Systems and System z must be installed again when the kernel or InitRAMFS is updated. A separate entry in the boot menu provides the ability to switch between the single and multipath mode when necessary.

See the Linux distribution specific documentation in 6.1.2, “Attaching Linux server to DS8000 resources” on page 82, for more detail.

The installer does not implement any device specific settings, such as creating the `/etc/multipath.conf` file. You must do this manually after the installation. See 6.2.9, “Attaching DS8000: Considerations” on page 104, for more information.

Because DM Multipath is already started during the processing of the InitRAMFS, you must also build a new InitRAMFS image after changing the DM Multipath configuration. See “Initial Fibre Channel driver availability” on page 91.

**Linux tips:** It is possible to add device mapper layers on top of DM Multipath, such as software RAID or LVM. The Linux installers support these options.

Red Hat Enterprise Linux 5.1 and later also supports multipathing for the installation. You can enable this option by adding the `mpath` option to the kernel boot line of the installation system. Anaconda, the Red Hat installer, then offers to install to multipath devices.



## VMware vSphere considerations

This chapter provides information about the technical specifications for attaching IBM System Storage DS8000 systems to host systems running VMware vSphere. There are multiple versions of VMware vSphere available. However, this chapter presents information based on vSphere Version 5. Most of this information also applies to earlier versions.

The following topics are covered:

- ▶ vSphere introduction
- ▶ vSphere storage concepts
- ▶ Multipathing
- ▶ Storage vMotion
- ▶ SSD detection
- ▶ Best practices

## 7.1 vSphere introduction

VMware vSphere consists of several different components. The major component is the ESXi server, a bare metal hypervisor that runs on x86 servers. ESXi provides a virtualization platform for a large variety of open systems such as Windows, Linux, or Solaris.

Before version 5, customers could choose between ESX and ESXi. While ESX contains an integrated service console for management, ESXi only consists of the hypervisor and requires the installation of external agents for monitoring and administration. ESXi has a small footprint that allows it to be included in the server firmware. Both ESX and ESXi provide the same capabilities in regard to virtual machines. Because vSphere 5 and later will only contain ESXi, this publication focuses on ESXi.

ESXi can connect to a variety of storage devices:

- ▶ Local disks installed in the physical server
- ▶ Fibre Channel SAN
- ▶ iSCSI SAN
- ▶ NAS devices, such as NFS shares

For the management of vSphere, VMware provides two components (Figure 7-1):

- ▶ vCenter Server:  
A management suite to manage multiple ESXi hosts and to provide advanced functions (such as on-line migrations, cloning, high availability, and so on)
- ▶ vSphere Client:  
A Windows software used to connect to either single ESXi hosts or vCenter Server that provides a graphical user interface.

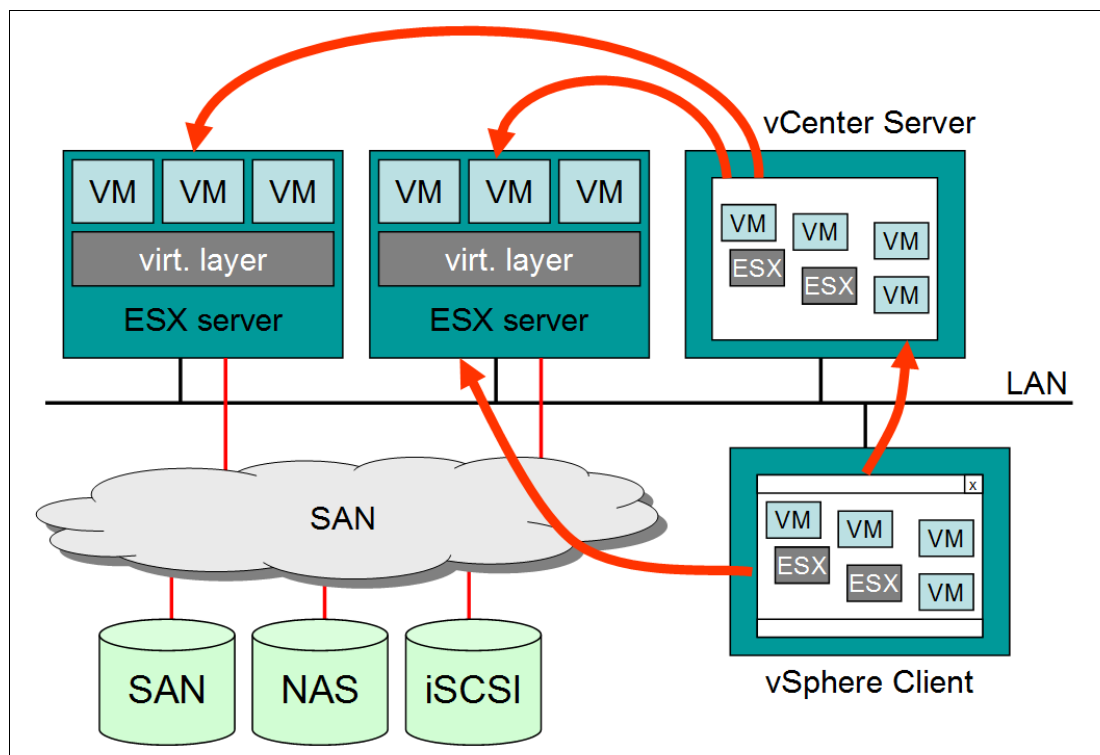


Figure 7-1 vSphere components

## 7.2 vSphere storage concepts

This section focuses on the concepts of how storage is handled by ESXi.

### 7.2.1 VMFS

ESXi uses a special file system to store virtual machines called Virtual Machine File System (VMFS). VMFS is a clustered file system that can be accessed concurrently by up to 8 ESXi hosts.

A VMFS partition can be spread over 32 logical volumes called extents. It can have a maximum size of 64 TB, and the maximum file size on a VMFS partition is 2 TB.

On a VMFS partition, ESXi stores all the files of which a VM consists:

- ▶ Virtual machine configuration file (.vmx)
- ▶ Virtual machine BIOS file (.nvram)
- ▶ Virtual disk files or raw device mappings (.vmdk)
- ▶ Virtual machine log files (.log)

VMs can use two different concepts for their virtual harddrives: virtual disks and raw device mappings (RDM). Both concepts are shown in Figure 7-2 and explained further in this chapter.

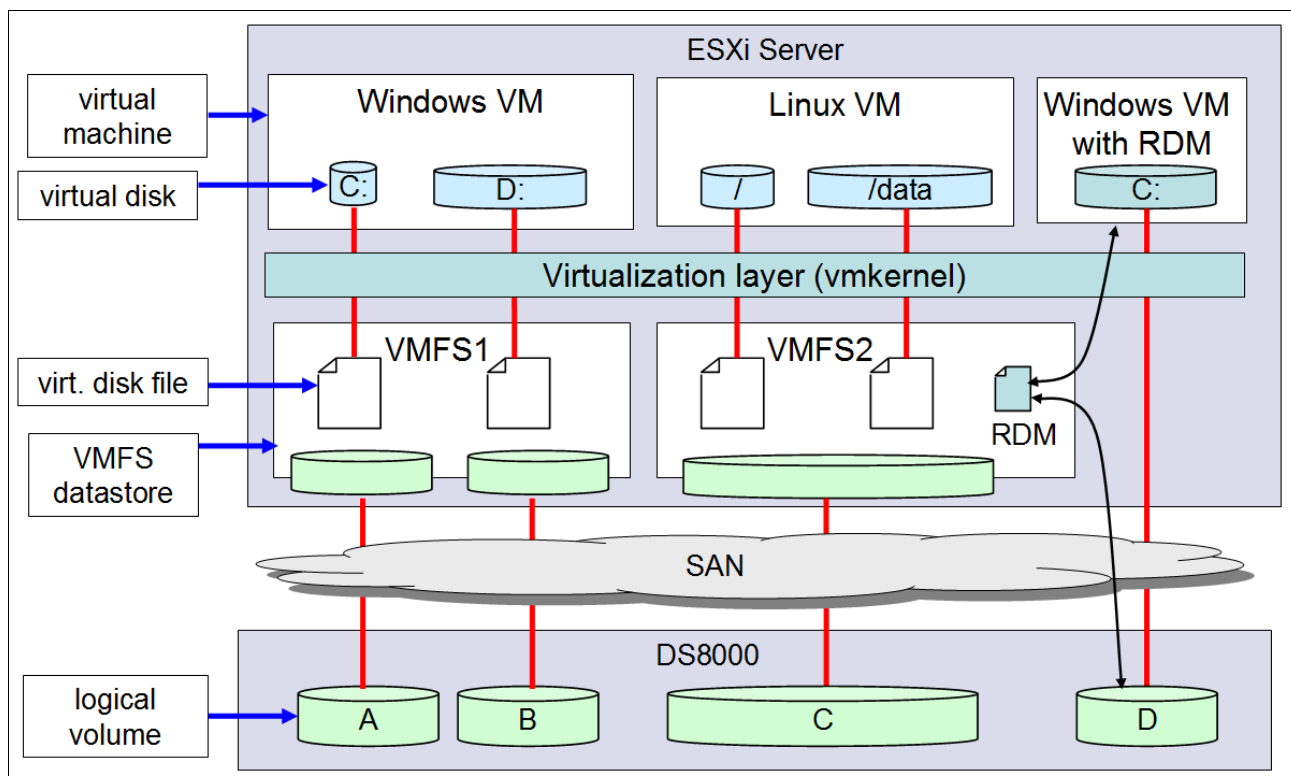


Figure 7-2 Storage concepts of ESXi

## Creating a VMFS datastore

The following steps describe the tasks required to create a datastore on a DS8000 volume:

1. Ensure that the host is correctly zoned to the DS8000 (see 1.3, “SAN considerations” on page 3 for details on the SAN configuration).

**Tip:** To find out the WWPN of the HBA in ESXi, select the host, click the **Configuration** tab, and select **Storage Adapters**. The WWNN and WWPN are listed in a column next to the adapter name. See Figure 7-3.

Device	Type	WWN
<b>82801G (ICH7 Family) IDE Controller</b>		
vmhba0	Block SCSI	
vmhba32	Block SCSI	
<b>ISP81xx-based 10 GbE FCoE to PCI Express CNA</b>		
vmhba6	Fibre Channel	20:00:00:c0:dd:18:3b:79 21:00:00:c0:dd:18:3b:79
vmhba7	Fibre Channel	20:00:00:c0:dd:18:3b:7b 21:00:00:c0:dd:18:3b:7b
<b>ISP2532-based 8Gb Fibre Channel to PCI Express HBA</b>		
vmhba2	Fibre Channel	20:00:00:24:ff:2d:c3:40 21:00:00:24:ff:2d:c3:40
vmhba3	Fibre Channel	20:00:00:24:ff:2d:c3:41 21:00:00:24:ff:2d:c3:41
vmhba4	Fibre Channel	20:00:00:24:ff:2d:c5:68 21:00:00:24:ff:2d:c5:68
vmhba5	Fibre Channel	20:00:00:24:ff:2d:c5:69 21:00:00:24:ff:2d:c5:69

Figure 7-3 Find WWPN in ESXi

2. Create at least one FB volume.
3. Create a volume group using hosttype **VMware** and assign the volume(s) to the group.

**Create New Volume Group**

**Define Volume Group Properties**  
Define the volume group properties. A volume group is a set of logical volumes that can be accessed by a host.

\*Volume Group Nickname:

\*Host Type:

Figure 7-4 DS Storage Manager GUI: Select hosttype for volume group

4. Create host connections using hosttype **VMware**.

**Create New Host**

**Define Host Ports**  
Define one or more host ports that you will use to map hosts to a volume group in the next step. After you add the table is updated. All of the host ports in the table will be mapped to the same volume group when you create

\*Host Nickname:

\*Port Type:

\*Host Type:

Figure 7-5 DS Storage Manager GUI: Select hosttype for host connection

5. Map the volume group to the host.
6. In the vSphere client, go to **Configuration** → **Storage adapters** and click **Rescan All** in the top right corner (see Figure 7-6).

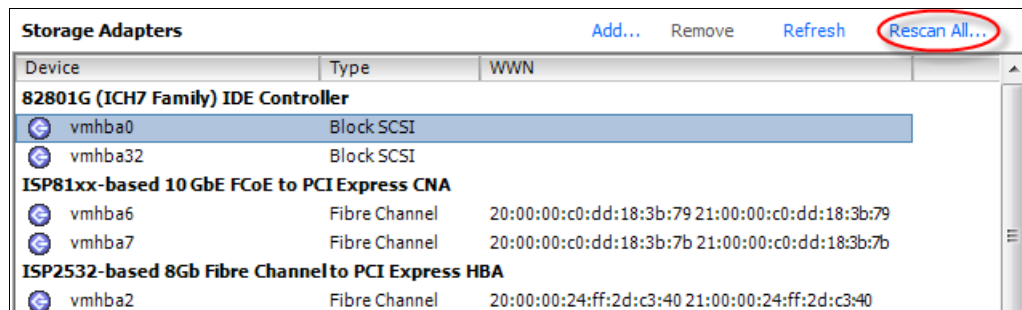


Figure 7-6 Rescan devices

7. Verify that all volumes are correctly detected by selecting each vmhba connected to the DS8000. The detected volumes are listed in the details (see Figure 7-7).

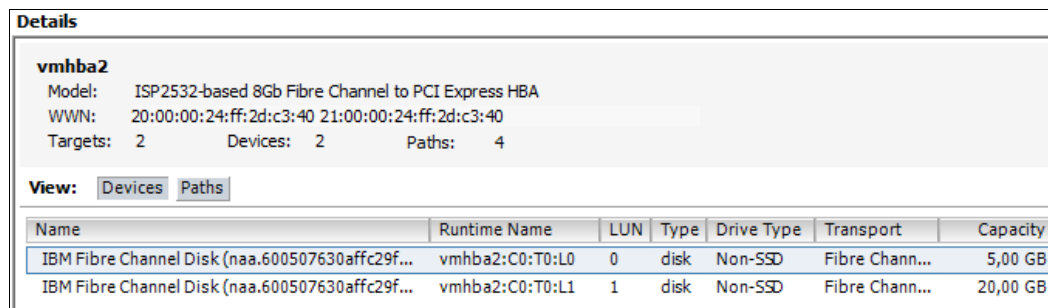


Figure 7-7 detected volumes

8. To create a new datastore on the detected volumes, go to **Configuration** → **Storage** and click **Add storage** as shown in Figure 7-8. This action starts a wizard that guides you through the steps to create a new VMFS datastore.

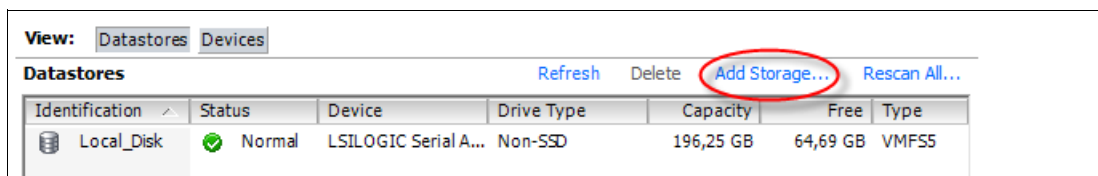


Figure 7-8 Add Storage

- a. In Figure 7-9, select storage type **Disk/LUN** and click **Next**.

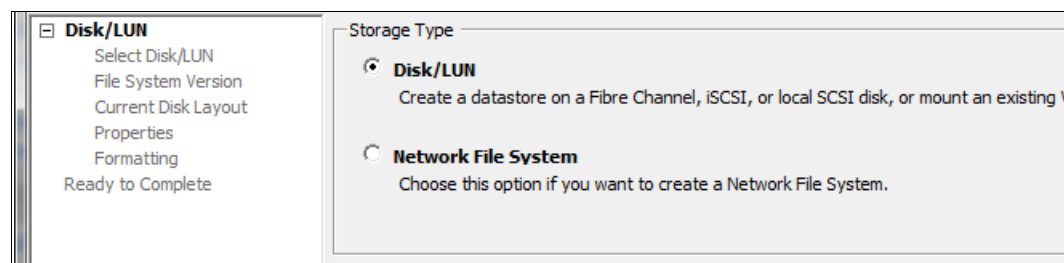


Figure 7-9 Select Storage Type

- b. Select the LUN to be used for the datastore from the list in Figure 7-10 and click **Next**.

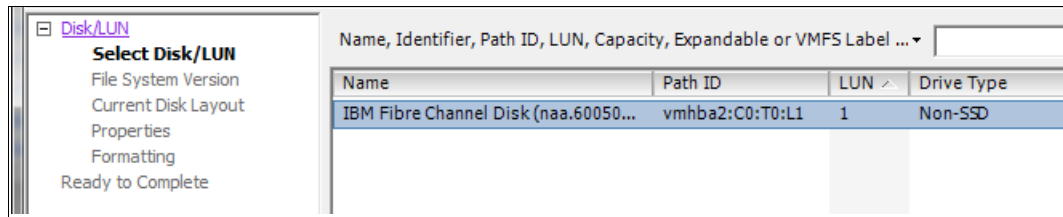


Figure 7-10 Select Disk/LUN

- c. Select the VMFS version as shown in Figure 7-11 and click **Next**.

**Tip:** If the datastore must be shared with ESX hosts under version 5, select VMFS-3.

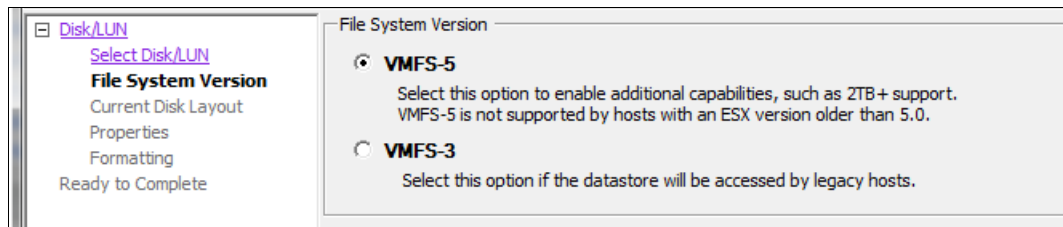


Figure 7-11 File System Version

- d. Review the current disk layout in the page shown in Figure 7-12. Ensure that the LUN is empty and that no data will be overwritten. Click **Next**.

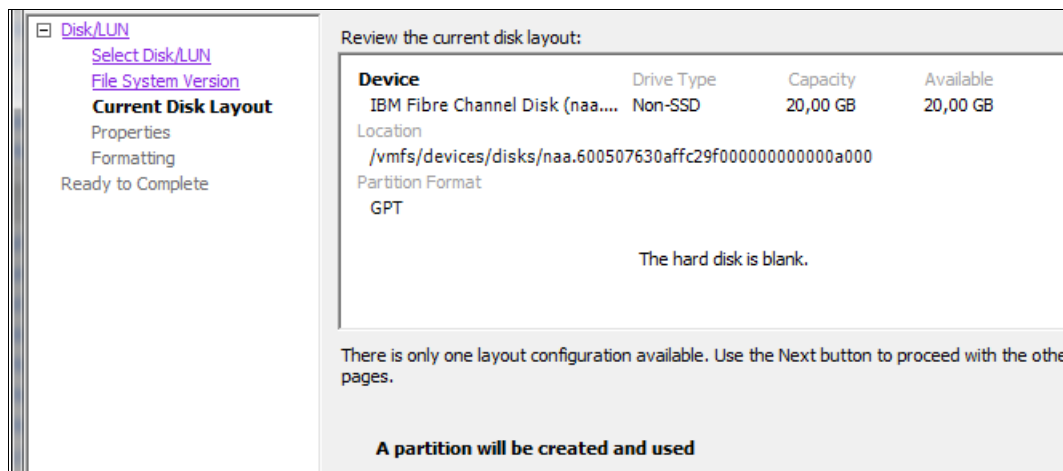


Figure 7-12 Current disk layout



- e. Enter a name for the new datastore and click **Next** (Figure 7-13).

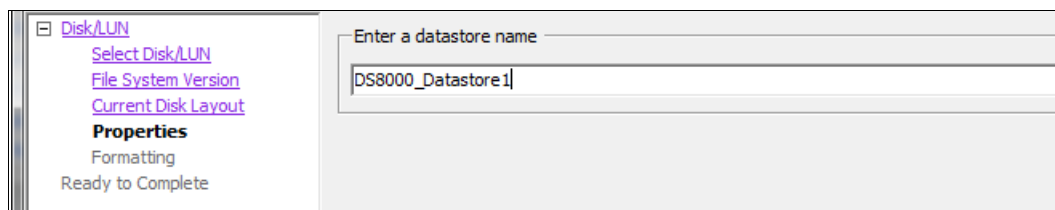


Figure 7-13 Datastore name

- f. Select the size of the new datastore from the page shown in Figure 7-14. Click **Next**.

**Tip:** Always use the full size of the LUN. Having multiple datastores or partitions on one LUN is not a good practice.

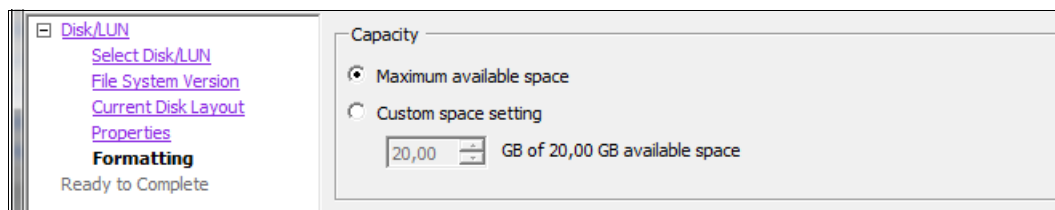


Figure 7-14 Datastore size

- g. Check the summary in the Ready to Complete window shown in Figure 7-15 and click **Finish**.

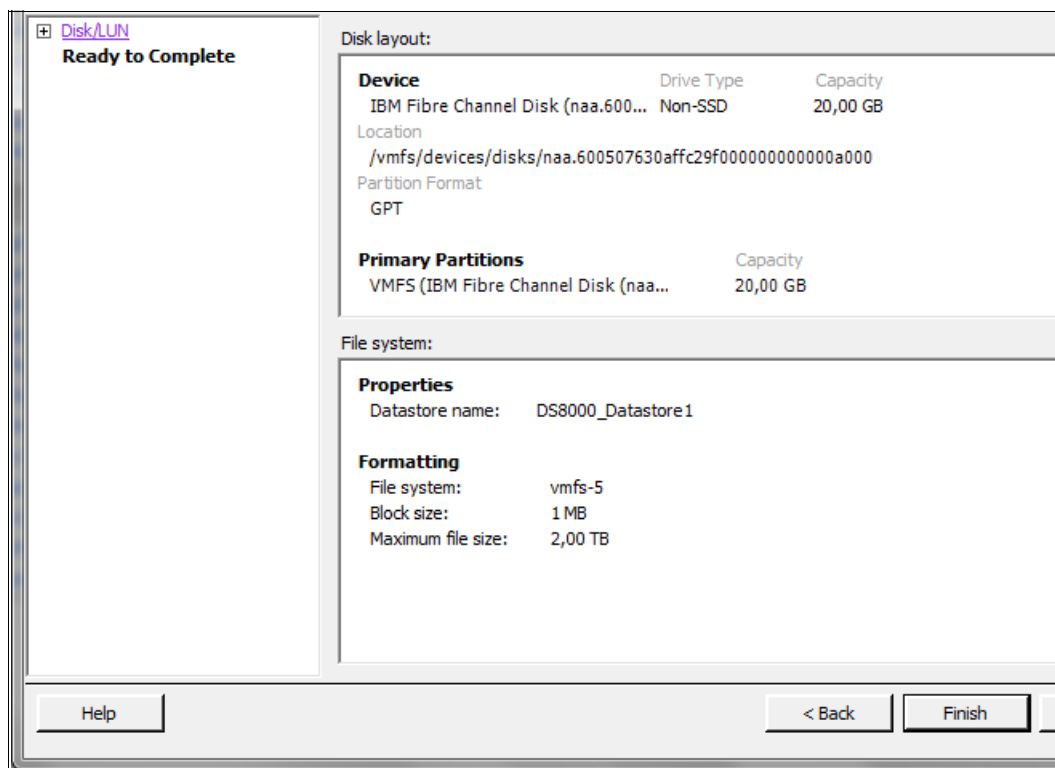


Figure 7-15 Add storage summary page

- The new datastore will be listed under **Configuration** → **Storage**. If you select the datastore, more information is shown in the **Datastore Details** area (Figure 7-16).

The screenshot shows the vSphere Storage configuration interface. At the top, there are tabs for 'View: Datastores' and 'Devices'. Below this is a table of Datastores:

Identification	Status	Device	Drive Type	Capacity	Free	Type	Last Update
DS8000_Datastore1	Normal	IBM Fibre Channel...	Non-SSD	19,75 GB	18,85 GB	VMFS5	31.05.2012 13:52:34
Local_Disk	Normal	LSILOGIC Serial A...	Non-SSD	196,25 GB	64,69 GB	VMFS5	31.05.2012 13:52:34

Below the table is the 'Datastore Details' section for 'DS8000\_Datastore1'. It shows the location, hardware acceleration status, and a pie chart indicating storage usage (920,00 MB Used, 18,85 GB Free). The 'Path Selection' section shows 'Round Robin (VM...)' and 'Storage I/O Control' is 'Disabled'. The 'Extents' section shows 'IBM Fibre Channel Disk (naa..)' with a capacity of 20,00 GB. The 'Formatting' section shows 'File System: VMFS 5.54' and 'Block Size: 1 MB'. The 'Paths' section shows 'Total: 8', 'Broken: 0', and 'Disabled: 0'.

Figure 7-16 Datastore details

## Fixed LUN ID for ESXi volumes

For certain scenarios, it is required that the same volume is represented with the same LUN or SCSI ID on all hosts. The LUN ID represents the position on the SCSI bus.

By default, the DS8000 does not specify this ID, and the ESXi host attaches the volumes in the order they appear starting with 0. It might lead to two ESXi hosts seeing the same volume on different positions in the SCSI bus.

In Example 7-1, volume A005 is assigned to two hosts. While host 1 sees the volume in position 00, host 2 sees it in position 01.

### Example 7-1 Different LUN ID for two hosts

```

dscli> showvolgrp -lunmap v64
Date/Time: 1. Juni 2012 12:41:00 CEST IBM DSCLI Version: 7.6.20.221 DS: 75-TV181
Name ITS0-ESX-1
ID V64
Type SCSI Map 256
Vols A005
=====LUN Mapping=====
vol lun
=====
A005 00

dscli> showvolgrp -lunmap v65
Date/Time: 1. Juni 2012 12:41:02 CEST IBM DSCLI Version: 7.6.20.221 DS: 75T-V181
Name ITS0-ESX-2
ID V65
Type SCSI Map 256
Vols A000 A005

```

```

=====LUN Mapping=====
vol lun
=====
A000 00
A005 01

```

To avoid a situation where two hosts see the same LUN with different SCSI IDs, it is possible to specify the SCSI ID when assigning the volume to a volume group.

### Using DS Storage Manager GUI

When assigning a volume to a volume group, follow these steps:

1. Go to **Volumes** → **Volume Groups** → **select volume group** → **Action** → **Properties**.
2. From the **Action** drop-down menu, select **Add volumes**. Select the volume and click **Next**.
3. In the LUN ID assignment window, double-click the value in the **LUN ID** column and enter the desired value as in Figure 7-17.



Figure 7-17 Specify LUN ID

4. Click **Next** and **Finish**.

### Using DS CLI

When assigning a volume to the volume group with **chvolgrp**, use the parameter **-lun** to specify the LUN ID as shown in Example 7-2.

*Example 7-2 Specify LUN ID*

```

dscli> chvolgrp -action add -volume A005 -lun 1 v64
Date/Time: 1. Juni 2012 13:11:08 CEST IBM DSCLI Version: 7.6.20.221 DS:
IBM.2107-75TV181
CMUC00031I chvolgrp: Volume group V64 successfully modified.

dscli> showvolgrp -lunmap v64
Date/Time: 1. Juni 2012 13:11:12 CEST IBM DSCLI Version: 7.6.20.221 DS:
IBM.2107-75TV181
Name ITSO-ESX-1
ID V64
Type SCSI Map 256
Vols A005
=====LUN Mapping=====
vol lun
=====
A005 01

```

## Increasing the capacity of a datastore

If the storage usage of the VMs on a datastore grows, it might be necessary to increase its capacity, which can be achieved in two ways:

1. Add an additional extent (LUN) to the datastore.
2. Increase the size of the LUN on which the datastore resides.

A datastore can be spread over up to 32 extents. However, ESXi does not use striping to spread the data across all extents. Therefore, adding extents to existing datastores can lead to performance issues. It causes increased administrative effort because all extents of a datastore need to be handled simultaneously (for example, during array based snapshots).

**Tip:** An ESXi host can handle a maximum of 256 LUNs.

To increase the size of the LUN on which the datastore resides, perform the following steps:

1. Increase the size of the volume:
  - a. Using the DS Storage Manager:

Go to **FB Volumes** → **Manage existing volumes** → **Select the volume** → **Action** → **Increase capacity**.

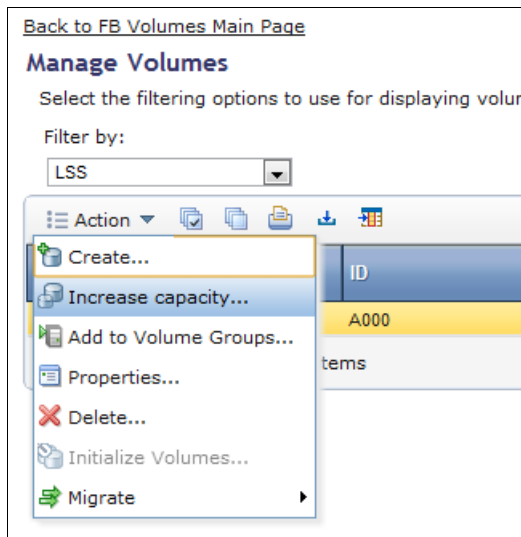


Figure 7-18 Increase volume size by GUI

In the pop-up window shown in Figure 7-18, enter the new size and click **OK**.

- b. Using the DS CLI:

Use the DS CLI command **chfbvol** as in Example 7-3 and specify the new capacity and the volume ID.

### Example 7-3 increase capacity using chfbvol

```
dscli> chfbvol -cap 40 A000
Date/Time: 31. Mai 2012 13:33:37 CEST IBM DSCLI Version: 7.6.20.221 DS:
IBM.2107-75TV181
CMUC00332W chfbvol: Some host operating systems do not support changing the volume
size. Are you sure that you want to resize the volume? [y/n]: y
CMUC00026I chfbvol: FB volume A000 successfully modified.
```

2. Increase the size of the VMFS datastore in vSphere:
  - a. Go to **Configuration** → **Storage** and click **Rescan All...**
  - b. Next, select the datastore. Right-click and select **Properties**. See Figure 7-19 for an example.

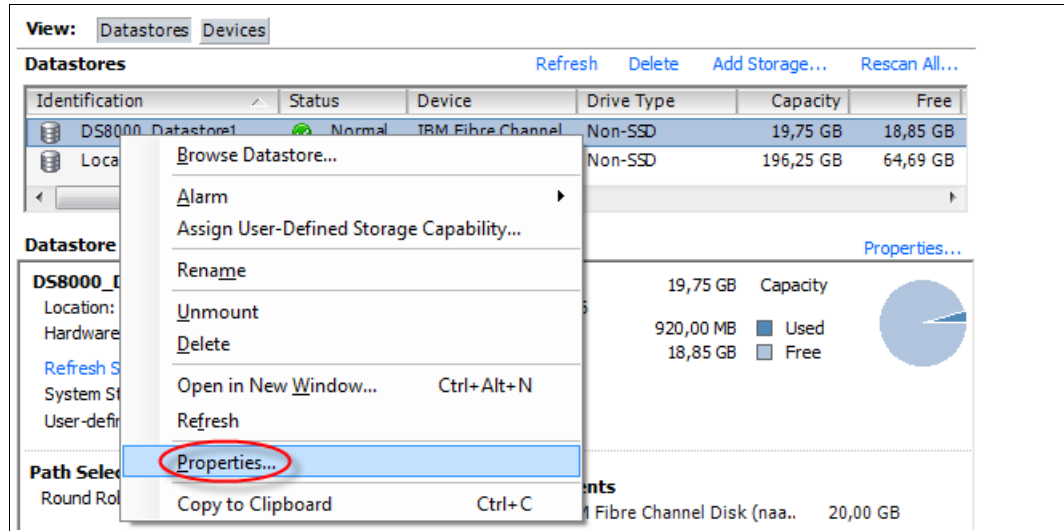


Figure 7-19 Datastore properties

- c. In the volume properties window shown in Figure 7-20, click **Increase**.

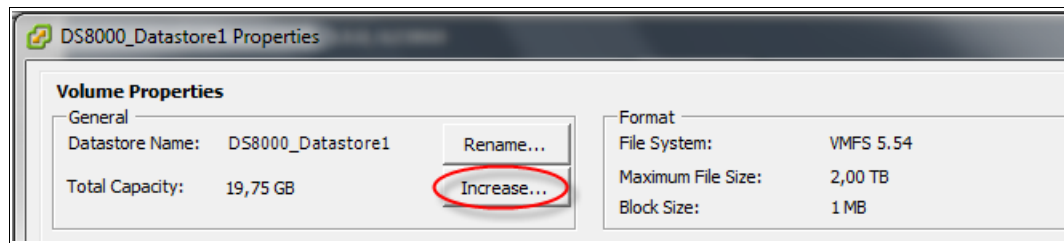


Figure 7-20 Volume properties

- d. Select the LUN (notice the new size as shown in Figure 7-21).

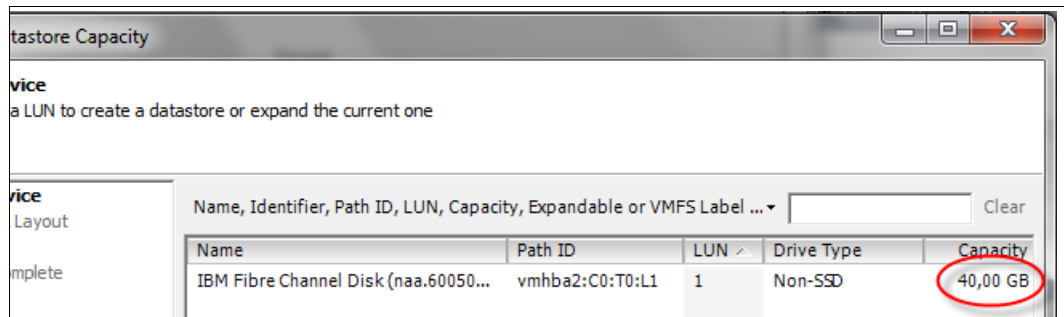


Figure 7-21 Select LUN

- e. Review the current disk layout as in Figure 7-22 and ensure that it only contains free space besides the already existing VMFS partition.

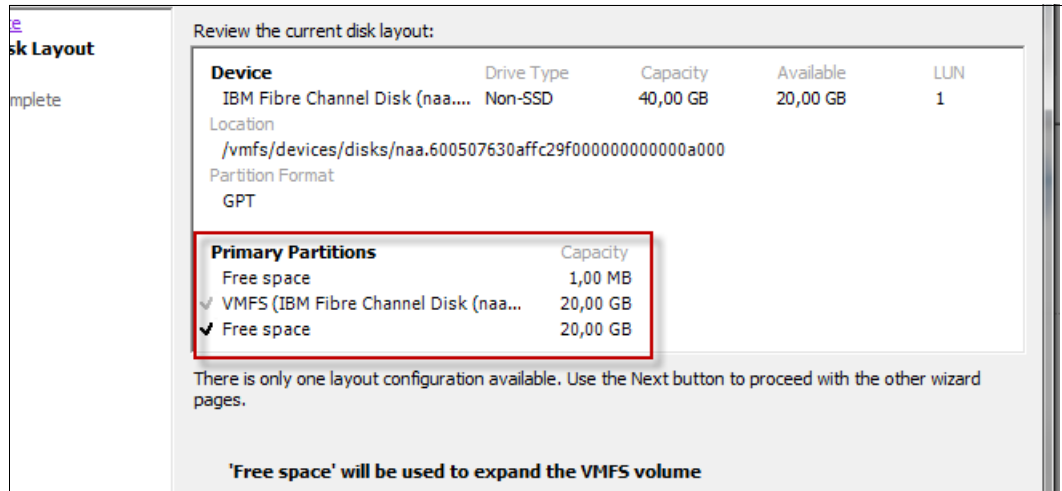


Figure 7-22 Disk layout

- f. Select all of the available capacity to be added to the datastore (Figure 7-23).

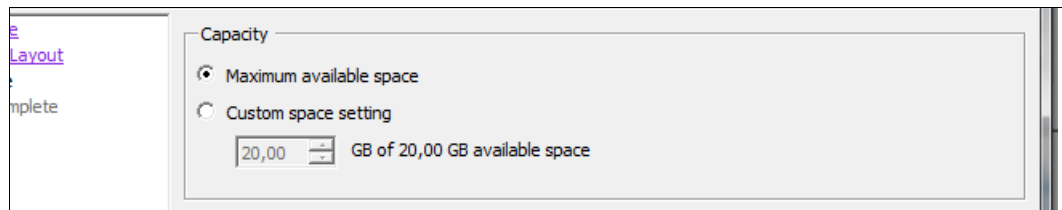


Figure 7-23 Capacity to add

- g. Review the summary page as in Figure 7-24 and click **Finish**.

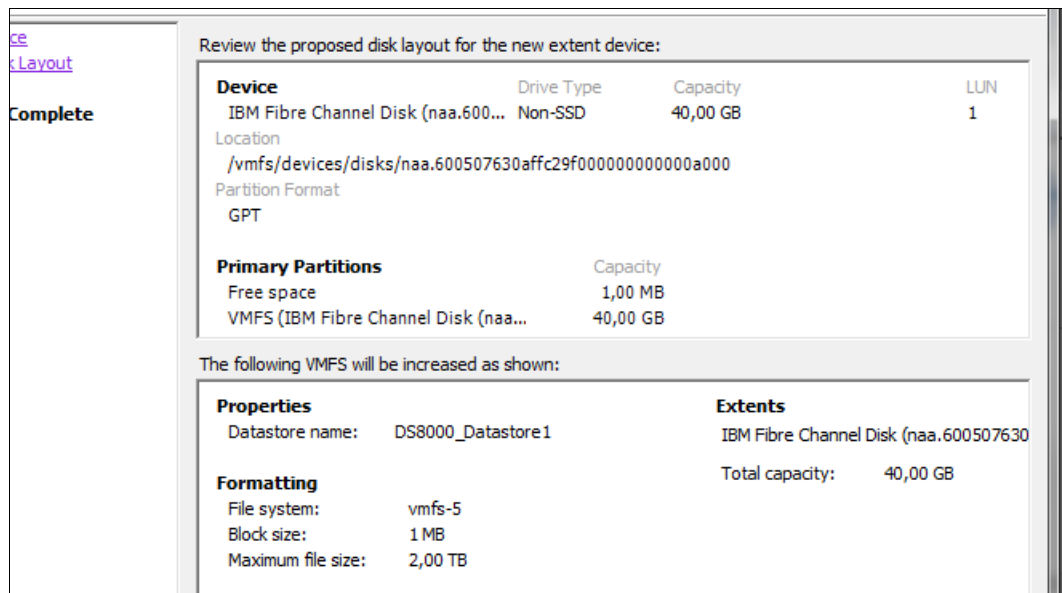


Figure 7-24 Summary page

- h. Close the **Datastore Properties** window.
3. Verify, from the Storage view, that the datastore has successfully been expanded.

## 7.2.2 Virtual disks

The most common concept used with ESXi are virtual disks.

### Virtual disk concepts

From an ESXi view, virtual disks are large files that represent the virtual hard drives. Due to the maximum file size of 2 TB on VMFS, a virtual disk can have a maximum size of 2 TB.

Inside a VM, a virtual disk looks like a SCSI hard drive. All SCSI I/O commands issued by the operating system inside the VM are converted into file I/O operations by the hypervisor. This processing is transparent to the operating system.

### Assigning a virtual disk to a VM

To assign a virtual disk to a VM, perform the following steps:

1. Select the VM from the list, right-click, and select **Edit Settings** as in Figure 7-25.

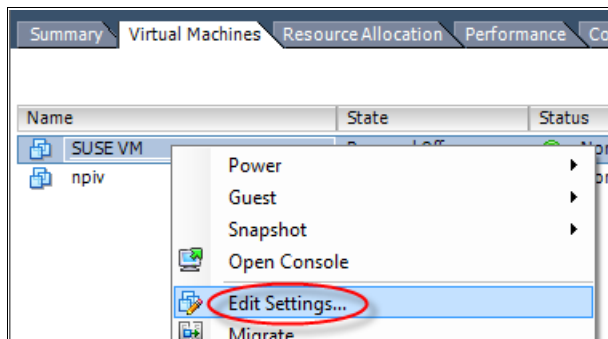


Figure 7-25 Edit Settings of VM

2. Click **Add** (Figure 7-26).

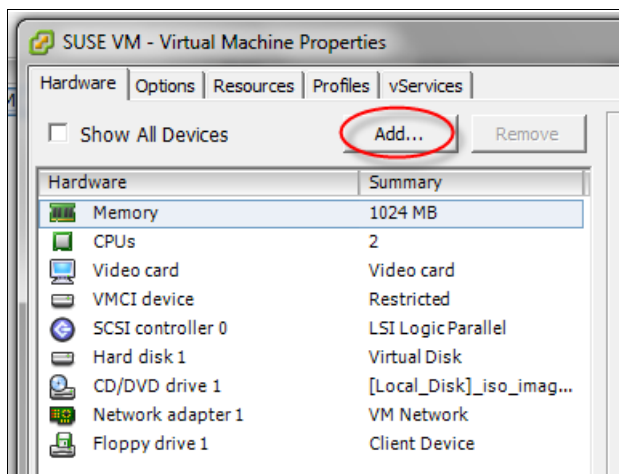


Figure 7-26 Add device to VM

3. From the list of devices in Figure 7-27, select **Hard Disk**.

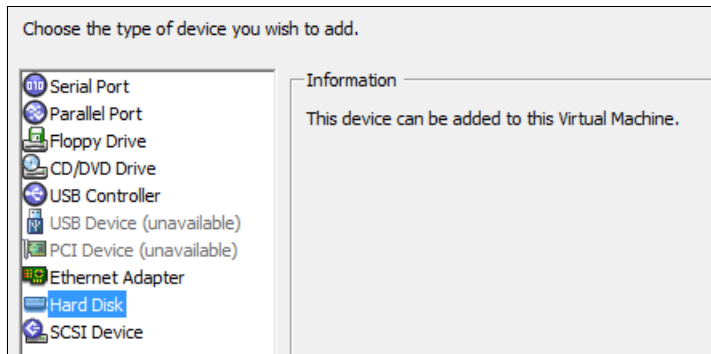


Figure 7-27 Add Hard Disk

4. Select **Create a new virtual disk** as shown in Figure 7-28.

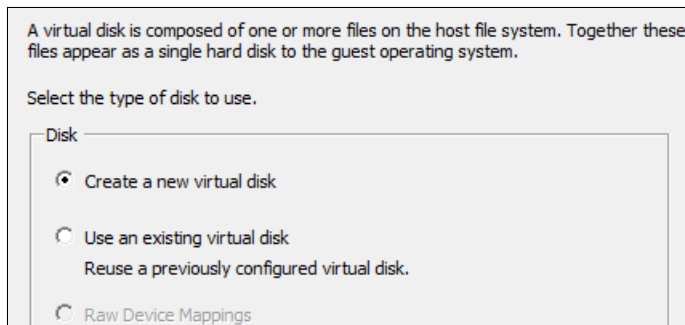


Figure 7-28 Create virtual disk

**Tip:** Select **Use an existing virtual disk** if you want to share virtual disks between VMs or to assign an existing virtual disk.

5. Define the properties of the new virtual disk as shown in Figure 7-29:

a. Disk Size:

Size of the new virtual disk. It is the size that the operating system will see.

b. Disk Provisioning:

- Thick Provision Lazy Zeroed:

The full capacity is allocated on the datastore on creation. The blocks are not overwritten by zeros, the VM might be able to read any data that was previously stored on these blocks. The virtual disk is available almost instantly.

- Thick Provision Eager Zeroed:

The full capacity is allocated on the datastore on creation. The blocks are all overwritten by zeros to delete anything that was previously stored on these blocks. This method requires the most time, but ensures that a VM cannot read any old information from the blocks.

- Thin Provision:

The capacity is allocated on demand when the VMs operating system tries to access it. This method is saving physical space on the datastore and allows for overcommitment.



c. Storage location of the new virtual disk:

- Store with the virtual machine:

The new virtual disk will be stored on the same datastore as the rest of the virtual machine.

- Specify a datastore:

The new virtual disk will be stored on a different datastore than the rest of the virtual machine. Possible reasons could be that the original datastore does not have enough capacity or a different performance characteristic is required for the new virtual disk.

The screenshot shows a dialog box for configuring virtual disk parameters. It is divided into three sections: Capacity, Disk Provisioning, and Location. In the Capacity section, the Disk Size is set to 10 GB. In the Disk Provisioning section, the 'Thick Provision Lazy Zeroed' option is selected. In the Location section, the 'Store with the virtual machine' option is selected. Below this, there is a radio button for 'Specify a datastore or datastore cluster:' with an empty text field and a 'Browse...' button next to it.

Figure 7-29 virtual disk parameters

6. In the advanced options page, the default values fit in most usage scenarios. Figure 7-30 shows an example with default values:

- Virtual Device Node:

This node specifies where in the virtual SCSI bus of the VM the new virtual disk should be attached.

- Mode:

This mode specifies whether the virtual disk is included in snapshots. If *Independent* is selected, the virtual disk will not be included in any snapshots. For independent disks, there are two options:

- Persistent:

All data is written directly to the disk and saved across power cycles.

- Nonpersistent:

All data written to this disk is temporarily saved and lost after a power cycle.

Specify the advanced options for this virtual disk. These options do not normally need to be changed.

Virtual Device Node

SCSI (0:1) ▼

IDE (0:0) ▼

Mode

Independent  
Independent disks are not affected by snapshots.

Persistent  
Changes are immediately and permanently written to the disk.

Nonpersistent  
Changes to this disk are discarded when you power off or revert to the snapshot.

Figure 7-30 Advanced disk options

7. Verify all options in the Ready to Complete window and click **Finish** (Figure 7-31).

Options:

Hardware type:	Hard Disk
Create disk:	New virtual disk
Disk capacity:	10 GB
Disk provisioning:	Thick Provision Lazy Zeroed
Datastore:	Local_Disk
Virtual Device Node:	SCSI (0:1)
Disk mode:	Persistent

Figure 7-31 Ready to Complete

8. Close the Virtual Machine Properties window.

### 7.2.3 Raw device mappings (RDMS)

Raw device mappings (RDMS) allow a VM machine to store its data directly on a logical volume instead of writing to a virtual disk file. A mapping file stored on the VMFS contains all information the VM requires to access the logical volume.

**Tip:** RDMS can have a maximum size of 64 TB.

RDMS allow for SAN-aware applications on the VM and provide the possibility to use N Port ID Virtualization (NPIV) on a virtual machine level. See “Using NPIV with RDMS” on page 141 for details on NPIV.

## Assigning an RDM to a VM

The following steps describe how to assign a physical LUN to a VM:

1. Select the VM from the list, right-click, and select **Edit Settings**.
2. Click **Add**.
3. From the list of devices, select **Hard Disk**.
4. Select **Raw Device Mappings** as shown in Figure 7-32.

A virtual disk is composed of one or more files on the host file system. Together these files appear as a single hard disk to the guest operating system.

Select the type of disk to use.

Disk

Create a new virtual disk

Use an existing virtual disk  
Reuse a previously configured virtual disk.

Raw Device Mappings  
Give your virtual machine direct access to SAN. This option allows you to use existing SAN commands to manage the storage and continue to access it using a datastore.

Figure 7-32 Raw Device Mapping

5. Select the LUN to use as RDM from the list as shown in Figure 7-33.

Name, Identifier, Path ID, LUN or Capacity contains:  Clear

Name	Path ID	LUN	Capacity	Hard
IBM Fibre Channel Disk (naa.60050...	vmhba2:C0:T0:L2	2	25,00 GB	Unk

Figure 7-33 Select LUN for the RDM

6. Select where to store the mapping file for the RDM as shown in Figure 7-34.

Select the datastore on which to store the LUN mapping. You will use the disk map on this datastore to access the virtual disk.

Store with Virtual Machine

Specify datastore

Datastore	# Hosts	Datastore Cluster
Local_Disk	1	N/A
DS8000_Datastore1	1	N/A

Figure 7-34 Datastore for the RDM mapping file

**Tip:** Keep all files of a VM together in one folder for easier backup and management.

7. Select Compatibility Mode from the available options as shown in Figure 7-35.

– Physical:

The RDM will not be affected by snapshots, similar to a virtual disk in mode **independent, persistent**.

**Tip:** Use this mode if the LUN also needs to be used by a physical server (such as for a cluster).

– Virtual:

The RDM can be included in snapshots and can benefit from other virtual disk features.

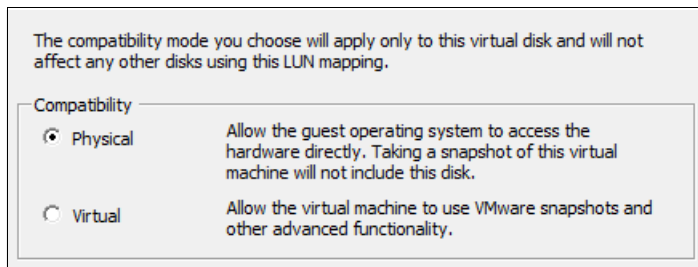


Figure 7-35 Compatibility mode for RDM

8. In the advanced options in Figure 7-36 the virtual device node can be specified.

It specifies where in the virtual SCSI bus of the VM the RDM should be attached.

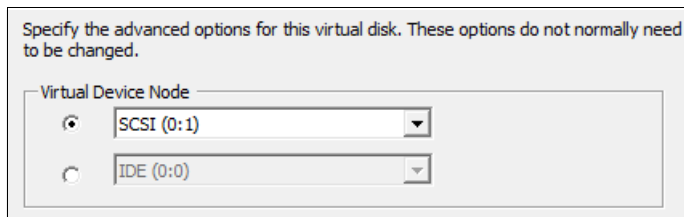


Figure 7-36 virtual device node for RDM

9. Verify all options in the Ready to Complete window and click **Finish**.

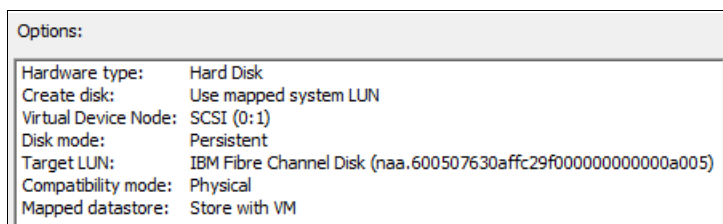


Figure 7-37 Ready to Complete window for RDM

10. Close the Virtual Machine Properties window.

## Using NPIV with RDMs

More information about NPIV can be found in the following VMware publications:

[http://www.vmware.com/files/pdf/techpaper/vsp\\_4\\_vsp4\\_41\\_npivconfig.pdf](http://www.vmware.com/files/pdf/techpaper/vsp_4_vsp4_41_npivconfig.pdf)

**Attention:** NPIV can only be used with RDMs.

For more granular QoS measurement and monitoring, a virtual WWN can be created for a VM. All I/O from this VM will be done using this virtual WWN.

To use NPIV with a virtual machine, perform the following steps:

**Important:** The virtual machine needs to be powered off for these steps.

1. Create an RDM as described in “Assigning an RDM to a VM” on page 139.
2. In the Virtual Machine Properties window of the VM, select the **Options** tab and click **Fibre Channel NPIV** as shown in Figure 7-38.

Remove the checkmark from **Temporarily Disable...**, select **Generate new WWNs**, and choose the amounts that you want.

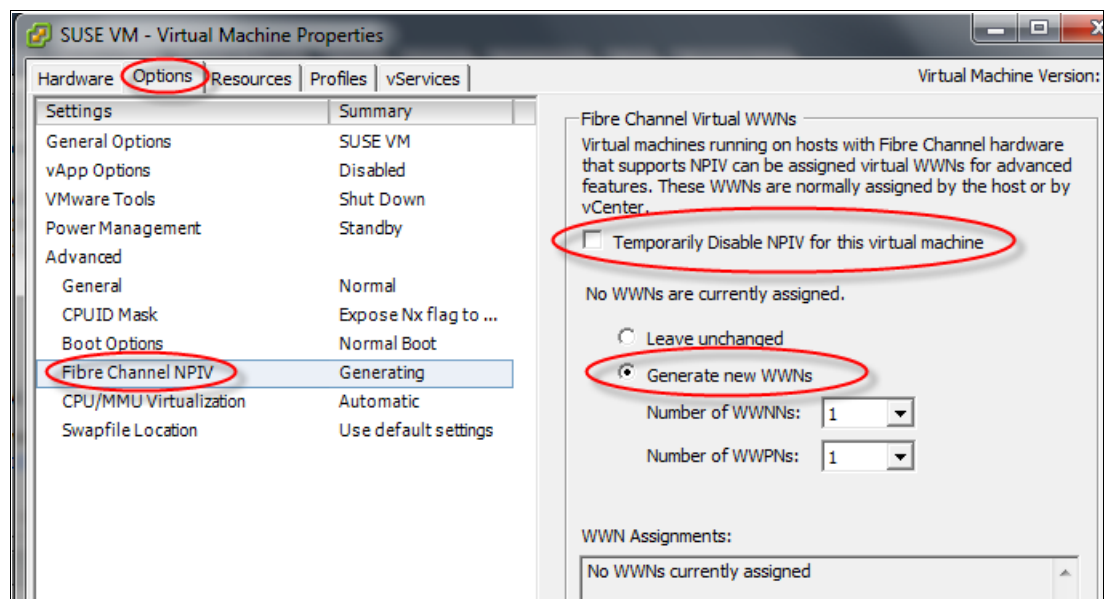


Figure 7-38 NPIV options tab

3. Click **OK** to close the Virtual Machine Properties window.  
The WWNs are now created.
4. Reopen the Virtual Machine Properties, select the **Options** tab, click **Fibre Channel NPIV**, and note the WWNs for the VM as shown in Figure 7-39.

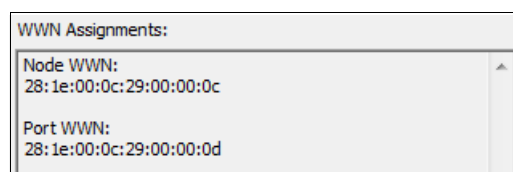


Figure 7-39 WWN assignments of a VM

5. Create zoning for the virtual machines WWPN(s).
6. Create host connection and volume group for the virtual machines WWPN(s) using hosttype **VMware**.

**Important:**

- ▶ The RDM volume needs to be mapped to all ESXi WWPNs that can access the DS8000 and in addition to the virtual machines WWPN(s).
- ▶ If separate volume groups are used for the ESXi host and the virtual machine, the RDM volume must have the same SCSI ID in both volume groups!
- ▶ See “Fixed LUN ID for ESXi volumes” on page 130 for details.

7. When the VM is powered on, it will log in to the DS8000 with its virtual WWPN.

To check, use the DS CLI command **lshostconnect -login**. See Example 7-4.

*Example 7-4 VM logged in with NPIV*

```
dsccli> lshostconnect -login
Date/Time: 1. Juni 2012 11:58:03 CEST IBM DSCLI Version: 7.6.20.221 DS:75-TV181
WWNN          WWPN          ESSIOport LoginType Name          ID
=====
281E000C2900000C 281E000C2900000D I0001      SCSI      ITS0-ESXi-NPIV      0075
281E000C2900000C 281E000C2900000D I0005      SCSI      ITS0-ESXi-NPIV      0075
```

**Tip:** If the VM is not able to access the RDM using its virtual WWN, it will access the LUN using the ESXi hosts WWN (as if NPIV was not configured).

## 7.3 Multipathing

The following sections provide an insight into the way VMware handles multiple paths to a storage device.

### 7.3.1 Pluggable Storage Architecture

For multipathing ESXi uses a concept called Pluggable Storage Architecture (PSA). As shown in Figure 7-40, it consists of three levels of plug-ins:

- ▶ **Native Multipathing (NMP):**  
This plug-in chooses where to send I/O requests and handles the different storage devices connected to the ESXi host.
- ▶ **Storage Array Type Plug-in (SATP):**  
This plug-in is responsible for array specific operations such as path monitoring and activating. SATP is chosen by NMP based on the type of storage device detected. The user is able to change the SATP by **esxc1i** commands.
- ▶ **Path Selection Plug-in (PSP):**  
This plug-in is responsible for choosing the physical path for an I/O request. NMP assigns PSP to SATP, but the user can change the PSP concurrently by the GUI.

**Tip:** The PSP can be chosen individually for each LUN.

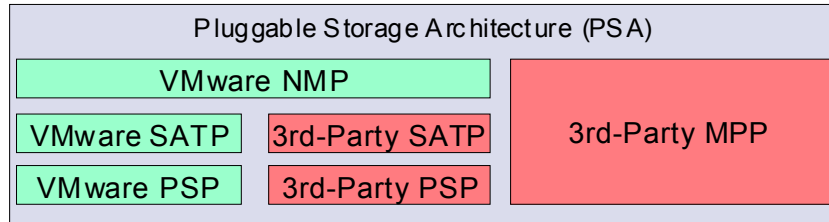


Figure 7-40 ESXi Pluggable Storage Architecture (PSA)

ESXi contains default plug-ins for most storage devices, but vendors can provide their own plug-ins to fully utilize any special features of their storage devices.

While the SATP should not be changed from what ESXi has chosen, all three of the following PSPs are supported for the DS8000:

- ▶ Fixed:
 

All I/O is always routed over one of the available paths. The user can set one path as the preferred one which is always used. If this path fails, a failover to one of the remaining paths is done. If the preferred path returns, a failback is done. With PSP Fixed, a manual load balancing is possible by choosing different paths for each LUN.
- ▶ Most Recently Used (MRU):
 

Similar as for Fixed, all I/O is always routed over one of the available paths. In case of a failure, a failover to another path is done. But if the initial path is recovered, no failback will happen. Manual load balancing is not possible for this. This PSP has initially been developed for active-passive arrays.
- ▶ Round Robin (RR):
 

This PSP is the only one that evenly uses all available paths. I/O is routed over all active paths alternating. If one path fails it is skipped. After its recovery, it is used for I/O again automatically.

### 7.3.2 Path naming

ESXi uses a special scheme to name the paths. Each path to a LUN gets a runtime name according to the scheme in Figure 7-41.

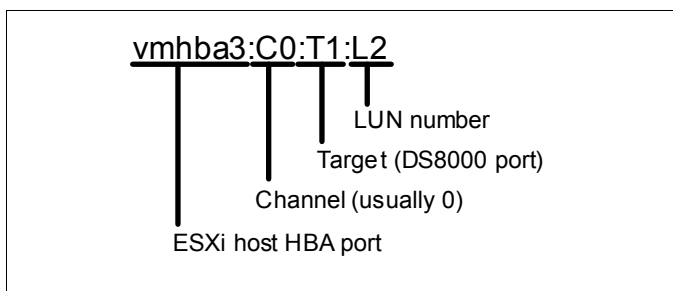


Figure 7-41 Path naming scheme

Each port is specified by four parameters:

- ▶ ESXi host HBA port: The Fibre Channel adapter port on the ESXi host
- ▶ Channel: The storage channel
- ▶ Target: The DS8000 host adapter port that the path is connected to
- ▶ LUN number: The LUN from the DS8000

The path list for a volume can be viewed by right-clicking the volume in the **Storage Adapter** view and selecting **Manage Paths...**

Paths			
Runtime Name	Target	LUN	Status
vmhba5:C0:T0:L0	50:05:07:63:0a:ff:c2:9f 50:05:07:63:0a:00:42:9f	0	◆ Active (I/O)
vmhba5:C0:T1:L0	50:05:07:63:0a:ff:c2:9f 50:05:07:63:0a:40:42:9f	0	◆ Active (I/O)
vmhba4:C0:T1:L0	50:05:07:63:0a:ff:c2:9f 50:05:07:63:0a:40:42:9f	0	◆ Active (I/O)
vmhba4:C0:T0:L0	50:05:07:63:0a:ff:c2:9f 50:05:07:63:0a:00:42:9f	0	◆ Active (I/O)

Figure 7-42 Path information

In the example in Figure 7-42, the volume has four paths from two HBA ports (**vmhba4** and **vmhba5**) to two different DS8000 ports (**T0** and **T1**).

The Target column provides information about the WWNN and WWPN of the storage device.

### 7.3.3 Multipathing and DS8000

IBM does not provide a specific multipathing plug-in for DS8000. Therefore, ESXi automatically uses the following default plug-ins:

► **SATP:**

**VMW\_SATP\_ALUA** is chosen because the DS8000 is an Asymmetric Logical Unit Access (ALUA) capable storage array.

► **PSP:**

The default PSP for **VMW\_SATP\_ALUA** is **MRU** (most recently used).

**Tip:** The interoperability matrix at [vmware.com](http://vmware.com) will list “Fixed” as default PSP for DS8000. However, the code will automatically choose “MRU”. Both PSP values are supported for DS8000.



In order to better utilize all available paths to the DS8000, users should switch to **RR** (round robin). To do it, select the ESXi host and perform the following steps:

1. Go to **Configuration** → **Storage Adapters**, select one of the Fibre Channel HBAs connected to the DS8000, right-click one of the LUNs listed, and select **Manage Paths** as outlined in Figure 7-43.

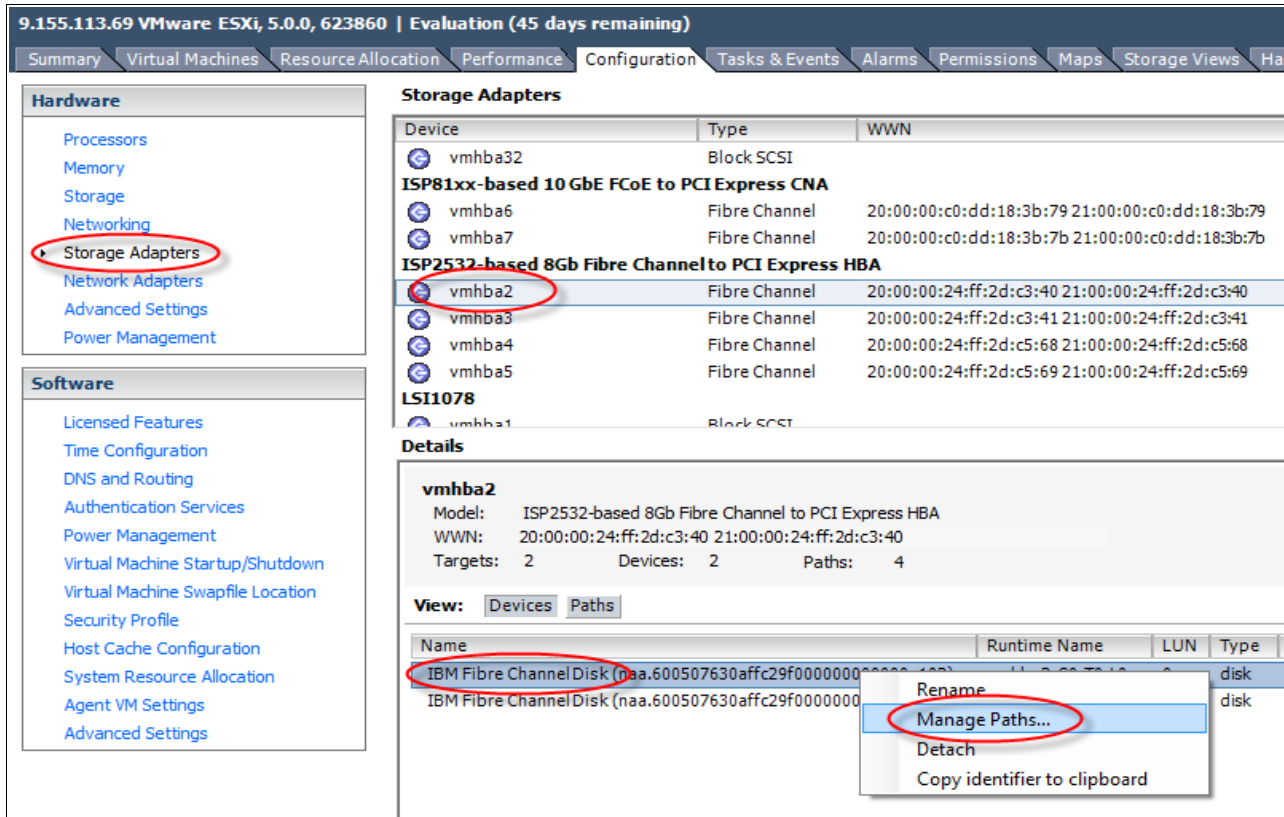


Figure 7-43 Show LUNs attached to ESXi host

2. From the drop-down list at the top, select **Round Robin (VMware)** and click **Change** (Figure 7-44).

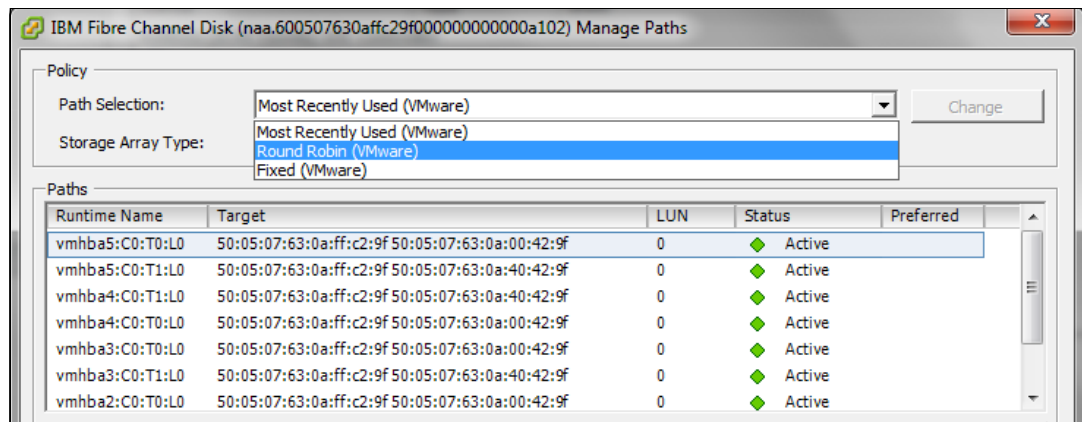


Figure 7-44 Change path selection plug-in

3. The status field for the different paths should all change to **Active (I/O)**.

## 7.4 Storage vMotion

vSphere has a built in feature called “Storage vMotion”. It provides the ability to “live migrate” virtual machines from one datastore to another without interruption. Storage vMotion works independently from the underlying storage and is initiated by the GUI.

This section presents the following usage scenarios:

- ▶ Migrating VMs from one storage device to another (for example, technology change)
- ▶ Evacuating a storage device for maintenance
- ▶ Load balancing datastore usage

### 7.4.1 Steps to migrate a VM from one datastore to another

To live migrate a VM onto another datastore, perform the following steps:

1. Select the VM from the list.
2. Right-click the VM and select **Migrate** as shown in Figure 7-45.

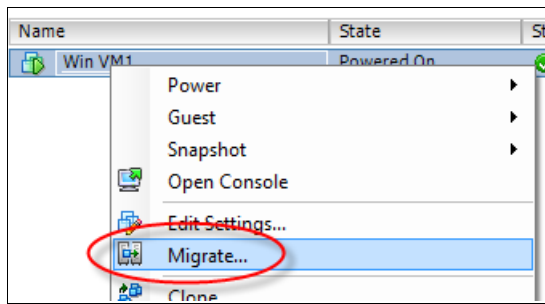


Figure 7-45 Start VM migration

3. In the wizard, select **Change Datastore** and click **Next** (Figure 7-46).

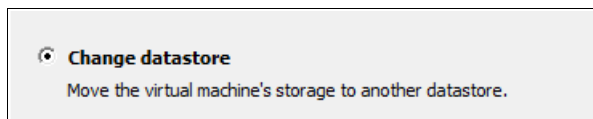


Figure 7-46 Change datastore

4. Select the target datastore from the list and click **Next**.
5. In the Ready to Complete window, click **Finish**.

The duration of the migration depends on several factors such as workload, size of VM, number of available paths, storage device, and so on.

### 7.4.2 Limitations

Storage vMotions require a host to read all data from the VM on one datastore and write the data to another. This generates additional load on the storage adapters and paths which need to be taken into consideration when performing Storage vMotions.

vSphere limits the amount of parallel Storage vMotion operations as listed in Table 7-1.

Table 7-1 Storage vMotion limits

Operation	Limit
Concurrent Storage vMotion operations per datastore	8
Concurrent Storage vMotion operations per host	2

## 7.5 Using SSD volumes as cache

ESXi allows for overcommitment of resources including memory. In times of high memory usage by the VMs, the ESXi host could run out of physical memory and the hypervisor needs to swap out memory pages. These memory pages are stored on a swapfile residing on physical disks or SAN volumes.

ESXi can detect volumes that reside on SSD ranks automatically. Such volumes can be used as cache by the ESXi host. Figure 7-47 shows how such volumes are identified in the **Storage Adapters** view.

Details							
vmhba5							
Model: ISP2532-based 8Gb Fibre Channel to PCI Express HBA							
WWN: 20:00:00:24:ff:2d:c5:69 21:00:00:24:ff:2d:c5:69							
Targets: 2      Devices: 4      Paths: 8							
View: <b>Devices</b> Paths							
Name	Runtime Name	LUN	Type	Drive Type	Transport	Capacity	Owner
IBM Fibre Channel Disk (naa.6005076...	vmhba2:C0:T0:L0	0	disk	Non-SSD	Fibre Chann...	5,00 GB	NMP
IBM Fibre Channel Disk (naa.6005076...	vmhba2:C0:T0:L1	1	disk	Non-SSD	Fibre Chann...	40,00 GB	NMP
IBM Fibre Channel Disk (naa.6005076...	vmhba2:C0:T0:L2	2	disk	Non-SSD	Fibre Chann...	25,00 GB	NMP
IBM Fibre Channel Disk (naa.6005076...	vmhba2:C0:T0:L3	3	disk	SSD	Fibre Chann...	100,00 GB	NMP

Figure 7-47 SSD detection

To use such a volume as host cache, first a VMFS datastore must be created on the volume. “Creating a VMFS datastore” on page 126 describes the required steps.

In the **Configuration** tab, select **Host Cache Configuration** as shown in Figure 7-48.

Host Cache Configuration				
SSD Datastores				
Refresh    Add Storage...    Rescan All...    Properties...				
Identification	Capacity	Host Cache Space	Free Space	
DS8000_SDD_DS1	99,75 GB	0,00 B	98,80 GB	

Figure 7-48 SSD datastore list

Select the SSD datastore from the list and click **Properties**.

Set the check mark at **Allocate space for host cache** and choose how much space to use as cache, as shown in Figure 7-49.

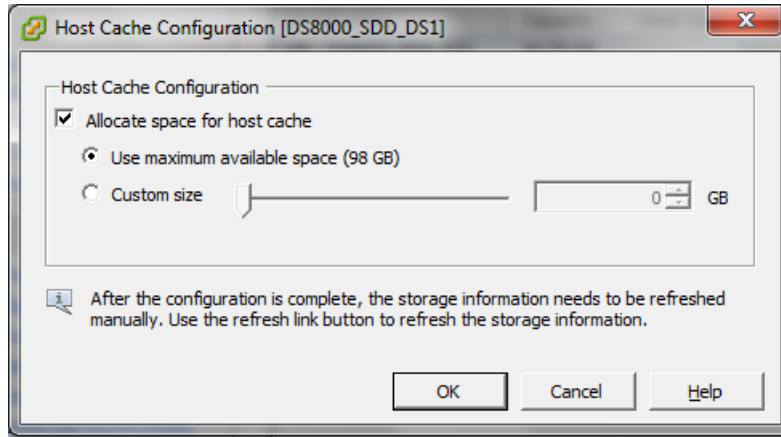


Figure 7-49 Host cache configuration

Click **OK** when done.

**Tip:** Space not used as host cache can still be used to store VMs.

## 7.6 Best practices

This section covers best practices for datastores and multipathing.

### 7.6.1 Datastores

It is advisable to have a one-to-one relationship between LUNs and datastores:

- ▶ One LUN should only contain one datastore
- ▶ One datastore should not be spread over more than one LUN

There is no general advice possible regarding number and size of datastores. The matrix in Table 7-2 can help you decide what is best for the individual setup.

Table 7-2 Datastore decision matrix

Concept	Pro:	Con:
Few, large datastores	<ul style="list-style-type: none"> <li>▶ Less administrative work</li> <li>▶ Better scalability</li> </ul>	<ul style="list-style-type: none"> <li>▶ VMs on one datastore can influence each other</li> </ul>
More, small datastores	<ul style="list-style-type: none"> <li>▶ Fewer VMs per datastore</li> <li>▶ Better workload isolation</li> <li>▶ Use of RDMS allows for NPIV</li> </ul>	<ul style="list-style-type: none"> <li>▶ Architectural limits</li> <li>▶ Administrative overhead</li> <li>▶ Need to spread VMs over multiple datastores</li> </ul>

## 7.6.2 Multipathing

To utilize all available paths, the PSP needs to be set to **Round Robin (RR)** as described in 7.3, “Multipathing” on page 142.

While there is no general advice on the number of paths to use, various configuration maximums need to be observed to ensure scalability in the future. See Table 7-3.

*Table 7-3 Configuration Maximums*

<b>Item</b>	<b>Maximum</b>
Number of paths to a LUN	32
Number of total paths per ESXi host	1024
Number of LUNs per ESXi host	256

If each LUN has four paths to the storage device, the maximum number of 256 LUNs is possible. With an increasing number of paths, the maximum number of LUNs will decrease.

While a high number of paths per LUN increases reliability and throughput, the possible number of LUNs is reduced. RDMS will further decrease the number LUNs available for datastores.





## Apple considerations

This chapter provides information for the specifics of attaching IBM DS8000 storage system to host systems running Mac OS X.

The following topics are covered:

- ▶ Configuring the Apple host on a DS8000
- ▶ Installing the ATTO software
- ▶ Using the ATTO Configuration Tool
- ▶ Creating a file system on Apple Mac OS X
- ▶ Troubleshooting

**Attention:** At the time of writing this book, no 8Gb/s Host Bus Adapters (HBAs) were officially supported by the DS8800 storage system. The ATTO 8Gb HBA is awaiting certification by IBM and was used successfully in the examples in this chapter.

## 8.1 Available resources

For the latest available, supported Mac OS X configuration and required software patches, see the IBM SSIC at the following website:

<http://www.ibm.com/systems/support/storage/ssic/interoperability.wss>

For information to prepare the host to attach the DS8000 storage system, see the DS8000 Information Center at the following website:

<http://publib.boulder.ibm.com/infocenter/dsichelp/ds8000ic/index.jsp>

In the Information Center, select **Configuring** → **Attaching Hosts** → **Apple Macintosh host attachment**.

For information regarding the ATTO-HBA, see the Product Information pages for ATTO Technology, Inc. at the following vendor website.

**Vendor Website:** <http://www.attotech.com>

## 8.2 Configuring the Apple host on a DS8000

To configure a server on the DS8000, define a host connection using the DS graphical user interface (GUI) or the DS command line interface (CLI). On the DS8000 storage system, use the predefined host type **Apple OS X** to automatically configure the volumes in an Mac OS X preferred method.

For general information, see the *IBM System Storage DS8000 Host Systems Attachment Guide*, GC27-2298-02.

**Tip:** A restart of the Apple server is required to detect any newly created LUNs.

The LUN-numbering seen in the ATTO Configuration Tool is created by the software and it is set by the order that the LUNs appear during detection.

## 8.3 Installing the ATTO software

The following sections examine the installation of the ATTO driver software for OS X and the ATTO configuration tool. The procedure for updating the flash firmware on the ATTO card is also covered.

### 8.3.1 ATTO OS X driver installation

The DS8000-supported driver does not ship with ATTO products, it must be downloaded:

1. Download the supported “Mac OS X Multipath Director™ Driver” file from ATTO directly. Look for the driver for IBM DS Series from the *Fibre Channel HBAs for Storage Partners* page. On the HBA-Vendor webpage, click **Solutions** → **IBM solutions** → **DS Solutions** to get to the DS-specific main page. Get the needed installer(s) files by selecting your HBA-type and login to the download page. Again, you need to reference the SSIC page at IBM to find supported driver levels.



**Important:** The Multipath Director Driver software is not included with the Host Bus Adapter product, you need to download it from ATTO.

2. Install the HBA software. For more information, see the readme file that should be included. When the software driver is installed successfully and the final **OK** button is clicked, the server will need to be restarted.
3. Verify that the ATTO adapter driver module is loaded into memory using the host system **kextstat** command as shown in Example 8-1.

*Example 8-1 Using kextstat to verify driver installation*

---

```
macintosh-4:~ admin$ sudo kextstat | grep ATTO
 49  0 0x93a000  0x49000  0x48000  com.ATTO.driver.ATTOCelerityFC8
(1.4.4) <48 14 5 4 3>
```

---

### 8.3.2 ATTO Configuration Tool Installation and Flash Update

Use the following procedure:

1. Download the “Mac OS X Config Tool” file from the ATTO webpage as described under 8.3.1, “ATTO OS X driver installation”
2. Unpack the file and perform a full installation. For more information, see the readme file shipped with the driver.
3. The default path for installation is: /Applications/ATTO Configuration Tool
4. Launch the ATTO Configuration Tool (double-click the ATTO Configuration Tool located in the /Applications/ATTO Configuration Tool folder).
5. Check the installed *Celerity Flash* version to be sure it is the latest.

Expand **host** → **localhost** → select **Celerity FC-8xEN** → **Flash**

- If there is a more current version available from the ATTO website, download and unpack the new Celerity Flash contained in “Mac OS X Flash Bundle”.
  - Click **Browse** and select the current Celerity Flash bundle for your adapter card. The **Update** button should become available.
  - Click **Update** to overwrite the adapter flash.
  - A server reboot is required for this update to take effect.
6. Set each channel NVRAM parameter Port Down Retry Count to 8 in case there are more than two ports or channels. Set it to 30 if only one channel or port is available. Do this by choosing **host** → **localhost** → **Celerity FC-8xEN** → **Channel 1** → **NVRAM** Save and Commit the changes. Repeat these steps for all channels. Also make a note of the “Port Name” as it will be needed for switch-zoning and host creation on the DS8000.

**Tip:** Write down the “Port Name” for each Channel that you can see in the ATTO Configuration Tool. It will be needed in the configuration of the DS8000.

## 8.4 Using the ATTO Configuration Tool

The information contained in this section is based on using the GUI of the Mac OS X Server. The user should have access to the server console directly or through a VNC connection to use the ATTO Configuration Tool.

Launch the ATTO Configuration Tool by double-clicking the ATTO Configuration Tool icon located in the Applications folder. The ATTO Configuration Tool consists of three sections:

- Device Listing:** Lists all installed devices.
- Info/Configuration:** Show details of the device selected in the Device Listing section.
- Status:** Warnings, references are displayed.

To view the DS8000 storage, expand the tree in the *Device Listing* window on the left until you see your device (**2107**) and disks (**LUNs**).

### 8.4.1 Paths

The Paths tab in the multipathing window displays path information as shown in Figure 8-1.

The lower half of the window displays information regarding the Target, Host Bus Adapter and Statistics.

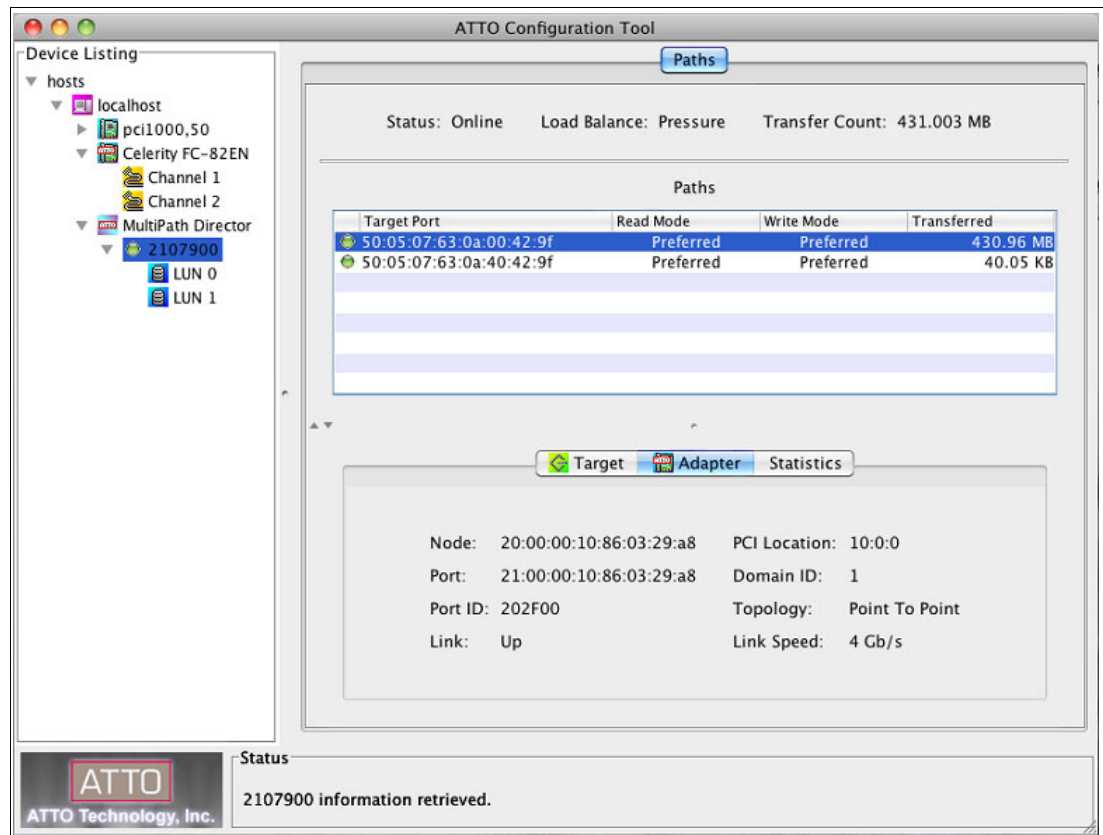


Figure 8-1 ATTO Configuration Tool: Paths

The path information displayed is based on a per Target or per LUN basis, depending on what device is selected in the Device Listing pane.

## 8.4.2 Target Base

Click the Target in the device tree (**2107900** in Figure 8-1 on page 154) to display the Target Multipathing window with all available paths to the target. The icon to the left of the device indicates the multipathing status for the target, as shown in Figure 8-2.

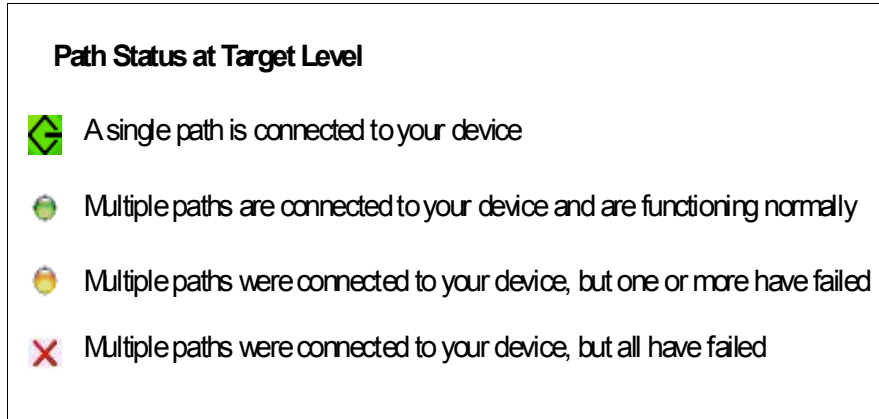


Figure 8-2 Path Status at the Target Level

## 8.4.3 LUN Base

Click a LUN (such as LUN 1 shown in Figure 8-1 on page 154) to display the LUN multipathing window. The icon to the left of the path shows the status, as shown in Figure 8-3.

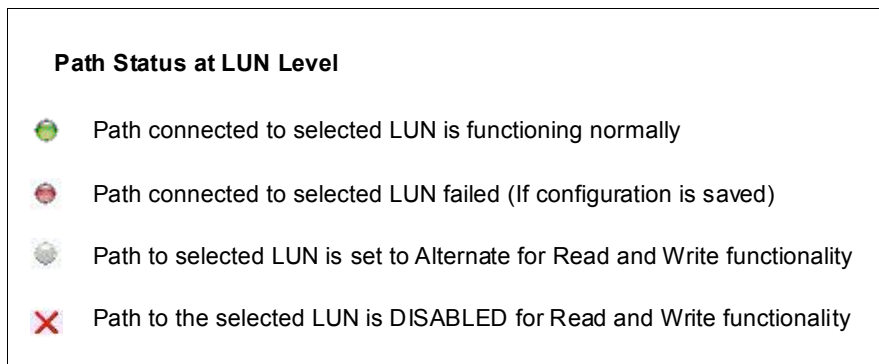


Figure 8-3 Path status at the LUN Level

## 8.4.4 Path Actions

With the pull-down menu, **Paths**, you have various options to modify path settings and statistics depending on the device that was selected in the Device Listing section. See Figure 8-4 for an example of the **Paths** menu.



Figure 8-4 Paths options

**Setup:** Select **Setup** if you need to change Load Balancing Policy or Path behavior. Load Balancing can be configured on a per LUN or per Target basis depending on which device is highlighted in the Device Listing pane. If a Target is selected, then all LUNs controlled by it will be configured the same way.

**Important:** If a LUN is selected in the Device Listing pane, then only that LUN will be affected by configuration changes. If a Target is selected, then *all* LUNs controlled by that Target will be affected by configuration changes.

There are three options for setting the method used for Load Balancing across multiple paths to the storage. The Load Balancing policies are as follows:

- ▶ Pressure: The path with the fewest bytes in the queue will be used if needed.
- ▶ Queue Depth: The path with the fewest commands in the queue will be used.
- ▶ Round Robin: The least used path is selected.

By clicking **Next**, you can modify the highlighted path for write and read mode.

**Save / Delete Configurations:** This option saves the configuration and will apply any changes made. If new LUNs were created, then you need to save the new configuration, otherwise the newly created LUN(s) are not visible.

**Locate:** Use this button to blink the particular FC adapter's channel LED.

**Reset Statistics:** Statistics are set to '0' (no data is preserved) when this button is pressed.

**Refresh:** If the connection status for a path changes, the Configuration Tool will automatically refresh the display. The Configuration Tool will periodically (~30 seconds) refresh the path statistics for all devices. Nevertheless, you can manually refresh the displayed information by choosing "refresh".

## 8.4.5 Host bus adapter configuration

Changes to the adapter configuration are made in the NVRAM tab of the ATTO Configuration Tool. In the *Device Listing* section, select the adapter channel, then select NVRAM at the top of the panel on the right. Here you can view and change the Frame Size, Connection Mode, Date Rate, and many other options. This panel also allows the user to reset all the fields to their default values.

To make NVRAM changes permanent, be sure to perform all three of the following steps:

1. Make the change on each channel as needed.
2. Click the **Commit** button.
3. Restart the Apple server.

See the *IBM System Storage DS8000 Host Systems Attachment Guide*, GC27-2298-02 for any special settings that might be needed.

## 8.5 Creating a file system on Apple Mac OS X

You can create a file system using either the GUI or the CLI of the Mac OS X Server. This section describes the use of both methods.

## 8.5.1 Using the GUI: Disk Utility

The **Disk Utility** is a standard part of Mac OS X. It can be found under **Utilities** or by clicking **initialize** when a pop-up window appears for a newly created LUN after detection.

From the left pane, locate the disk you want to use and click **Partition**. In the Volume Scheme select the number of partitions. Select the volume format and provide a volume name. After clicking **Apply**, you need to click **Partition** to start the format.

## 8.5.2 Using the CLI: diskutil

The `diskutil` command is used to partition, repair, or erase the volume. Use this command to list the attached disks to the host system as shown in Example 8-2.

*Example 8-2 Using diskutil*

---

```
macintosh-4:~ admin$ diskutil list
/dev/disk0
#:  
0:      GUID_partition_scheme      *82.0 GB  disk0  
1:      EFI                        209.7 MB  disk0s1  
2:      Apple_HFS HDD                34.0 GB   disk0s2  
3:      Apple_HFS HDD 2              47.5 GB   disk0s3
/dev/disk1
#:  
0:      GUID_partition_scheme      *128.8 GB  disk1  
1:      EFI                        209.7 MB  disk1s1  
2:      Apple_HFS DS8k02            128.5 GB   disk1s2
/dev/disk2
#:  
0:      GUID_partition_scheme      *107.4 GB  disk2  
1:      EFI                        209.7 MB  disk2s1  
2:      Apple_HFS DS8k01            107.0 GB   disk2s2
```

---

The system manpages (`man diskutil`) should be referenced for detailed information and examples of using the `diskutil` command.

## 8.6 Troubleshooting

The ATTO Technology, Inc webpage contains very useful information for troubleshooting.

**Website:** <http://attotech.com>

On the ATTO web site, click **support** → **Troubleshooting, Tips & FAQs** → **Celerity Fibre Channel Host Adapters** → **Troubleshooting Celerity Fibre Channel Host Adapters - Mac**

### 8.6.1 Useful Utilities on Apple Mac OSX

**Disk Utility:** OS X disk repair and formatting tool. Location: `/Applications/Utilities/DiskUtility`

<b>Console:</b>	Allows the user to see application logs in real-time. The ATTO Celerity FC Driver does not use the system log to report events. Location: /Applications/Utilities/Console
<b>Apple System Profiler:</b>	This System Utility shows useful system information. All displayed information can be saved in a file. Location: <b>Apple Menu → About This Mac → More Info</b>
<b>ATTO Configuration Tool:</b>	This tool assists in identifying a failed or inactive path.
<b>kextstat-cmd:</b>	Use the <b>kextstat</b> command to check which ATTO FC driver is loaded into memory. Run it in a terminal window as shown in Example 8-1 on page 153.

## 8.6.2 Troubleshooting checklist

The following items can be helpful if you need to report a problem to ATTO or IBM. This list is not exhaustive, but it helps to have this information handy when troubleshooting any adapter or connectivity problem.

- ▶ Computer Model:  
(see **Apple System Profiler → Hardware**)
- ▶ Operating System:  
(see **Apple System Profiler → Software**)
- ▶ OS Patch Level:  
(see **Apple System Profiler → Software**)
- ▶ PCI or PCIe slot # and type:  
(see **Apple System Profiler → Hardware → PCI Cards**)
- ▶ ATTO driver version (from the Configuration Tool):  
(see **ATTO Configuration Tool → localhost → Celerity FC-xxx → Basic Info**)
- ▶ List of all of the devices attached to the ATTO HBA:  
(see ATTO Configuration Tool or **Apple System Profiler → Hardware → Fibre Channel → select SCSI Target Device**)
- ▶ Did this configuration ever work?
- ▶ Is this a new error that just started happening, or is this an error that was around since the first use.
- ▶ Can the error be duplicated? Does the error occur sporadically/randomly, or can it be reproduced each and every time?
- ▶ Apple System Profiler Output (ASP)
- ▶ **ioreg** output  
(open terminal window to run command and redirect the output into a file)
- ▶ **kextstat | grep ATTO**  
(open terminal window to run command and redirect the output into a file)
- ▶ **java -version**  
(open terminal window to run command and redirect the output into a file)



## Solaris considerations

This chapter provides information about the specifics of attaching IBM System Storage DS8000 systems to host systems running Oracle Solaris.

The following topics are covered:

- ▶ Working with Oracle Solaris and DS8000
- ▶ Locating the WWPNs of your HBAs
- ▶ Attaching Solaris to DS8000
- ▶ Multipathing in Solaris
- ▶ Expanding dynamic volume with VxVM and DMP
- ▶ Booting from SAN

## 9.1 Working with Oracle Solaris and DS8000

As with the previous models, the IBM System Storage DS8000 continues to provide extensive support for the Solaris operating systems. Currently, the DS8000 supports Solaris 8, 9, and 10, on a variety of platforms. It also supports the VERITAS Cluster Server and the Oracle Solaris Cluster. The IBM SSIC provides information about supported configurations, including information about supported HBAs, SAN switches, and multipathing technologies.

The IBM SSIC website can be found at this website:

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

Useful information about setting parameters of specific HBAs is also available in the IBM System Storage DS8000 Host Systems Attachment Guide GC27-2298-02 at this website:

<http://www-304.ibm.com/support/docview.wss?uid=ssg1S7001161&aid=1>

**Sun operating system:** Oracle has acquired Sun. For this reason, the operating system that was known as *Sun Solaris* was officially renamed as *Oracle Solaris*. Likewise, the *Sun Cluster*, also known as the *Solaris Cluster*, is now called the *Oracle Solaris Cluster*.

Before Solaris can access LUNs on a DS8000, some prerequisites need to be met:

- ▶ The Solaris server is equipped with a Fibre Channel host bus adapter (HBAs). For redundancy, it is best to have two HBAs.
- ▶ The Solaris server is physically connected with Fibre Channel cabling to a SAN, to which the DS8000 is also attached.
- ▶ SAN zoning is configured to allow the server HBAs to communicate with the DS8000 host adapters.

When this preparation work is complete, you can configure the LUNs in the DS8000 to be accessed by the Solaris server. It is described in the following chapters.

## 9.2 Locating the WWPNS of your HBAs

For a server to access LUNs configured in a DS8000 storage system, these LUNs need to be mapped to the server HBA's WWPNS. An object in the DS8000 is created for each WWPNS, called **hostconnection**. One or more **hostconnections** can be mapped to one or more LUNs in the DS8000 by adding those LUNs to an object called a **volume group**. The mapping in the DS8000 is managed by connecting a **hostconnection** to a volume group containing the LUNs.

In order to create these **hostconnections**, identify the WWPNS of the server's HBAs. One popular method of locating the WWPNS is to scan the `/var/adm/messages` file. Often, the WWPNS only shows up in the file after a restart. Also, the string to search for depends on the type of HBA that you have. Specific details are available in the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917.

In many cases, you can also use the **prtconf** command to list the WWPNS (Example 9-1).

*Example 9-1 Listing the WWPNS*

---

```
# prtconf -vp | grep port-wwn
port-wwn: 21000003.ba43fdc1
port-wwn: 210000e0.8b099408
port-wwn: 210000e0.8b0995f3
port-wwn: 210000e0.8b096cf6
port-wwn: 210000e0.8b098f08
```

---



With Solaris 10, the `fcinfo` command provides the ability to find the WWPN immediately, as shown in Example 9-2.

*Example 9-2 Using fcinfo to find the WWPNs*

---

```
# fcinfo hba-port
HBA Port WWN: 10000000c95f3592
    OS Device Name: /dev/cfg/c4
    Manufacturer: Emulex
    Model: LP10000
    Firmware Version: 1.92a1
    FCode/BIOS Version: none
    Type: L-port
    State: offline
    Supported Speeds: 1Gb 2Gb
    Current Speed: not established
    Node WWN: 20000000c95f3592
HBA Port WWN: 10000000c95f35dc
    OS Device Name: /dev/cfg/c3
    Manufacturer: Emulex
    Model: LP10000
    Firmware Version: 1.92a1
    FCode/BIOS Version: none
    Type: unknown
    State: offline
    Supported Speeds: 1Gb 2Gb
    Current Speed: not established
    Node WWN: 20000000c95f35dc
```

---

## 9.3 Attaching Solaris to DS8000

Solaris uses the LUN polling method to discover DS8000 LUNs. For this reason, each Solaris host is limited to 256 LUNs per HBA and volume group from the DS8000. IBM offers a mechanism to overcome this limitation. This workaround provides the ability to map multiple LUN groups of 256.

Using the `he1p mkhostconnect` DSCLI command, presents the output as shown in Example 9-3.

*Example 9-3 Extract of the help mkhostconnect DSCLI command output*

---

```
“-wwname wwpn
(Required) Specifies the worldwide port name (WWPN). The WWPN is a 16-character
hexadecimal ID. The names are host attachment specific; for example,
12341234000A000F.
```

Note: You should not create more than one hostconnect per WWPN, except for SUN hosts. Creating more than one hostconnect per WWPN (each with a separate volume group) is only supported for SUN.”

---

Setting up this approach requires an approved SCORE; it is a special IBM release process to get a formal approval from development labs for not generally supported environments. For more information about SCORE, see 2.3, “Additional supported configurations” on page 13.

You can assign LUNs using any of the supported DS8000 user interfaces, including the DS CLI, the DS GUI, and the DS Open API. When using the DS CLI, make the host connections using the flags **-addrdiscovery lunpolling**, **-lbs 512**, and **-profile "SUN - Solaris"**. Another option is to use the **-hosttype Sun** parameter. When making the volume groups, use the parameter **-type scsimap256**.

For native HBA drivers using the sd stack with the `/kernel/drv/sd.conf` file, it is best to use persistent binding. If you do not use persistent binding, Solaris can assign a different SCSI device identifier (SCSI ID) other than the SCSI ID it used previously each time it rescans the bus; for example, in search for new devices. Without persistent binding and changing SCSI IDs, you would need to reconfigure applications or even the operating system which is in most cases an undesired behavior.

The methods of enabling persistent binding differ, depending on your HBA. The *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917, contains HBA settings for each supported type.

After creating the **hostconnection** objects in the DS8000 and connecting them to a volume group containing LUNs, the storage system is providing them across the SAN allowing the Solaris server to scan the SCSI bus for new devices.

## 9.4 Multipathing in Solaris

As with other operating systems, you typically use multiple paths to access a LUN from your Solaris server on the DS8000 system. Multiple paths can help maximize the reliability and performance of your operating environment. The DS8000 supports three multipathing technologies on Solaris:

- ▶ IBM provides the System Storage Multipath SDD as part of the DS8000 at no extra charge.
- ▶ Oracle Solaris has a native multipathing software called the StorEdge Traffic Manager Software (STMS). STMS is commonly known as multiplexed I/O (MPxIO) in the industry, and the remainder of this book references this technology as MPxIO.
- ▶ IBM supports VERITAS Volume Manager (VxVM) Dynamic Multipathing (DMP), a part of the VERITAS Storage Foundation suite.

The multipathing technology that you use depends predominantly on your operating environment and your business requirements. There are a few limitations depending on your operating system version, your HBAs, and whether you use clustering. Details are available in the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917.

One difference between the multipathing technologies is in whether they suppress the redundant paths to the storage. MPxIO and DMP both suppress all paths to the storage except for one, and the device appears to the application as a single-path device. However, SDD allows the original paths to be seen, but creates its own virtual device, called a *vpath*, for applications to use.

If you assign LUNs to a server before you install multipathing software, you can see each LUN show up as two or more devices, depending on how many paths you have. In Example 9-4, the **iostat -nE** command shows that the volume 75207814206 appears twice, as `c2t1d1` on the first HBA, and as `c3t1d1` on the second HBA.

*Example 9-4 Device listing without multipath software*

---

```
# iostat -nE
c2t1d1      Soft Errors: 0 Hard Errors: 0 Transport Errors: 0
Vendor: IBM      Product: 2107900      Revision: .212 Serial No: 75207814206
Size: 10.74GB <10737418240 Bytes>
Media Error: 0 Device Not Ready: 0 No Device: 0 Recoverable: 0
Illegal Request: 0 Predictive Failure Analysis: 0
c2t1d0      Soft Errors: 0 Hard Errors: 0 Transport Errors: 0
Vendor: IBM      Product: 2107900      Revision: .212 Serial No: 75207814205
Size: 10.74GB <10737418240 Bytes>
Media Error: 0 Device Not Ready: 0 No Device: 0 Recoverable: 0
Illegal Request: 0 Predictive Failure Analysis: 0
c3t1d1      Soft Errors: 0 Hard Errors: 0 Transport Errors: 0
Vendor: IBM      Product: 2107900      Revision: .212 Serial No: 75207814206
Size: 10.74GB <10737418240 Bytes>
Media Error: 0 Device Not Ready: 0 No Device: 0 Recoverable: 0
Illegal Request: 0 Predictive Failure Analysis: 0
c3t1d0      Soft Errors: 0 Hard Errors: 0 Transport Errors: 0
Vendor: IBM      Product: 2107900      Revision: .212 Serial No: 75207814205
Size: 10.74GB <10737418240 Bytes>
Media Error: 0 Device Not Ready: 0 No Device: 0 Recoverable: 0
Illegal Request: 0 Predictive Failure Analysis: 0
```

---

## 9.4.1 Working with IBM System Storage Multipath SDD

SDD is available from your local IBM support team or it can be downloaded from the Internet. Both the SDD software and supporting documentation are available on this website:

[http://www.ibm.com/support/entry/portal/Downloads/Hardware/System\\_Storage/Storage\\_software/Other\\_software\\_products/System\\_Storage\\_Multipath\\_Subsystem\\_Device\\_Driver](http://www.ibm.com/support/entry/portal/Downloads/Hardware/System_Storage/Storage_software/Other_software_products/System_Storage_Multipath_Subsystem_Device_Driver)

After you install the SDD software, you can see that the paths were grouped into virtual vpath devices. Example 9-5 shows the output of the **showvpath** command.

*Example 9-5 Output of the showvpath command*

---

```
# /opt/IBMsdd/bin/showvpath
vpath1:      Serial Number : 75207814206
  c2t1d1s0    /devices/pci@6,4000/fibre-channel@2/sd@1,1:a,raw
  c3t1d1s0    /devices/pci@6,2000/fibre-channel@1/sd@1,1:a,raw

vpath2:      Serial Number : 75207814205
  c2t1d0s0    /devices/pci@6,4000/fibre-channel@2/sd@1,0:a,raw
  c3t1d0s0    /devices/pci@6,2000/fibre-channel@1/sd@1,0:a,raw
```

---

For each device, the operating system creates a node in the `/dev/dsk` and `/dev/rdisk` directories. After SDD is installed, you can see these new vpaths by listing the contents of those directories. With SDD, the old paths are not suppressed. Instead, new vpath devices show up as `/dev/rdisk/vpath1a`, for example. When creating volumes and file systems, be sure to use the vpath device instead of the original device.

SDD also offers parameters that you can tune for your environment. Specifically, SDD offers three separate load balancing schemes:

- ▶ Failover:
  - No load balancing.
  - The second path is used only if the preferred path fails.
- ▶ Round-robin:
  - The paths used are chosen at random, but separate paths than the most recent I/O.
  - If there are only two paths, then they alternate.
- ▶ Load balancing:
  - The path chosen based on estimated path load.
  - Default policy.

The policy can be set through the `datapath set device policy` command.

## 9.4.2 Using MPxIO

On Solaris 8 and Solaris 9 systems, MPxIO is available as a bundling of OS patches and packages. You must install these patches and packages to use MPxIO. On Solaris 10 systems, MPxIO is installed by default. In all cases, MPxIO needs to be enabled and configured before it can be used with the DS8000.

**Important:** Although widely assumed, it is not a must to use Sun badged HBAs to run MPxIO. The native HBAs from vendors such as Emulex and QLogic can be used as well. For more details, see the following websites:

- ▶ For Emulex:  
<http://www.oracle.com/technetwork/server-storage/solaris/overview/emulex-corporation-136533.html>
- ▶ For Qlogic:  
<http://www.oracle.com/technetwork/server-storage/solaris/overview/qlogic-corp--139073.html>

Before you enable MPxIO, configure the host bus adapters. Issue the `cfgadm -la` command for the current state of your adapters. Example 9-6 shows two adapters, c3 and c4, of type fc.

*Example 9-6* `cfgadm -la` command output

---

```
# cfgadm -la
```

Ap_Id	Type	Receptacle	Occupant	Condition
c3	fc	connected	unconfigured	unknown
c4	fc	connected	unconfigured	unknown

---

The command output shows that both adapters are unconfigured. To configure the adapters, issue `cfgadm -c configure cx`, where *x* is the adapter number, in this case, 3 and 4. Now, both adapters show up as configured.

**Solaris 10:** In Solaris 10, the `cfgadm` command is only needed if the LUNs are not visible when issuing the `format` command, after having mapped the LUNs to the Solaris host.

## Solaris 8 and 9

To configure MPxIO in Solaris 8 and 9, enable it by editing the `/kernel/drv/scsi_vhci.conf` file. Find the `mpxio-disable` parameter and change the setting to `no` (Example 9-7).

*Example 9-7 Changes to `/kernel/drv/scsi_vhci.conf`*

---

```
...
mpxio-disable="no";
```

---

## Solaris 10

To enable MPxIO in Solaris 10, complete these steps:

1. Execute the `stmsboot -e` command.
2. Add the following stanza to supply the vendor identification (VID) and product identification (PID) information to MPxIO in the `/kernel/drv/scsi_vhci.conf` file:

```
device-type-scsi-options-list =
"IBM      2107900", "symmetric-option";
symmetric-option = 0x1000000;
```

**Attention:** The vendor string must be exactly 8 bytes, so you must type IBM followed by five spaces.

3. Restart the system with the `reboot` command. After the restart, MPxIO is ready to be used. For example, the LUNs can now be prepared with the `format` command, and the partitions and filesystems can be created.

For more information about MPxIO, see the following websites:

- ▶ See the Oracle Solaris I/O Multipathing page at this website:  
<http://download.oracle.com/docs/cd/E19680-01/821-1253/intro-46/index.html>
- ▶ See the Oracle Wiki home page for MPxIO at this website:  
<http://wikis.sun.com/dosearchsite.action?queryString=MPxIO>
- ▶ See the Oracle Documentation page at this website:  
<http://www.oracle.com/technetwork/indexes/documentation/index.html>

### 9.4.3 Working with VERITAS Volume Manager dynamic multipathing

Before using VERITAS Volume Manager (VxVM) DMP, a part of the VERITAS Storage Foundation suite make sure you have downloaded and installed the latest maintenance pack.

You also need to download and install the Array Support Library (ASL) for the DS8000. To download ASL, select the respective product from the following website:

<https://sort.symantec.com/asl>

Upon selecting an item, such as Patches, you are redirected to the Veritas Operations site.

#### Using VxVM DMP

During device discovery, the `vxconfigd` daemon compares the serial numbers of the separate devices. If two devices have the same serial number, then they are the same LUN, and DMP combines the paths. Listing the contents of the `/dev/vx/rdmp` directory provides only one set of devices.

The `vxdisk path` command also demonstrates DMP's path suppression capabilities. In Example 9-8, you can see that device `c7t2d0s2` is suppressed and is only shown as a subpath of `c6t1d0s2`.

*Example 9-8 vxdisk path command output*

```
# vxdisk path
```

SUBPATH	DANAME	DMNAME	GROUP	STATE
c6t1d0s2	c6t1d0s2	Ethan01	Ethan	ENABLED
c7t2d0s2	c6t1d0s2	Ethan01	Ethan	ENABLED
c6t1d1s2	c7t2d1s2	Ethan02	Ethan	ENABLED
c7t2d1s2	c7t2d1s2	Ethan02	Ethan	ENABLED
c6t1d2s2	c7t2d2s2	Ethan03	Ethan	ENABLED
c7t2d2s2	c7t2d2s2	Ethan03	Ethan	ENABLED
c6t1d3s2	c7t2d3s2	Ethan04	Ethan	ENABLED
c7t2d3s2	c7t2d3s2	Ethan04	Ethan	ENABLED

Now, you create volumes using the device name listed under the DANAME column. In Figure 9-1, a volume is created using four disks, even though there are actually eight paths.

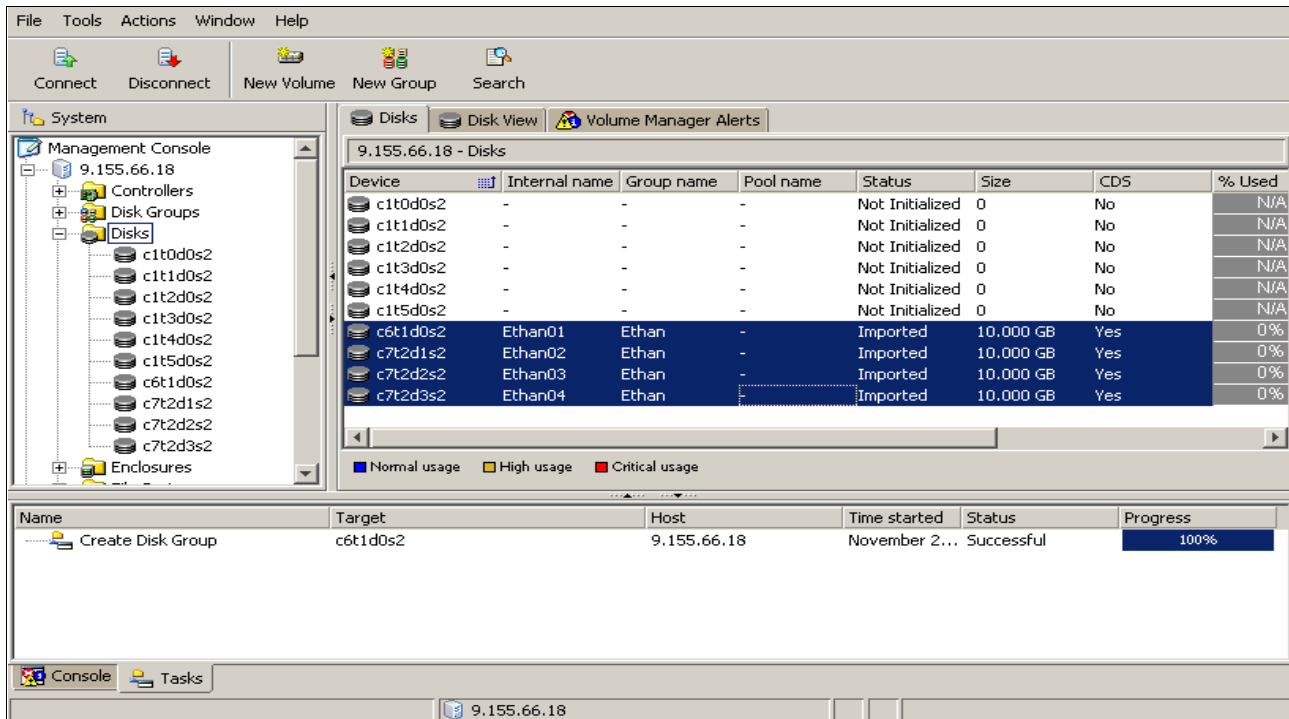


Figure 9-1 VERITAS DMP disk view

## Tuning parameters

As with other multipathing software, DMP provides a number of parameters that you can tune to maximize the performance and availability in your environment. For example, it is possible to set a load balancing policy to dictate how the I/O be shared between the separate paths. It is also possible to select which paths get used, and in which order, in case of a failure. You can find further details about the capabilities of DMP at the following website:

<http://www.symantec.com/business/support/index?page=content&id=H0WT074584>

## 9.5 Expanding dynamic volume with VxVM and DMP

Starting with licensed machine code 5.3.xx.xx, it is possible to expand a volume in the DS8000, even if it is mapped to a host. However, a volume that is in a FlashCopy, Metro Mirror, or Global Mirror relationship, cannot be expanded unless the relationship is removed, which means the FlashCopy, Metro Mirror, or Global Mirror on that volume must be removed before you can expand the volume. Information in this section explains how to expand a volume on a Solaris host, with VxVM on a DS8000 volume.

To display the volume size, use the command `lsfbvol`, as shown in Example 9-9.

*Example 9-9 lsfbvol -fullid before volume expansion*

```
dscli> lsfbvol -fullid 4704
Name          ID          accstate  datastate  configstate  deviceMTM  datatype  extpool          cap(2^30B)  cap(10^9B)  cap(blocks)
-----
ITS0_v880_4704 IBM.2107-7520781/4704 OnLine    Normal    Normal      2107-900  FB 512    IBM.2107-7520781/P53  12.0        -          25165824
```

In Example 9-9, the capacity is 12 GB and the volume ID is 4704. To determine the corresponding disk on the Solaris host, install the DS CLI on this host and execute the `lshostvol` command.

Example 9-10 shows the output.

*Example 9-10 lshostvol output*

```
bash-3.00# /opt/ibm/dscli/bin/lshostvol.sh
Device Name          Volume ID
-----
c2t50050763030CC1A5d0    IBM.2107-7520781/4704
c2t50050763030CC08Fd0    IBM.2107-7503461/4704
c2t5005076303000663d0    IBM.2107-75ABTV1/4704
c3t50050763030B0663d0    IBM.2107-75ABTV1/4704
c3t500507630319C08Fd0    IBM.2107-7503461/4704
c3t50050763031CC1A5d0    IBM.2107-7520781/4704
```

In Example 9-10, the volume with ID 75207814704 is c2t50050763030CC1A5d0 or c3t50050763031CC1A5d0 on the Solaris host.

To see the size of the volume on the Solaris host, use the `luxadm` command, as shown in Example 9-11.

*Example 9-11 luxadm output before volume expansion*

```
bash-3.00# luxadm display /dev/rdisk/c2t50050763030CC1A5d0s2
DEVICE PROPERTIES for disk: /dev/rdisk/c2t50050763030CC1A5d0s2
Status(Port A):      O.K.
Vendor:              IBM
Product ID:          2107900
WWN(Node):           5005076303ffc1a5
WWN(Port A):         50050763030cc1a5
Revision:            .991
Serial Num:          75207814704
Unformatted capacity: 12288.000 MBytes
Write Cache:         Enabled
Read Cache:          Enabled
  Minimum prefetch: 0x0
  Maximum prefetch: 0x16
Device Type:         Disk device
Path(s):
```

```

/dev/rdisk/c2t50050763030CC1A5d0s2
/devices/pci@9,600000/SUNW,q1lc@1/fp@0,0/ssd@w50050763030cc1a5,0:c,raw
/dev/rdisk/c3t50050763031CC1A5d0s2
/devices/pci@9,600000/SUNW,q1lc@2/fp@0,0/ssd@w50050763031cc1a5,0:c,raw

```

---

This indicates that the volume size is 12288 MB, equal to 12 GB. To obtain the dmpnodename of this disk in VxVM, use the **vxddmpadm** command used for Example 9-12.

*Example 9-12 VxVM commands before volume expansion*

---

```

bash-3.00# vxddmpadm getsubpaths c1r=c2
NAME          STATE[A]  PATH-TYPE[M]  DMPNODENAME  ENCLR-TYPE  ENCLR-NAME  ATTRS
=====
NONAME        DISABLED  -             IBM_DS8x002_1  IBM_DS8x00  IBM_DS8x002  -
c2t50050763030CC08Fd0s2  ENABLED(A) -             IBM_DS8x002_0  IBM_DS8x00  IBM_DS8x002  -
c2t50050763030CC1A5d0s2  ENABLED(A) -             IBM_DS8x001_0  IBM_DS8x00  IBM_DS8x001  -
c2t5005076303000663d0s2  ENABLED(A) -             IBM_DS8x000_0  IBM_DS8x00  IBM_DS8x000  -

bash-3.00# vxdisk list IBM_DS8x001_0
Device:      IBM_DS8x001_0
devicetag:   IBM_DS8x001_0
type:        auto
hostid:      v880
disk:        name=IBM_DS8x001_0 id=1194446100.17.v880
group:       name=20781_dg id=1194447491.20.v880
info:        format=cdsdisk,privoffset=256,pubslice=2,privslice=2
flags:       online ready private autoconfig autoimport imported
pubpaths:    block=/dev/vx/dmp/IBM_DS8x001_0s2 char=/dev/vx/rdmp/IBM_DS8x001_0s2
guid:        {9ecb6cb6-1dd1-11b2-af7a-0003ba43fdc1}
udid:        IBM%5F2107%5F7520781%5F6005076303FFC1A50000000000004704
site:        -
version:     3.1
iosize:      min=512 (Bytes) max=2048 (blocks)
public:    slice=2 offset=65792 len=25095808 disk_offset=0
private:     slice=2 offset=256 len=65536 disk_offset=0
update:      time=1194447493 seqno=0.15
ssb:         actual_seqno=0.0
headers:     0 240
configs:     count=1 len=48144
logs:        count=1 len=7296
Defined regions:
  config  priv 000048-000239[000192]: copy=01 offset=000000 enabled
  config  priv 000256-048207[047952]: copy=01 offset=000192 enabled
  log     priv 048208-055503[007296]: copy=01 offset=000000 enabled
  lockrgn priv 055504-055647[000144]: part=00 offset=000000
Multipathing information:
numpaths:    2
c2t50050763030CC1A5d0s2 state=enabled
c3t50050763031CC1A5d0s2 state=enabled

```

---

In Example 9-12, the capacity of the disk, as shown in VxVM, can be found on the output line labeled **public:**, after issuing a **vxdisk list <dmpnodename>** command. Multiply the value for **len** by 512 bytes, which is equal to 12 GB (25095808 x 512).



After the file system on the logical volume of the VxVM diskgroup is created, the size of the file system, 11 GB, is mounted on /20781 and displayed using the **df** command (Example 9-13).

*Example 9-13 The df command before volume expansion*

```

bash-3.00# df -k
Filesystem          kBytes    used   avail capacity  Mounted on
/dev/dsk/c1t0d0s0  14112721 4456706 9514888    32%      /
/devices            0         0       0         0%      /devices
ctfs                0         0       0         0%      /system/contract
proc               0         0       0         0%      /proc
mnttab             0         0       0         0%      /etc/mnttab
swap              7225480   1160   7224320    1%      /etc/svc/volatile
objfs              0         0       0         0%      /system/object
fd                 0         0       0         0%      /dev/fd
swap              7224320    0   7224320    0%      /tmp
swap              7224368    48   7224320    1%      /var/run
swap              7224320    0   7224320    0%      /dev/vx/dmp
swap              7224320    0   7224320    0%      /dev/vx/rdmp
/dev/dsk/c1t0d0s7  20160418 2061226 17897588   11%      /export/home
/dev/vx/dsk/03461_dg/03461_vol
                  10485760  20062 9811599    1%      /03461
/dev/vx/dsk/20781_dg/20781_vol
                  11534336  20319 10794398    1%      /20781

```

To expand the volume on the DS8000, use the DS CLI **chfbvol** command, as shown in Example 9-14.

*Example 9-14 Expanding a volume*

```

dscli> chfbvol -cap 18 4704
IBM.2107-7520781
CMUC00332W chfbvol: Some host operating systems do not support changing the volume
size. Are you sure that you want to resize the volume? [y/n]: y
CMUC00026I chfbvol: FB volume 4704 successfully modified.

```

The new capacity must be larger than the previous one; you *cannot* shrink the volume.

To check that the volume was expanded, use the **lsfbvol** command, as shown in Example 9-15.

*Example 9-15 lsfbvol after expansion*

```

dscli> lsfbvol 4704
Name          ID  accstate  datastate  configstate  deviceMTM  datatype  extpool  cap (2^30B)  cap (10^9B)  cap (blocks)
-----
ITS0_v880_4704 4704 Online    Normal    Normal      2107-900   FB 512    P53          18.0         -           37748736

```

In Example 9-15, the volume 4704 is expanded to 18 GB in capacity.

To see the changed size of the volume on the Solaris host after the expansion, use the **luxadm** command (Example 9-16).

*Example 9-16 luxadm output after volume expansion*

```

bash-3.00# luxadm display /dev/rdisk/c2t50050763030CC1A5d0s2
DEVICE PROPERTIES for disk: /dev/rdisk/c2t50050763030CC1A5d0s2
Status(Port A):      O.K.

```

```

Vendor:                IBM
Product ID:            2107900
WWN(Node):             5005076303ffca5
WWN(Port A):           50050763030cca5
Revision:              .991
Serial Num:            75207814704
Unformatted capacity: 18432.000 MBytes
Write Cache:           Enabled
Read Cache:            Enabled
  Minimum prefetch:    0x0
  Maximum prefetch:    0x16
Device Type:           Disk device
Path(s):
/dev/rdisk/c2t50050763030CC1A5d0s2
/devices/pci@9,600000/SUNW,q1c@1/fp@0,0/ssd@w50050763030cca5,0:c,raw
/dev/rdisk/c3t50050763031CC1A5d0s2
/devices/pci@9,600000/SUNW,q1c@2/fp@0,0/ssd@w50050763031cca5,0:c,raw

```

---

The disk now has a capacity of 18 GB. To use the additional capacity, issue the **vxdisk resize** command, as shown in Example 9-17.

*Example 9-17 VxVM commands after volume expansion*

---

```

bash-3.00# vxdisk resize IBM_DS8x001_0

bash-3.00# vxdisk list IBM_DS8x001_0
Device:      IBM_DS8x001_0
devicetag:   IBM_DS8x001_0
type:        auto
hostid:      v880
disk:        name=IBM_DS8x001_0 id=1194446100.17.v880
group:       name=20781_dg id=1194447491.20.v880
info:        format=cdsdisk,privoffset=256,pubslice=2,privslice=2
flags:       online ready private autoconfig autoimport imported
pubpaths:    block=/dev/vx/dmp/IBM_DS8x001_0s2 char=/dev/vx/rdmp/IBM_DS8x001_0s2
guid:        {fbdbfe12-1dd1-11b2-af7c-0003ba43fdc1}
udid:        IBM%5F2107%5F7520781%5F6005076303FFC1A500000000000004704
site:        -
version:     3.1
iosize:      min=512 (Bytes) max=2048 (blocks)
public:      slice=2 offset=65792 len=37677568 disk_offset=0
private:     slice=2 offset=256 len=65536 disk_offset=0
update:      time=1194473744 seqno=0.16
ssb:         actual_seqno=0.0
headers:     0 240
configs:     count=1 len=48144
logs:        count=1 len=7296
Defined regions:
  config    priv 000048-000239[000192]: copy=01 offset=000000 enabled
  config    priv 000256-048207[047952]: copy=01 offset=000192 enabled
  log       priv 048208-055503[007296]: copy=01 offset=000000 enabled
  lockrgn   priv 055504-055647[000144]: part=00 offset=000000
Multipathing information:
numpaths:    2
c2t50050763030CC1A5d0s2 state=enabled
c3t50050763031CC1A5d0s2 state=enabled

```

---

After the volume expansion, the disk size is 37677568. When multiplied by 512 bytes, it equals 18 GB.

**Tip:** You need at least two disks in the diskgroup where you want to resize a disk, otherwise, the `vxdisk resize` command will fail.

To expand the logical volume and the file system in VxVM, first you need to determine the maximum size that you can expand to, then you can expand the logical volume and the file system. To find the maximum size that you can expand to, issue the `vxvoladm` command, as shown in Example 9-18.

*Example 9-18 VxVM logical volume expansion*

```
bash-3.00# vxvoladm -g 20781_dg maxgrow 20781_vol
Volume can be extended to: 37677056(17.97g)

bash-3.00# vxvoladm -g 20781_dg growto 20781_vol 37677056

bash-3.00# /opt/VRTS/bin/fsadm -b 17g /20781
UX:vxfs fsadm: INFO: V-3-25942: /dev/vx/rdsk/20781_dg/20781_vol size increased
from 23068672 sectors to 35651584 sectors
```

After the file system expansion, the `df` command shows a size of 17825792 KB, equal to approximately 17 GB, on file system `/dev/vx/dsk/20781_dg/20781_vol`, as shown in Example 9-19.

*Example 9-19 df command after file system expansion*

```
bash-3.00# df -k
```

Filesystem	kBytes	used	avail	capacity	Mounted on
/dev/dsk/c1t0d0s0	14112721	4456749	9514845	32%	/
/devices	0	0	0	0%	/devices
ctfs	0	0	0	0%	/system/contract
proc	0	0	0	0%	/proc
mnttab	0	0	0	0%	/etc/mnttab
swap	7222640	1160	7221480	1%	/etc/svc/volatile
objfs	0	0	0	0%	/system/object
fd	0	0	0	0%	/dev/fd
swap	7221480	0	7221480	0%	/tmp
swap	7221528	48	7221480	1%	/var/run
swap	7221480	0	7221480	0%	/dev/vx/dmp
swap	7221480	0	7221480	0%	/dev/vx/rdmp
/dev/dsk/c1t0d0s7	20160418	2061226	17897588	11%	/export/home
/dev/vx/dsk/03461_dg/03461_vol	10485760	20062	9811599	1%	/03461
/dev/vx/dsk/20781_dg/20781_vol	17825792	21861	16691193	1%	/20781

## 9.6 Booting from SAN

Solaris can be installed and booted off a SAN volume as opposed to a server internal disk. This section provides information about how to boot Solaris from a DS8000 LUN.

To install an operating system onto a LUN that resides in the SAN, the server needs to have access to it. It means that the Fibre Channel adapters in the server need to have a facility to log in to a SAN fabric and scan and provision a LUN as a disk available to the operating system installation routine. This facility is provided by special boot code that needs to be loaded into the Fibre Channel adapters. For this load, you need a running operating system environment, for example, Solaris installed on an internal disk device. Example 9-20 shows the boot device and the mount points of such an environment:

### Example 9-20 Current setup

```
root@prime-lab-04/export/home/test# eeprom | grep boot-device
boot-device=/pci@83,4000/FJSV,ulsa@2,1/disk@0,0:a disk
root@prime-lab-04/# format
Searching for disks...done
```

#### AVAILABLE DISK SELECTIONS:

0. **c0t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>**  
**/pci@83,4000/FJSV,ulsa@2,1/sd@0,0**
1. c1t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>  
/pci@83,4000/FJSV,ulsa@2/sd@0,0
2. c3t500507630A03029Fd0 <IBM-2107900-.268 cyl 4367 alt 2 hd 30 sec 64>  
/pci@80,2000/fibre-channel@2/fp@0,0/ssd@w500507630a03029f,0
3. c3t500507630A03029Fd1 <IBM-2107900-.268 cyl 8736 alt 2 hd 30 sec 64>  
/pci@80,2000/fibre-channel@2/fp@0,0/ssd@w500507630a03029f,1
4. c3t500507630A03029Fd2 <IBM-2107900-.268 cyl 17918 alt 2 hd 64 sec 256>  
/pci@80,2000/fibre-channel@2/fp@0,0/ssd@w500507630a03029f,2
5. c4t500507630A00029Fd0 <IBM-2107900-.268 cyl 4367 alt 2 hd 30 sec 64>  
/pci@80,2000/fibre-channel@1/fp@0,0/ssd@w500507630a00029f,0
6. c4t500507630A00029Fd1 <IBM-2107900-.268 cyl 8736 alt 2 hd 30 sec 64>  
/pci@80,2000/fibre-channel@1/fp@0,0/ssd@w500507630a00029f,1
7. c4t500507630A00029Fd2 <IBM-2107900-.268 cyl 17918 alt 2 hd 64 sec 256>  
/pci@80,2000/fibre-channel@1/fp@0,0/ssd@w500507630a00029f,2

```
root@prime-lab-04/# df
/ (/dev/dsk/c0t0d0s0 ):21248478 blocks 1706174 files
/devices (/devices ): 0 blocks 0 files
/system/contract (ctfs ): 0 blocks 2147483598 files
/proc (proc ): 0 blocks 29889 files
/etc/mnttab (mnttab ): 0 blocks 0 files
/etc/svc/volatile (swap ):22660000 blocks 567860 files
/system/object (objfs ): 0 blocks 2147483414 files
/dev/fd (fd ): 0 blocks 0 files
/var (/dev/dsk/c0t0d0s5 ): 2064656 blocks 591399 files
/tmp (swap ):22660000 blocks 567860 files
/var/run (swap ):22660000 blocks 567860 files
/export/home (/dev/dsk/c0t0d0s7 ):53769918 blocks 4725414 files
```

For these examples, a Fujitsu Siemens Computers Primepower 450 server, equipped with two LP10000 Emulex HBAs was used, as you can see in Example 9-21. Taking it into account, you can find these processes in the Emulex Bootcode User Manual, “Boot Code User Manual For Emulex Adapters” at this website:

<http://www.emulex.com/files/downloads/hardware/bootcode.pdf>

Follow the steps described in the topic *Configure Boot from SAN on Solaris SFS (SPARC)* of the Emulex document. All the steps need to be executed from the Open Boot Prompt (OBP) environment, thus without running Solaris. This document instructs that boot code must be installed and enabled to boot from SAN. Also note that the boot code for Solaris emlxs (SFS) systems is enabled automatically when it is installed, so no utility is needed

## 9.6.1 Displaying the boot code

In Example 9-21 you can see that the boot code installed is FCode/BIOS Version: 3.10a3. If the boot code is not installed on your HBAs, download it from the Emulex website. Issue the `fcinfo` command to see the boot code installed and general HBA information.

*Example 9-21 Displaying the boot code*

---

```
bash-3.00# fcinfo hba-port
HBA Port WWN: 10000000c95f3592
  OS Device Name: /dev/cfg/c4
  Manufacturer: Emulex
  Model: LP10000
  Firmware Version: 1.92a1 (T2D1.92A1)
  FCode/BIOS Version: 3.10a3
  Serial Number: MS65035391
  Driver Name: emlxs
  Driver Version: 2.31h (2008.06.16.08.54)
  Type: N-port
  State: online
  Supported Speeds: 1Gb 2Gb
  Current Speed: 2Gb
  Node WWN: 20000000c95f3592
HBA Port WWN: 10000000c95f35dc
  OS Device Name: /dev/cfg/c3
  Manufacturer: Emulex
  Model: LP10000
  Firmware Version: 1.92a1 (T2D1.92A1)
  FCode/BIOS Version: 3.10a3
  Serial Number: MS65035465
  Driver Name: emlxs
  Driver Version: 2.31h (2008.06.16.08.54)
  Type: N-port
  State: online
  Supported Speeds: 1Gb 2Gb
  Current Speed: 2Gb
  Node WWN: 20000000c95f35dc
```

---

When the Fibre Channel adapters are prepared with the boot code as described, they can be used to install a new instance of Solaris on a LUN that resides in the SAN.

## 9.6.2 Booting off a DS8000 LUN with Solaris

Example 9-22 shows a Solaris system that was installed from a CDROM on a DS8000 LUN with file systems residing on it. Solaris can be booted off that DS8000 LUN afterwards.

*Example 9-22 Solaris that was booted off a DS8000 LUN*

---

```
bash-3.00# df
/ (/dev/dsk/c5t600507630AFFC29F000000000003101d0s0): 3667996
blocks 564069 files
/devices (/devices ): 0 blocks 0 files
/system/contract (ctfs ): 0 blocks 2147483621 files
/proc (proc ): 0 blocks 29933 files
/etc/mnttab (mnttab ): 0 blocks 0 files
/etc/svc/volatile (swap ): 6904240 blocks 558165 files
/system/object (objfs ): 0 blocks 2147483497 files
/etc/dfs/sharetab (sharefs ): 0 blocks 2147483646 files
/dev/fd (fd ): 0 blocks 0 files
/tmp (swap ): 6904240 blocks 558165 files
/var/run (swap ): 6904240 blocks 558165 files
/export/home (/dev/dsk/c5t600507630AFFC29F000000000003101d0s7): 3263196
blocks 407676 files
```

```
bash-3.00# format
Searching for disks...done
```

AVAILABLE DISK SELECTIONS:

```
0. c0t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>
   /pci@83,4000/FJSV,ulsa@2,1/sd@0,0
1. c1t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>
   /pci@83,4000/FJSV,ulsa@2/sd@0,0
2. c5t600507630AFFC29F0000000000003100d0 <IBM-2107900-.268 cyl 4367 alt 2
   hd 30 sec 64>
   /scsi_vhci/ssd@g600507630affc29f000000000003100
3. c5t600507630AFFC29F0000000000003101d0 <IBM-2107900-.268 cyl 8736 alt 2
   hd 30 sec 64>
   /scsi_vhci/ssd@g600507630affc29f000000000003101
4. c5t600507630AFFC29F0000000000003102d0 <IBM-2107900-.268 cyl 17918 alt 2
   hd 64 sec 256>
   /scsi_vhci/ssd@g600507630affc29f000000000003102
```

---

### 9.6.3 Supplying the VID or PID string

In order to take symmetric arrays under control of MPxIO, as shown in Example 9-22 on page 174, its VID or PID string must be supplied in the `/kernel/drv/scsi_vhci.conf` file. For more information, see 9.4.2, "Using MPxIO" on page 164.

However, because the operating system used for these examples was installed from scratch, there was no possibility to supply the VID or PID string shown in Example 9-23. The DS8000 LUNs are already under control of MPxIO. For example, DS8000 LUN has one entry instead of two, as shown in Example 9-22 on page 174. Example 9-23 here shows the content of the `scsi_vhci.conf` file.

*Example 9-23 Content of the `scsi_vhci.conf` file*

---

```
bash-3.00# more /kernel/drv/scsi_vhci.conf
#
# Copyright 2004 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
#pragma ident    "@(#)scsi_vhci.conf    1.9    04/08/26 SMI"
#
name="scsi_vhci" class="root";
#
# Load balancing global configuration: setting load-balance="none" will cause
# all I/O to a given device (which supports multipath I/O) to occur via one
# path. Setting load-balance="round-robin" will cause each path to the device
# to be used in turn.
#
load-balance="round-robin";
#
# Automatic failback configuration
# possible values are auto-failback="enable" or auto-failback="disable"
auto-failback="enable";
#
# For enabling MPxIO support for 3rd party symmetric device need an
# entry similar to following in this file. Just replace the "SUN    SENA"
# part with the Vendor ID/Product ID for the device, exactly as reported by
# Inquiry cmd.
#
# device-type-scsi-options-list =
# "SUN    SENA", "symmetric-option";
#
# symmetric-option = 0x1000000;
```

---

To supply the VID/PID string, complete the following steps:

1. Disable MPxIO, as shown in Example 9-24.

*Example 9-24 Disabling MPxIO*

---

```
bash-3.00# stmsboot -d
```

```
WARNING: stmsboot operates on each supported multipath-capable controller
detected in a host. In your system, these controllers are
```

```
/devices/pci@80,2000/emlx@2/fp@0,0
/devices/pci@80,2000/emlx@2/fp@1,0
/devices/pci@80,2000/emlx@1/fp@0,0
/devices/pci@80,2000/emlx@1/fp@1,0
```

If you do NOT wish to operate on these controllers, please quit stmsboot and re-invoke with `-D { fp | mpt }` to specify which controllers you wish to modify your multipathing configuration for.

```
Do you wish to continue? [y/n] (default: y)
Checking mpzio status for driver fp
Checking mpzio status for driver mpt
WARNING: This operation will require a reboot.
Do you want to continue ? [y/n] (default: y)
The changes will come into effect after rebooting the system.
Reboot the system now ? [y/n] (default: y)
updating /platform/sun4us/boot_archive
```

---

2. Restart your system. Example 9-25 shows that after restarting your system, the following actions were taken:
  - The file systems are now reported as residing on `/dev/dsk/c4t500507630A00029Fd1s0` and `/dev/dsk/c4t500507630A00029Fd1s7`.
  - The DS8000 LUN selected when installing Solaris from CDROM is `c4t500507630A00029Fd1`.
  - MPxIO was disabled and you can see twice as many DS8000 LUNs, each separate path is reported, as shown in Example 9-25.

*Example 9-25 Solaris that was booted off a DS8000 LUN with MPxIO disabled*

---

```
bash-3.00# df
/                (/dev/dsk/c4t500507630A00029Fd1s0): 3667182 blocks  564064
files
/devices         (/devices         ):      0 blocks      0 files
/system/contract (ctfs             ):      0 blocks 2147483622 files
/proc           (proc            ):      0 blocks   29934 files
/etc/mnttab     (mnttab          ):      0 blocks      0 files
/etc/svc/volatile (swap            ): 6897264 blocks  558173 files
/system/object  (objfs           ):      0 blocks 2147483465 files
/etc/dfs/sharetab (sharefs         ):      0 blocks 2147483646 files
/dev/fd         (fd              ):      0 blocks      0 files
/tmp            (swap            ): 6897264 blocks  558173 files
/var/run        (swap            ): 6897264 blocks  558173 files
/export/home    (/dev/dsk/c4t500507630A00029Fd1s7): 3263196 blocks  407676
files
bash-3.00# format
```



Searching for disks...done

AVAILABLE DISK SELECTIONS:

0. c0t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>  
/pci@83,4000/FJSV,ulsa@2,1/sd@0,0
  1. c1t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>  
/pci@83,4000/FJSV,ulsa@2/sd@0,0
  2. c3t500507630A03029Fd0 <IBM-2107900-.268 cyl 4367 alt 2 hd 30 sec 64>  
/pci@80,2000/emlx@2/fp@0,0/ssd@w500507630a03029f,0
  3. **c3t500507630A03029Fd1** <IBM-2107900-.268 cyl 8736 alt 2 hd 30 sec 64>  
/pci@80,2000/emlx@2/fp@0,0/ssd@w500507630a03029f,1
  4. c3t500507630A03029Fd2 <IBM-2107900-.268 cyl 17918 alt 2 hd 64 sec 256>  
/pci@80,2000/emlx@2/fp@0,0/ssd@w500507630a03029f,2
  5. c4t500507630A00029Fd0 <IBM-2107900-.268 cyl 4367 alt 2 hd 30 sec 64>  
/pci@80,2000/emlx@1/fp@0,0/ssd@w500507630a00029f,0
  6. **c4t500507630A00029Fd1** <IBM-2107900-.268 cyl 8736 alt 2 hd 30 sec 64>  
/pci@80,2000/emlx@1/fp@0,0/ssd@w500507630a00029f,1
  7. c4t500507630A00029Fd2 <IBM-2107900-.268 cyl 17918 alt 2 hd 64 sec 256>  
/pci@80,2000/emlx@1/fp@0,0/ssd@w500507630a00029f,2
- 

3. Customize the /kernel/drv/scsi\_vhci.conf file, as shown in Example 9-26.

*Example 9-26 Customizing the /kernel/drv/scsi\_vhci.conf file*

---

```
bash-3.00# more /kernel/drv/scsi_vhci.conf
#
# Copyright 2004 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
#pragma ident    "@(#)scsi_vhci.conf    1.9    04/08/26 SMI"
#
name="scsi_vhci" class="root";
#
# Load balancing global configuration: setting load-balance="none" will cause
# all I/O to a given device (which supports multipath I/O) to occur via one
# path. Setting load-balance="round-robin" will cause each path to the device
# to be used in turn.
#
load-balance="round-robin";
#
# Automatic failback configuration
# possible values are auto-failback="enable" or auto-failback="disable"
auto-failback="enable";
#
# For enabling MPxIO support for 3rd party symmetric device need an
# entry similar to following in this file. Just replace the "SUN    SENA"
# part with the Vendor ID/Product ID for the device, exactly as reported by
# Inquiry cmd.
#
# device-type-scsi-options-list =
device-type-scsi-options-list =
# "SUN    SENA", "symmetric-option";
"IBM    2109700", "symmetric-option";
#
```

```
# symmetric-option = 0x1000000;  
symmetric-option = 0x1000000;
```

---

4. Enter the **stmsboot -e** command to reactivate MPxIO, as shown in Example 9-27.

*Example 9-27 Reactivating MPxIO*

---

```
bash-3.00# stmsboot -e
```

```
WARNING: stmsboot operates on each supported multipath-capable controller  
detected in a host. In your system, these controllers are
```

```
/devices/pci@80,2000/emlx@2/fp@0,0  
/devices/pci@80,2000/emlx@2/fp@1,0  
/devices/pci@80,2000/emlx@1/fp@0,0  
/devices/pci@80,2000/emlx@1/fp@1,0
```

```
If you do NOT wish to operate on these controllers, please quit stmsboot  
and re-invoke with -D { fp | mpt } to specify which controllers you wish  
to modify your multipathing configuration for.
```

```
Do you wish to continue? [y/n] (default: y)  
Checking mpxio status for driver fp  
Checking mpxio status for driver mpt  
WARNING: This operation will require a reboot.  
Do you want to continue ? [y/n] (default: y)  
The changes will come into effect after rebooting the system.  
Reboot the system now ? [y/n] (default: y)  
updating /platform/sun4us/boot_archive
```

---

5. Restart your system. After restarting your system, you can see that Solaris was booted off a DS8000 LUN with MPxIO enabled, as shown in Example 9-28.

*Example 9-28 Solaris that was booted off a DS8000 LUN (MPxIO is enabled)*

---

```
bash-3.00# df  
/  
 (/dev/dsk/c5t600507630AFFC29F00000000003101d0s0): 3667056  
blocks 564060 files  
/devices (/devices ): 0 blocks 0 files  
/system/contract (ctfs ): 0 blocks 2147483621 files  
/proc (proc ): 0 blocks 29932 files  
/etc/mnttab (mnttab ): 0 blocks 0 files  
/etc/svc/volatile (swap ): 6897584 blocks 558165 files  
/system/object (objfs ): 0 blocks 2147483465 files  
/etc/dfs/sharetab (sharefs ): 0 blocks 2147483646 files  
/dev/fd (fd ): 0 blocks 0 files  
/tmp (swap ): 6897584 blocks 558165 files  
/var/run (swap ): 6897584 blocks 558165 files  
/export/home (/dev/dsk/c5t600507630AFFC29F00000000003101d0s7): 3263196  
blocks 407676 files  
bash-3.00# format  
Searching for disks...done
```

```

AVAILABLE DISK SELECTIONS:
  0. c0t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>
    /pci@83,4000/FJSV,ulsa@2,1/sd@0,0
  1. c1t0d0 <FUJITSU-MAW3073NC-3701 cyl 24345 alt 2 hd 8 sec 737>
    /pci@83,4000/FJSV,ulsa@2/sd@0,0
  2. c5t600507630AFFC29F0000000000003100d0 <IBM-2107900-.268 cyl 4367 alt
2 hd 30 sec 64>
    /scsi_vhci/ssd@g600507630affc29f0000000000003100
  3. c5t600507630AFFC29F0000000000003101d0 <IBM-2107900-.268 cyl 8736 alt
2 hd 30 sec 64>
    /scsi_vhci/ssd@g600507630affc29f0000000000003101
  4. c5t600507630AFFC29F0000000000003102d0 <IBM-2107900-.268 cyl 17918 alt
2 hd 64 sec 256>
    /scsi_vhci/ssd@g600507630affc29f0000000000003102
Specify disk (enter its number):

```

---

## 9.6.4 Associating the MPxIO device file and underlying paths

Example 9-29 shows the association between the MPxIO device special file representation and its underlying paths.

*Example 9-29 MPxIO device file representation and underlying paths*

```

bash-3.00# luxadm display /dev/rdisk/c5t600507630AFFC29F0000000000003101d0s2
DEVICE PROPERTIES for disk: /dev/rdisk/c5t600507630AFFC29F0000000000003101d0s2
Vendor:                IBM
Product ID:            2107900
Revision:              .268
Serial Num:            75TV1813101
Unformatted capacity: 8192.000 MBytes
Write Cache:           Enabled
Read Cache:            Enabled
  Minimum prefetch:    0x0
  Maximum prefetch:    0x16
Device Type:           Disk device
Path(s):

/dev/rdisk/c5t600507630AFFC29F0000000000003101d0s2
/devices/scsi_vhci/ssd@g600507630affc29f0000000000003101:c,raw
Controller              /devices/pci@80,2000/emlx@1/fp@0,0
  Device Address          500507630a00029f,1
  Host controller port WWN 10000000c95f3592
  Class                   primary
  State                   ONLINE
Controller              /devices/pci@80,2000/emlx@2/fp@0,0
  Device Address          500507630a03029f,1
  Host controller port WWN 10000000c95f35dc
  Class                   primary
  State                   ONLINE

```

---





## HP-UX considerations

This chapter provides information for the specifics of attaching IBM System Storage DS8000 systems to host systems running Hewlett-Packard UNIX (HP-UX). There are multiple versions of HP-UX available, however, this chapter presents information based on HP-UX 11iv3.

The following topics are covered:

- ▶ Working with HP-UX
- ▶ Available resources
- ▶ Identifying available HBAs
- ▶ Identifying WWPNS of HBAs
- ▶ Configuring the HP-UX host for the DS8000
- ▶ Multipathing
- ▶ Working with VERITAS Volume Manager on HP-UX
- ▶ Working with LUNs

## 10.1 Working with HP-UX

This chapter presents configuration information for attaching a host running HP-UX to the point where the host is capable of running I/O to the DS8000 device. HP-UX is the UNIX System V implementation offered by Hewlett-Packard (HP). The DS8000 supports the attachment to hosts running HP-UX 11i or later.

There is a release name and identifier assigned to each release of HP-UX that can be displayed with the `uname -r` command. The response of B.11.31 indicates that the host is running HP-UX 11i v3, as shown in Example 10-1.

*Example 10-1 Determining the version of HP-UX*

---

```
# uname -r
B.11.31
```

---

### 10.1.1 The agile view

HP-UX commands that deal with storage have a choice in how the information is presented. The industry usage of Device Special Files (DSF), discussed further in 10.5.1, “Device special files” on page 185, leads to two different ways of describing the mass storage: the *legacy view* and the *agile view*. The legacy view of storage devices is simply the legacy DSFs and legacy hardware paths utilizing the familiar description of **cXtYdZ** that was in common use for a long time. The agile view uses LUN hardware paths and persistent DSFs.

Different commands offer their information in the different views. Some commands, such as `iocscan`, will show the legacy view by default. But they can be made to show the agile view with the use of the `-N` option. Be aware of this option as you work through the remainder of this chapter.

### 10.1.2 Notes on multipathing

For providing a fault tolerant connection to a HP-UX system, there are three possible multipathing solutions. However, only one, native multipathing, is appropriate for the DS8000. A more complete discussion of multipathing can be found in “Multipathing” on page 187. The following options are available:

- ▶ HP-UX native multipathing, introduced with HP-UX 11iv3.
- ▶ HP PVLINKS, as provided in earlier releases of HP-UX. PVLINKS is still available in HP-UX 11iv3. However, when attaching to DS8000, you need to use *native multipathing*.
- ▶ IBM Multipath Subsystem Device Driver (SDD).

**HP-UX 11iv3:** SDD from IBM is not supported with HP-UX 11iv3.

Native multipathing from HP offers load balancing over the available I/O paths and I/O path failover, in case of a connection failure. The older PVLINKS provides a failover solution, but does not offer the feature of automatic load balancing.

### 10.1.3 Notes on naming conventions

HP-UX 11iv3 supports two different Volume Managers:

- LVM** The Logical Volume Manager (LVM) is the default Volume Manager for HP-UX 11i. More information about LVM can be found in the HP publication, *HP-UX System Administrator's Guide: Logical Volume Management*.
- VxVM** The Veritas Volume Manager (VxVM) is a full-featured Volume Manager and is included with HP-UX 11i. However, there is only a base license provided that does not activate all of the features of this software. More information about VxVM can be found in the *Veritas Volume Manager Release Notes*.

**Important:** Each physical disk can only be managed by one Volume Manager at a time, either LVM or VxVM, but not both.

Both of these Volume Manager software packages can co-exist on a HP-UX 11i server, each keeping track of the physical disks and logical volumes that it manages. Both of these software packages are responsible for managing a “*pool of space*” and provisioning that space as logical containers. According to the HP documentation, these pools of space are referred to as *volume groups* by LVM. VxVM refers to them as *disk groups*. These two separate terms refer to the same concept, the logical representation of one or more physical disks.

**Tip:** LVM volume groups = VxVM disk groups.

### 10.1.4 Notes on enumeration

Due to limitations within HP-UX, there is a maximum number that can be assigned when configuring LUNs. ID Numbers greater than x'3FFF' (Decimal: 16383) are not supported. It creates a limitation of 16,384 LUNs that can be enumerated on the host. See the IBM Information Center section on “HP-UX host attachment” for more information.

**Important:** LUN IDs cannot exceed x'3FFF'.

[http://publib.boulder.ibm.com/infocenter/dsichelp/ds8000ic/index.jsp?topic=/com.ibm.storage.ssic.help.doc/f2c\\_attchnghpux\\_1tlxvv.html](http://publib.boulder.ibm.com/infocenter/dsichelp/ds8000ic/index.jsp?topic=/com.ibm.storage.ssic.help.doc/f2c_attchnghpux_1tlxvv.html)

## 10.2 Available resources

For the latest available, supported HP-UX configuration and required software patches, see the IBM SSIC at the following website:

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

For information about how to prepare the host to attach the DS8000, see the DS8000 Information Center at the following website:

<http://publib.boulder.ibm.com/infocenter/dsichelp/ds8000ic/index.jsp>

In the Information Center, select **Configuring** → **Attaching Hosts** → **HP-UX host attachment**.

To get the latest version of the DS CLI to install on your host, use the version delivered with the DS8000 Microcode Bundle. Alternatively, download the latest available International Organization for Standardization (ISO) image for the DS CLI client download at this website:

[ftp://ftp.software.ibm.com/storage/ds8000/updates/DS8K\\_Customer\\_Download\\_Files/CLI](ftp://ftp.software.ibm.com/storage/ds8000/updates/DS8K_Customer_Download_Files/CLI)

## 10.3 Identifying available HBAs

To see which Host Bus Adapters (HBAs) are available, use the `ioscan -fnC fc` command as shown in Example 10-2. The output shows four HBAs in this server identified as `/dev/fcdN`, where  $0 \leq N \leq 3$ .

*Example 10-2 Identifying HBAs with ioscan -fnC fc*

---

```
# ioscan -fnC fc
Class      I  H/W Path  Driver S/W State  H/W Type  Description
-----
fc         0  0/3/1/0  fcd   CLAIMED   INTERFACE  HP AB379-60101 4Gb Dual Port
PCI/PCI-X Fibre Channel Adapter (FC Port 1)
                /dev/fcd0
fc         1  0/3/1/1  fcd   CLAIMED   INTERFACE  HP AB379-60101 4Gb Dual Port
PCI/PCI-X Fibre Channel Adapter (FC Port 2)
                /dev/fcd1
fc         2  0/7/1/0  fcd   CLAIMED   INTERFACE  HP AB379-60101 4Gb Dual Port
PCI/PCI-X Fibre Channel Adapter (FC Port 1)
                /dev/fcd2
fc         3  0/7/1/1  fcd   CLAIMED   INTERFACE  HP AB379-60101 4Gb Dual Port
PCI/PCI-X Fibre Channel Adapter (FC Port 2)
                /dev/fcd3
```

---

## 10.4 Identifying WWPNS of HBAs

For zoning purposes, you need to know the respective WWPNS of the HBAs that are going to be connected to the DS8000. On HP-UX 11i, use the *Fibre Channel mass storage utility* (`fcmsutil`) to display the WWPNS, as shown in Example 10-3. The WWPNS is displayed in the middle of the listing.

*Example 10-3 Identifying WWPNS with fcmsutil*

---

```
# fcmsutil /dev/fcd0

Vendor ID is = 0x1077
Device ID is = 0x2422
PCI Sub-system Vendor ID is = 0x103C
PCI Sub-system ID is = 0x12D7
PCI Mode = PCI-X 266 MHz
ISP Code version = 4.0.90
ISP Chip version = 3
Topology = PTTOPT_FABRIC
Link Speed = 4Gb
Local N_Port_id is = 0x011d00
Previous N_Port_id is = None
N_Port Node World Wide Name = 0x5001438001321d79
N_Port Port World Wide Name = 0x5001438001321d78
Switch Port World Wide Name = 0x201d00051e349e2d
Switch Node World Wide Name = 0x100000051e349e2d
Driver state = ONLINE
Hardware Path is = 0/3/1/0
Maximum Frame Size = 2048
Driver-Firmware Dump Available = NO
Driver-Firmware Dump Timestamp = N/A
Driver Version = @(#) fcd B.11.31.0709 Jun 11 2007
```

---



The HP-UX device file for the HBA (`/dev/fcdX`) is a required parameter. Example 10-3 on page 184 shows that the HBAs are *dual port* devices, which is why you see two WWPNs for each HBA.

## 10.5 Configuring the HP-UX host for the DS8000

The first step in configuring a host connection on the DS8000 is to define a host connection using the graphical interface (DS GUI) or the command line interface (DS CLI). The `hosttype` parameter is required for this definition. Predefined, `hosttype HP`, automatically configures the DS8000 to present the DS8000 volumes in an HP-UX preferred method.

Keep in mind that when you create or assign LUNs and volumes, only LUN and volume IDs less than `x'3FFF'` are recognized in HP-UX 11i v3, as mentioned in “Notes on enumeration” on page 183.

### 10.5.1 Device special files

In HP-UX 11i, device special files (DSF) appear in the `/dev` directory like regular files, but they are special. They are associated with either real physical devices or pseudo devices, allowing access to the devices and drivers through a virtualization layer. One of the types of DSF is the persistent device special file, which references a device by a unique name and is not dependent on any specific hardware path. With HP-UX 11i Version 3, HP introduced several major changes, especially in the I/O area. One of these is an alternate naming scheme for device special files:

- ▶ Persistent DSFs are created independently from the underlying hardware path information and the number of physical paths to a device. The generic syntax of a persistent DSF for a LUN is `/dev/(r)disk/diskN`, with `N` being the device instance number. This kind of representation scheme is called the *agile view*; for example: `/dev/(r)disk/disk14`.
- ▶ The older format for DSFs, such as `/dev/(r)disk/c12t0d1`, is still available and is called *legacy DSF* in HP-UX 11iv3. Accordingly, this representation is called the *legacy view*.

### 10.5.2 Device discovery

After you configure the volumes on the DS8000, you can connect your host to the fabric, then discover the devices by using the `ioscan` command. Example 10-4 shows that the DS8000 devices were discovered successfully, but the devices cannot be used yet, because no special device file is available.

*Example 10-4 Discovered DS8000 devices without a special device file*

```
# ioscan -fnC disk
Class      I  H/W Path      Driver          S/W State  H/W Type    Description
-----
disk       1  0/0/2/1.0.16  UsbScsiAdaptor CLAIMED     DEVICE      USB SCSI Stack Adaptor
/dev/deviceFileSystem/Usb/MassStorage/dsk/disk@hp-1008+294=A60020000001
/dev/deviceFileSystem/Usb/MassStorage/rdisk/disk@hp-1008+294=A60020000001
disk       6  0/3/1/0.1.12.255.0.0.0  sdisk CLAIMED     DEVICE      IBM          2107900
disk       7  0/3/1/0.1.12.255.0.0.1  sdisk CLAIMED     DEVICE      IBM          2107900
disk       8  0/3/1/0.1.13.255.0.0.0  sdisk CLAIMED     DEVICE      IBM          2107900
disk       9  0/3/1/0.1.13.255.0.0.1  sdisk CLAIMED     DEVICE      IBM          2107900
disk       4  0/3/1/0.1.14.255.0.0.0  sdisk CLAIMED     DEVICE      IBM          2107900
disk       5  0/3/1/0.1.14.255.0.0.1  sdisk CLAIMED     DEVICE      IBM          2107900
disk       2  0/3/1/0.1.15.255.0.0.0  sdisk CLAIMED     DEVICE      IBM          2107900
disk       3  0/3/1/0.1.15.255.0.0.1  sdisk CLAIMED     DEVICE      IBM          2107900
disk      10  0/4/1/0.0.0.0.0.0      sdisk CLAIMED     DEVICE      HP           DH072ABAA6
```

```

/dev/dsk/c30t0d0    /dev/dsk/c30t0d0s2  /dev/rdisk/c30t0d0  /dev/rdisk/c30t0d0s2
/dev/dsk/c30t0d0s1 /dev/dsk/c30t0d0s3  /dev/rdisk/c30t0d0s1 /dev/rdisk/c30t0d0s3
disk    11 0/4/1/0.0.0.1.0      sdisk CLAIMED    DEVICE    HP      DH072ABAA6
/dev/dsk/c30t1d0    /dev/rdisk/c30t1d0

```

---

There are two options to create the missing special device file. The first option is to restart the host, which is disruptive. The alternative is to run the command **insf -eC disk**, which will reinstall the special device files for all devices of the class disk.

After creating the special device files, the **ioscan** command displays discovered DS8000 devices, as shown in Example 10-5.

*Example 10-5 Discovered DS8000 devices with a special device file*

```

# ioscan -fnC disk
Class      I  H/W Path      Driver      S/W State  H/W Type  Description
=====
disk      1  0/0/2/1.0.16  UsbScsiAdaptor  CLAIMED    DEVICE    USB SCSI Stack Adaptor
/dev/deviceFileSystem/Usb/MassStorage/dsk/disk@hp-1008+294=A60020000001
/dev/deviceFileSystem/Usb/MassStorage/rdsk/disk@hp-1008+294=A60020000001
disk      6  0/3/1/0.1.12.255.0.0.0  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c2t0d0    /dev/rdsk/c2t0d0
disk      7  0/3/1/0.1.12.255.0.0.1  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c2t0d1    /dev/rdsk/c2t0d1
disk      8  0/3/1/0.1.13.255.0.0.0  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c3t0d0    /dev/rdsk/c3t0d0
disk      9  0/3/1/0.1.13.255.0.0.1  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c3t0d1    /dev/rdsk/c3t0d1
disk      4  0/3/1/0.1.14.255.0.0.0  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c1t0d0    /dev/rdsk/c1t0d0
disk      5  0/3/1/0.1.14.255.0.0.1  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c1t0d1    /dev/rdsk/c1t0d1
disk      2  0/3/1/0.1.15.255.0.0.0  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c0t0d0    /dev/rdsk/c0t0d0
disk      3  0/3/1/0.1.15.255.0.0.1  sdisk CLAIMED    DEVICE    IBM      2107900
/dev/dsk/c0t0d1    /dev/rdsk/c0t0d1
disk     10  0/4/1/0.0.0.0.0.0      sdisk CLAIMED    DEVICE    HP      DH072ABAA6
/dev/dsk/c30t0d0    /dev/dsk/c30t0d0s2  /dev/rdisk/c30t0d0  /dev/rdisk/c30t0d0s2
/dev/dsk/c30t0d0s1 /dev/dsk/c30t0d0s3  /dev/rdisk/c30t0d0s1 /dev/rdisk/c30t0d0s3
disk     11  0/4/1/0.0.0.1.0      sdisk CLAIMED    DEVICE    HP      DH072ABAA6
/dev/dsk/c30t1d0    /dev/rdsk/c30t1d0

```

---

The **ioscan -Nm** command also shows the relationship between persistent DSFs and earlier DSFs, as shown in Example 10-6.

*Example 10-6 Relationship between persistent DSFs and earlier DSFs*

```

# ioscan -Nm dsf
Persistent DSF      Legacy DSF(s)
=====
/dev/rdisk/disk12    /dev/rdsk/c2t0d0
                    /dev/rdsk/c3t0d0
                    /dev/rdsk/c0t0d0
                    /dev/rdsk/c1t0d0
/dev/rdisk/disk13    /dev/rdsk/c2t0d1
                    /dev/rdsk/c3t0d1
                    /dev/rdsk/c0t0d1
                    /dev/rdsk/c1t0d1
/dev/rdisk/disk14    /dev/rdsk/c30t0d0

```

```
/dev/rdisk/disk14_p1    /dev/rdisk/c30t0d0s1
/dev/rdisk/disk14_p2    /dev/rdisk/c30t0d0s2
/dev/rdisk/disk14_p3    /dev/rdisk/c30t0d0s3
/dev/rdisk/disk15       /dev/rdisk/c30t1d0
```

---

The results of the `ioscan -Nm` command display all of the connection paths for *agile devices*. Example 10-6 on page 186 shows that the agile device `/dev/rdisk/disk12` is connected through four paths: `/dev/rdisk/c2t0d0` and `/dev/rdisk/c3t0d0` and `/dev/rdisk/c0t0d0` and `/dev/rdisk/c1t0d0`. When the volumes are visible, you can then create volume groups (VGs), logical volumes, and file systems.

## 10.6 Multipathing

The IBM SSIC indicates that the DS8000 supports native multipathing with HP-UX. Along with the operating system delivery, HP also ships a version of Symantec's Veritas Volume Manager (VxVM) which is a software product designed to enable logical management of physical devices. However, a license for Dynamic Multipathing (DMP) is not included. This implies that any I/O is handled in pass-through mode and is executed by native multipathing.

If you want to manage a multipathing design using VxVM, including DMP, you need to contact IBM and request a SCORE. For more information about SCORE, see 2.3, "Additional supported configurations" on page 13.

### 10.6.1 HP-UX multipathing solutions

Up to HP-UX 11iv2, PVLINKS was HP's multipathing solution on HP-UX and was built into the LVM. PVLINKS is a failover solution only, which means that it performs a path failover to an alternate path if the primary path is not available. PVLINKS does not offer load balancing, which allows I/O to be split among multiple paths while they are all available. PVLINKS still exists in HP-UX 11iv3, but IBM advises the use of native multipathing. However, information is provided in this section for instances where PVLINKS is still used.

**Multipathing:** Although HP PVLINKS is still available in HP-UX 11iv3, IBM advises the use of native multipathing as a more complete method for implementing multiple paths to a DS8000 for both failover and load-balancing.

To use PVLINKS for multipathing, add the HP-UX special device files that represent an additional path to the LUN on a newly created LVM volume group. The first special device file for a disk device becomes the primary path and the other device files become alternate paths.

Before you can add the HP-UX special device files to the LUN, create a new LVM volume group (the commands are illustrated in Example 10-7).

1. Create a volume group:
  - a. Create a directory for the volume group:

```
mkdir /dev/vg08
```
  - b. Create a group file within this directory:

```
mknod /dev/vg08/group c 64 0x080000
```
  - c. Prepare the LUNs being represented by the DSFs to be used in a LVM group:

```
pvcreate /dev/rdsk/c11t0d1
pvcreate /dev/rdsk/c12t0d1
```

*Example 10-7 Volume group creation with earlier DSFs*

---

```
# vgcreate -A y -x y -l 255 -p 100 -s 16 /dev/vg08 /dev/dsk/c11t0d1
/dev/dsk/c15t0d1 /dev/dsk/c12t0d1 /dev/dsk/c13t0d1
Volume group "/dev/vg08" has been successfully created.
Volume Group configuration for /dev/vg08 has been saved in /etc/lvmconf/vg08.conf
# vdisplay -v vg08
--- Volume groups ---
VG Name                /dev/vg08
VG Write Access        read/write
VG Status               available
Max LV                 255
Cur LV                0
Open LV                0
Max PV                 100
Cur PV                2
Act PV                 2
Max PE per PV          1016
VGDA                   4
PE Size (MBytes)       16
Total PE               1534
Alloc PE               0
Free PE                1534
Total PVG              0
Total Spare PVs        0
Total Spare PVs in use 0

--- Physical volumes ---
PV Name                /dev/dsk/c11t0d1
PV Name                /dev/dsk/c15t0d1 Alternate Link
PV Status              available
Total PE               767
Free PE                767
Autoswitch             On

PV Name                /dev/dsk/c12t0d1
PV Name                /dev/dsk/c13t0d1 Alternate Link
PV Status              available
Total PE               767
Free PE                767
Autoswitch             On
```

---

In Example 10-7 here, the volume group, vg08, consists of two LUNs c11t0d1 and c12t0d1, each of which is dual-pathed, also called Alternate Linked.

2. Create the logical volumes in Example 10-7, which are LVM objects that the operating system can work with. To create the logical volumes, create file systems and mount them, by completing the following steps:
  - a. Create a logical volume, which creates a 50 MB logical volume named lv011:

```
lvcreate -L 50 /dev/vg08
```
  - b. Create a file system:

```
newfs -F vxfs /dev/vg08/r1lv011
```

c. Mount the file system:

```
mount /dev/vg01/lvo11 /mount1
```

This step assumes that the directory where the file system will be mounted already exists.

3. Create the volume group. To use the new agile addressing with a volume group, specify the persistent DSFs in the **vgcreate** command, as shown in Example 10-8.

*Example 10-8 Volume group creation with Persistent DSFs*

---

```
# vgcreate -A y -x y -l 255 -p 100 -s 16 /dev/vgagile /dev/disk/disk12  
/dev/disk/disk13 /dev/disk/disk14
```

Volume group "/dev/vgagile" has been successfully created.

Volume Group configuration for /dev/vgagile has been saved in  
/etc/lvmconf/vgagile.conf

```
# vgdisplay -v vgagile
```

```
--- Volume groups ---
```

VG Name	/dev/vgagile
VG Write Access	read/write
VG Status	available
Max LV	255
Cur LV	0
Open LV	0
Max PV	100
Cur PV	3
Act PV	3
Max PE per PV	1016
VGDA	6
PE Size (MBytes)	16
Total PE	2301
Alloc PE	0
Free PE	2301
Total PVG	0
Total Spare PVs	0
Total Spare PVs in use	0

```
--- Physical volumes ---
```

PV Name	/dev/disk/disk12
PV Status	available
Total PE	767
Free PE	767
Autoswitch	On

PV Name	/dev/disk/disk13
PV Status	available
Total PE	767
Free PE	767
Autoswitch	On

PV Name	/dev/disk/disk14
PV Status	available
Total PE	767
Free PE	767
Autoswitch	On

---

The **vgcreate** command takes many arguments, which can be quite confusing. A brief breakdown of the options used in these examples is as follows:

- A *y*     Setting this options to *y* will automatically back up the changes made to the volume group. This causes the **vgcfgbackup** command to be executed after the **vgcreate** command.
- x *y*     Setting this options to *y* allows the allocation of more physical extents on the physical volume.
- l 255    This argument determines the maximum number of logical volumes that the volume group can contain. The allowable range is 1 to 255, the default is 255.
- p 100    This argument determines the maximum number of physical volumes that the volume group can contain. The allowable range is 1 to 255, the default is 16.
- s 16     This argument determines the number of megabytes in each physical extent. The allowable range is 1 to 256 (expressed in powers of 2: 2, 4, 8, 16). The default is 4MB.

**Caution:** Do **not** use the **-f** option on the **vgcreate** command. It forces the creation of a volume group with a physical volume that has alternate blocks already allocated. It can be a dangerous command because the potential for data corruption exists.

## 10.6.2 Exposing link errors with HP-UX

If a Fibre Channel link from the HP-UX system to the DS8000 fails, native multipathing will automatically take the path offline. It should also automatically bring the path back online after it is established again. Example 10-9 shows the messages that are posted to the `syslog` when a link goes offline and then comes back online. On HP-UX 11iv3, this logfile is commonly found as `/var/adm/syslog/syslog.log` and it contains entries from all subsystems, so parsing through it might take a while. The keywords to look for are: *Link Dead*, *offline*, and *online*.

*Example 10-9 Entries of syslog.log by HP's native multipathing solution for Fibre Channel link failures*

```
May 30 12:38:03 rx6600-1 vmunix: 0/3/1/0: Fibre Channel Driver received Link Dead
Notification.
May 30 12:38:03 rx6600-1 vmunix: class : tgtpath, instance 3
May 30 12:38:03 rx6600-1 vmunix: Target path (class=tgtpath, instance=3) has gone
offline. The target path h/w path is 0/3/1/0.0x5005076303010143
.....(Later that same minute...)
May 30 12:38:08 rx6600-1 vmunix: Target path (class=tgtpath, instance=4) has gone
online. The target path h/w path is 0/3/1/0.0x5005076303080143
May 30 12:38:08 rx6600-1 vmunix: class : tgtpath, instance 5
```

A system that is experiencing I/O paths frequently dropping offline and coming back online will generate messages in the `syslog` file similar to those in Example 10-10.

*Example 10-10 Syslog entries for a large number of I/O errors*

```
May 30 12:38:08 rx6600-1 vmunix: DIAGNOSTIC SYSTEM WARNING:
May 30 12:38:08 rx6600-1 vmunix:     The diagnostic logging facility has started
receiving excessive
May 30 12:38:08 rx6600-1 vmunix:     errors from the I/O subsystem. I/O error
entries will be lost
```

## 10.7 Working with VERITAS Volume Manager on HP-UX

When working with HP-UX 11i 3, you can choose from two volume managers:

- ▶ HP Logical Volume Manager (LVM)
- ▶ VERITAS Volume Manager (VxVM): I/O is handled in pass-through mode and executed by native multipathing, *not* by DMP.

According to HP, both volume managers can coexist on an HP-UX server. You can use both simultaneously, on separate physical disks, but usually you will choose one or the other and use it exclusively. For more information, see the *HP-UX System Administrator's Guide: Overview HP-UX 11i Version 3* at this website:

<http://bizsupport1.austin.hp.com/bc/docs/support/SupportManual/c02281492/c02281492.pdf>

For the configuration of the DS8000 logical volumes on HP-UX with LVM, see 10.6.1, "HP-UX multipathing solutions" on page 187.

Example 10-11 shows the initialization of disks for VxVM use, and the creation of a disk group with the `vxdiskadm` utility.

*Example 10-11 Disk initialization and disk group creation with vxdiskadm*

---

```
# vxdisk list
DEVICE      TYPE      DISK      GROUP      STATUS
c2t0d0      auto:none -          -          online invalid
c2t1d0      auto:none -          -          online invalid
c10t0d1     auto:none -          -          online invalid
c10t6d0     auto:none -          -          online invalid
c10t6d1     auto:none -          -          online invalid
c10t6d2     auto:none -          -          online invalid

# vxdiskadm
Volume Manager Support Operations
Menu: VolumeManager/Disk

1      Add or initialize one or more disks
2      Remove a disk
3      Remove a disk for replacement
4      Replace a failed or removed disk
5      Mirror volumes on a disk
6      Move volumes from a disk
7      Enable access to (import) a disk group
8      Remove access to (deport) a disk group
9      Enable (online) a disk device
10     Disable (offline) a disk device
11     Mark a disk as a spare for a disk group
12     Turn off the spare flag on a disk
13     Remove (deport) and destroy a disk group
14     Unrelocate subdisks back to a disk
15     Exclude a disk from hot-relocation use
16     Make a disk available for hot-relocation use
17     Prevent multipathing/Suppress devices from VxVM's view
18     Allow multipathing/Unsuppress devices from VxVM's view
19     List currently suppressed/non-multipathed devices
20     Change the disk naming scheme
```

21 Change/Display the default disk layouts  
list List disk information

? Display help about menu  
?? Display help about the menuing system  
q Exit from menus

Select an operation to perform: 1

Add or initialize disks  
Menu: VolumeManager/Disk/AddDisks

Use this operation to add one or more disks to a disk group. You can add the selected disks to an existing disk group or to a new disk group that will be created as a part of the operation. The selected disks may also be added to a disk group as spares. Or they may be added as nohotuses to be excluded from hot-relocation use. The selected disks may also be initialized without adding them to a disk group leaving the disks available for use as replacement disks.

More than one disk or pattern may be entered at the prompt. Here are some disk selection examples:

all: all disks  
c3 c4t2: all disks on both controller 3 and controller 4, target 2  
c3t4d2: a single disk (in the c#t#d# naming scheme)  
xyz\_0: a single disk (in the enclosure based naming scheme)  
xyz\_: all disks on the enclosure whose name is xyz

Select disk devices to add: [<pattern-list>,all,list,q,?] c10t6d0 c10t6d1

Here are the disks selected. Output format: [Device\_Name]

c10t6d0 c10t6d1

Continue operation? [y,n,q,?] (default: y) y

You can choose to add these disks to an existing disk group, a new disk group, or you can leave these disks available for use by future add or replacement operations. To create a new disk group, select a disk group name that does not yet exist. To leave the disks available for future use, specify a disk group name of "none".

Which disk group [<group>,none,list,q,?] (default: none) dg01

There is no active disk group named dg01.

Create a new group named dg01? [y,n,q,?] (default: y)

Create the disk group as a CDS disk group? [y,n,q,?] (default: y) n

Use default disk names for these disks? [y,n,q,?] (default: y)



Add disks as spare disks for dg01? [y,n,q,?] (default: n) n

Exclude disks from hot-relocation use? [y,n,q,?] (default: n)

A new disk group will be created named dg01 and the selected disks will be added to the disk group with default disk names.

c10t6d0 c10t6d1

Continue with operation? [y,n,q,?] (default: y)

Do you want to use the default layout for all disks being initialized?  
[y,n,q,?] (default: y) n

Do you want to use the same layout for all disks being initialized?  
[y,n,q,?] (default: y)

Enter the desired format [cdsdisk,hpdisk,q,?] (default: cdsdisk) hpdisk

Enter the desired format [cdsdisk,hpdisk,q,?] (default: cdsdisk) hpdisk

Enter desired private region length  
[<privlen>,q,?] (default: 1024)

Initializing device c10t6d0.

Initializing device c10t6d1.

VxVM NOTICE V-5-2-120

Creating a new disk group named dg01 containing the disk device c10t6d0 with the name dg0101.

VxVM NOTICE V-5-2-88

Adding disk device c10t6d1 to disk group dg01 with disk name dg0102.

Add or initialize other disks? [y,n,q,?] (default: n) n

**# vxdisk list**

DEVICE	TYPE	DISK	GROUP	STATUS
c2t0d0	auto:none	-	-	online invalid
c2t1d0	auto:none	-	-	online invalid
c10t0d1	auto:none	-	-	online invalid
c10t6d0	auto:hpdisk	dg0101	dg01	online
c10t6d1	auto:hpdisk	dg0102	dg01	online
c10t6d2	auto:none	-	-	online invalid

---

The graphical equivalent for the `vxdiskadm` utility is the VERITAS Enterprise Administrator (VEA). Figure 10-1 shows the presentation of disks, using this GUI. After creating the disk groups and the VxVM disks, create file systems and mount them, as shown in Figure 10-1.

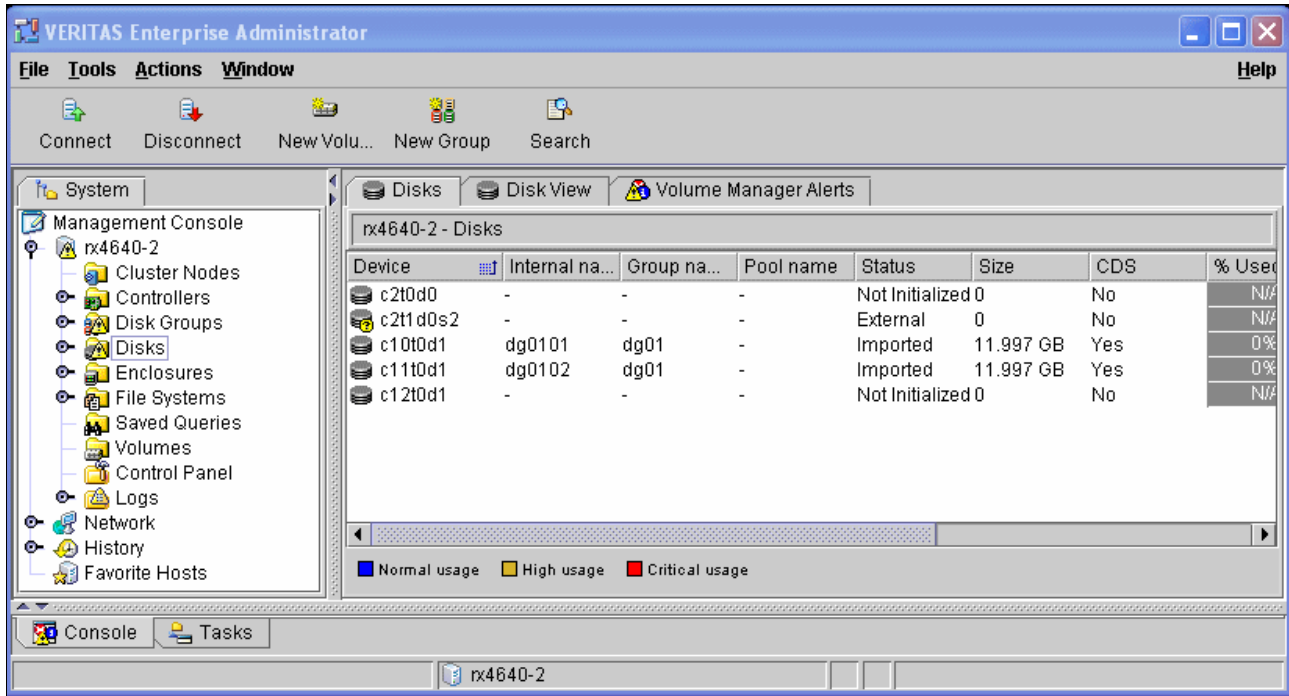


Figure 10-1 Disk presentation by VERITAS Enterprise Administrator

## 10.8 Working with LUNs

This section provides information and examples regarding expanding LUNs as well as working with large LUNs. The examples are based on a system that has a volume group named `vg01` and a logical volume named `lv011`.

### 10.8.1 Expanding LUNs

First, examine the characteristics of the volume group using the `vgdisplay` command. Note the section at the end of Example 10-12 where the information about the logical and physical volumes is displayed.

*Example 10-12 Displaying volume group details*

```
# vgdisplay -v vg01
--- Volume groups ---
VG Name                /dev/vg01
VG Write Access        read/write
VG Status              available
Max LV                 255
Cur LV                1
Open LV               1
Max PV                 16
Cur PV                1
Act PV                1
Max PE per PV         8000
```

```

VGDA                2
PE Size (Mbytes)    4
Total PE            3071
Alloc PE            3000
Free PE             71
Total PVG           0
Total Spare PVs     0
Total Spare PVs in use 0

  --- Logical volumes ---
  LV Name            /dev/vg01/lvo11
  LV Status          available/syncd
  LV Size (Mbytes)   12000
  Current LE         3000
  Allocated PE       3000
  Used PV            1

  --- Physical volumes ---
  PV Name            /dev/disk/disk14
  PV Status          available
  Total PE           3071
  Free PE            71
  Autoswitch        On

```

---

Now the details regarding the physical volume(s) can be examined by using the **pvdisk** command as shown in Example 10-13.

*Example 10-13 Displaying physical volume details*

---

```

# pvdisk /dev/disk/disk14
--- Physical volumes ---
PV Name            /dev/disk/disk14
VG Name            /dev/vg01
PV Status          available
Allocatable        yes
VGDA               2
Cur LV            1
PE Size (Mbytes)   4
Total PE           3071
Free PE            71
Allocated PE       3000
Stale PE           0
IO Timeout (Seconds) default
Autoswitch         On

```

---

At this point, it is time to increase the size of the DS8000 LUN. After the change is made, take the following steps at the HP-UX operating system level:

1. Unmount the file system that has `lv011` mounted.
2. Deactivate volume group `vg00`:

```
vgchange -a n vg00
```

3. Use the **vgmodify** command to let HP-UX know that the LUN size was altered, as shown in Example 10-14.

*Example 10-14 Applying the vgmodify command*

---

```
# vgmodify /dev/vg01
Current Volume Group settings:
                                Max LV      255
                                Max PV      16
                                Max PE per PV 8000
                                PE Size (Mbytes) 4
                                VGRA Size (Kbytes) 1088
"/dev/rdisk/disk14" size changed from 12582912 to 18874368kb
An update to the Volume Group IS required
New Volume Group settings:
                                Max LV      255
                                Max PV      16
                                Max PE per PV 8000
                                PE Size (Mbytes) 4
                                VGRA Size (Kbytes) 1088
New Volume Group configuration for "/dev/vg01" has been saved in
"/etc/lvmconf/vg01.conf"
Old Volume Group configuration for "/dev/vg01" has been saved in
"/etc/lvmconf/vg01.conf.old"
Starting the modification by writing to all Physical Volumes
Applying the configuration to all Physical Volumes from "/etc/lvmconf/vg01.conf"
Completed the modification process.
New Volume Group configuration for "/dev/vg01" has been saved in
"/etc/lvmconf/vg01.conf.old"
Volume group "/dev/vg01" has been successfully changed.
```

---

4. Reactivate volume group vg01:

**vgchange -a y vg01**

Run the **pvdisk** command again. Example 10-15 reflects the new physical volume details, indicating that HP-UX recognizes the LUN resizing.

*Example 10-15 inquiring the physical volume*

---

```
# pvdisk /dev/disk/disk14
--- Physical volumes ---
PV Name           /dev/disk/disk14
VG Name           /dev/vg01
PV Status         available
Allocatable       yes
VGDA              2
Cur LV           1
PE Size (Mbytes)  4
Total PE          4607
Free PE           1607
Allocated PE      3000
Stale PE          0
IO Timeout (Seconds) default
Autoswitch        On
```

---

5. Extend the size of the LVM logical volume by using the `lvextend` command as shown in Example 10-16.

*Example 10-16 Extending the size of the LVM logical volume*

---

```
# lvextend -l 4600 /dev/vg00/lvol1
Logical volume "/dev/vg00/lvol1" has been successfully extended.
Volume Group configuration for /dev/vg01 has been saved in /etc/lvmconf/vg01.conf
```

---

6. Increase the size of the file system and display the mounted file systems, as shown in Example 10-17.

*Example 10-17 Increasing the file system size*

---

```
# fsadm -F vxfs -b 18400000 /vmnt
vxfs fsadm: V-3-23585: /dev/vg01/r1vol1 is currently 12288000 sectors - size
will be increased
# bdf
Filesystem          kbytes   used   avail %used Mounted on
/dev/vg00/lvol3     1048576 312368 730480   30% /
/dev/vg00/lvol1     1835008 151848 1670048    8% /stand
/dev/vg00/lvol8     8912896 3442712 5430112   39% /var
/dev/vg00/lvol7     3964928 2844776 1111440   72% /usr
/dev/vg00/lvol4       524288  24136  496304    5% /tmp
/dev/vg00/lvol6     7798784 3582480 4183392   46% /opt
/dev/vg00/lvol5      131072  123136    7928   94% /home
DevFS                 3         3         0 100% /dev/deviceFileSystem
/dev/vg01/lvol1     18420000 21603 17248505    0% /vmnt
```

---

**Tip:** To run the `fsadm` command successfully, you need the appropriate license. If you do not have the license installed, you will see the following message:

```
“UX:vxfs fsadm: ERROR: V-3-25255: fsadm: You don't have a license to run this
program.”
```

The number following the `-b` option can be calculated as follows:

- The new size of the logical volume equals 4,600 physical extents (PEs).
- The PE size equals 4 MB. Therefore, in total, the new size equals to 4 MB times 4,600 PEs, which equals 18,400 MB, or 18,400,000 bytes.

## 10.8.2 Working with large LUNs

Large LUNs are multiple terabytes in size. The information presented up to this point cannot be applied when working with large LUNs. A new set of parameters and options are required. The rest of this chapter provides information about working with large LUNs.

Example 10-18 depicts the following situations:

- ▶ The `lvmadm -t` command demonstrates attaching support for the Version 2.1 volume group.
- ▶ The properties shown in **bold** highlight the differences between volume group versions 2.1, 2.0, and 1.0.

*Example 10-18 Showing the properties of separate volume group versions*

---

```
# lvmadm -t
--- LVM Limits ---
VG Version                1.0
Max VG Size (Tbytes)      510
Max LV Size (Tbytes)      16
Max PV Size (Tbytes)     2
Max VGs                   256
Max LVs                   255
Max PVs                  255
Max Mirrors               2
Max Stripes               255
Max Stripe Size (Kbytes)  32768
Max LXs per LV            65535
Max PXs per PV            65535
Max Extent Size (Mbytes)  256

VG Version                2.0
Max VG Size (Tbytes)      2048
Max LV Size (Tbytes)      256
Max PV Size (Tbytes)     16
Max VGs                   512
Max LVs                   511
Max PVs                  511
Max Mirrors               5
Max Stripes               511
Max Stripe Size (Kbytes)  262144
Max LXs per LV            33554432
Max PXs per PV            16777216
Max Extent Size (Mbytes)  256

VG Version                2.1
Max VG Size (Tbytes)      2048
Max LV Size (Tbytes)      256
Max PV Size (Tbytes)     16
Max VGs                   2048
Max LVs                   2047
Max PVs                  2048
Max Mirrors               5
Max Stripes               511
Max Stripe Size (Kbytes)  262144
Max LXs per LV            33554432
Max PXs per PV            16777216
Max Extent Size (Mbytes)  256
```

---

To create a volume group for large LUNs, complete the following steps:

1. Assign a large LUN, for example, 17 TB, to HP-UX and initialize the LUN by entering the **pvcreate** command, as shown in Example 10-19. The operating system will communicate that the traditional volume group version 1.0 cannot handle large LUNs.
2. Define a group special file using the **mknod** command, specifying a major number, such as 128, instead of 64.
3. Create the volume group by entering the **vgcreate** command, using the **-V** option for the Version 2.1 volume group.

*Example 10-19 Preparing a large LUN*

---

```
# pvcreate -f /dev/rdisk/disk1078
pvcreate: Warning: The physical volume "/dev/rdisk/disk1078", is initialized
to use only the first 2147483647KB of disk space in volume group version 1.0.
On volume group version 2.x, the physical volume size will be recalculated to
use full size upto a maximum of 17179869184KB.
Physical volume "/dev/rdisk/disk1078" has been successfully created.
# mnod /dev/vgMZOutCTest010/group c 128 0x0a0000
# vgcreate -V 2.1 -s 256 -S 32t /dev/vgMZOutCTest010 /dev/disk/disk1078
Volume group "/dev/vgMZOutCTest010" has been successfully created.
Volume Group configuration for /dev/vgMZOutCTest010 has been saved in
/etc/lvmconf/vgMZOutCTest010.conf
```

---

4. Set the following parameters:
  - a. **-s 256**, (lowercase/small s) sets the Physical Extents size to 256, the default is 4.
  - b. **-S 32t**, (uppercase/large S) sets the Maximum Volume Group size to 32t.

**Tip:** Upon entering the **vgcreate** command, if you specify a major number, such as 64, you see the following message:

```
vgcreate: Error: The major number for the group file
"/dev/vgMZOutCTest010/group" corresponds to volume group version 1.0.
```

5. Create the logical volumes and the corresponding file systems.

**Requirement:** A license is required to work with large lvols. If you do not have a license, you see the following message:

```
UX:vxfs mkfs: ERROR: V-3-26141: mkfs: You don't have a license to create a
file system of size > 2147483648 sectors (2048 GB)
```







## IBM i considerations

This chapter provides the specifics for the IBM System Storage DS8000 series system attachment to IBM i.

The following topics are covered:

- ▶ Supported environment
- ▶ Using Fibre Channel adapters
- ▶ Sizing and numbering of LUNs
- ▶ Using multipath
- ▶ Configuration guidelines
- ▶ Booting from SAN
- ▶ Installing IBM i with boot from SAN through VIOS NPIV
- ▶ Migrating

For more information about these topics, see *IBM i and IBM System Storage: A Guide to Implementing External Disks on IBM i*, SG24-7120.

## 11.1 Supported environment

This section describes the hardware and software prerequisites for attaching the DS8000 to an IBM i system.

**Scope:** This chapter provides a high level summary of needed servers and software levels to connect DS8000 to the IBM i. For detailed hardware and software prerequisites, see the System Storage Interoperation Center (SSIC) and the BladeCenter Interoperability Guide (BIG) listed in 11.1.4, “Useful websites” on page 204.

The following attachment methods are supported for IBM i:

- ▶ *Native attachment* by connecting to the DS8000 using physical adapters in IBM i (note that IBM i resides in a partition of a POWER system or in a former System i model)
- ▶ Attachment with virtual input and output server, (*VIOS*) node port ID virtualization (*NPIV*)
- ▶ Attachment with *VIOS* using *virtual SCSI* adapters

### 11.1.1 System hardware

*Native connection* is supported on IBM POWER7®, IBM POWER6® and POWER5 systems that allow an IBM i partition, System i models 270, 520, 525, 550, 570, 595, 800, 810, 820, 825, 830, 840, 870, 890, and IBM System p models 9117-570, 9119-590, and 9119-595 with feature 9411-100 that allow IBM i in a partition.

Attachment with *VIOS NPIV* and attachment with *VIOS with VSCSI* are supported on POWER7 and POWER6 systems that allow an IBM i partition. For more information about POWER systems, see the “IBM Power Systems Hardware Information Center” as listed in 11.1.4, “Useful websites” on page 204.

Attachment of DS8000 with *VIOS virtual SCSI* or *VIOS NPIV* is supported for IBM i in POWER6 and POWER7 based BladeCenters that allow for an IBM i LPAR.

**BladeCenters:** BladeCenters in chassis S support only VIOS VSCSI connection of DS8000 to IBM i, while BladeCenters in chassis H provide both VIOS virtual SCSI and VIOS NPIV connection.

For more information about supported attachments, see the “BladeCenter Interoperability Guide” as listed in 11.1.4, “Useful websites” on page 204.

For more information about the BladeCenters, see the *IBM Power Blade servers* listed in 11.1.4, “Useful websites” on page 204.

### 11.1.2 Software

This section covers the various software levels.

#### IBM i levels

The terminology of IBM i level is *Version.Release*; for example, V7R1, or V7.1, stands for Version 7 Release 1. Before IBM i V7.1, the software levels between the releases were denoted by modification levels, for example V6.1.1 stands for Version 6, Release 1, Modification level 1. When specifying support for a certain function just by version and release, it means that any modification level supports it. However, if support for a function needs a certain modification level, that level is specified.

From IBM i V7.1 on, IBM i modification levels are replaced by a new release delivery mechanism called a *Technology Refresh*. A Technology Refresh (TR) is delivered as a PTF Group for a base release. When specifying support for a certain function just by version and release, it means that any TR supports it. However, if support for a function needs a certain TR level, that level is specified. Here are some requirements:

- ▶ IBM i software levels V5.4, V6.1, and V7.1 support native attachment of the DS8000.
- ▶ Connection of DS8000 to IBM i with *VIOS NPIV* requires IBM i V6.1.1, V7.1.
- ▶ Connection of DS8000 to IBM i with *VIOS VSCSI* requires IBM i level V6.1 or V7.1.

### VIOS levels

For connecting DS8000 to IBM i with *VIOS NPIV*, you need the following VIOS levels:

- ▶ In POWER server, VIOS V2.1.2 or higher
- ▶ In BladeCenter, VIOS V2.2 or higher.

Attachment of IBM i with *VIOS VSCSI* requires VIOS V1.5.0 or higher.

### Needed levels for multipath

Multipath for *native* connection is supported on all IBM i levels that are listed to support native attaching.

When using *VIOS VSCSI* or *NPIV*, the multipath is achieved by connecting DS8000 to IBM i with multiple VIOS, each path through one VIOS. Multipath with VIOS in NPIV or VSCSI requires VIOS level V2.1.2 or higher.

### Console management levels

Make sure that the Hardware Management Console (HMC) of POWER server is on the latest firmware level, or that the Systems Director Management Console (SDMC) of POWER server or BladeCenter is on the latest software level.

## 11.1.3 Overview of hardware and software requirements

This section presents two tables that list hardware and software combinations for the connection methods. The tables provide a quick overview of the possible combinations that are currently supported. For more detailed, up-to-date information, see the websites listed in section 11.1.4, “Useful websites” on page 204.

Table 11-1 shows an overview of supported POWER and Blade models, and IBM i software levels.

Table 11-1 Supported hardware and software levels

DS8800 attach	Server	i 5.4	i 6.1	i 7.1
Native VIOS NPIV VIOS VSCSI	POWER7		Yes <sup>a</sup>	Yes
VIOS NPIV VIOS VSCSI	Blade servers based on POWER7		Yes <sup>a b</sup>	Yes <sup>b</sup>
Native VIOS NPIV VIOS VSCSI	IBM POWER6+™		Yes <sup>a</sup>	Yes

DS8800 attach	Server	i 5.4	i 6.1	i 7.1
VIOS NPIV VIOS VSCSI	Blade servers based on POWER6		Yes <sup>a b</sup>	Yes <sup>b</sup>
Native	POWER6	Yes	Yes	Yes
VIOS NPIV VIOS VSCSI	POWER6		Yes <sup>a</sup>	Yes
Native	POWER5 POWER5+	Yes	Yes	Yes
Native	System i models, IBM System p® models for IBM i in an LPAR	Yes	Yes	

a. The support of VIOS NPIV requires IBM i V6.1.1 on either POWER system or BladeCenter

b. BladeCenters in chassis S support only VIOS VSCSI connection of DS8000 to IBM i

Table 11-2 shows the minimal VIOS levels needed for NPIV support, VSCSI support, and multipath with two VIOS.

Table 11-2 VIOS levels

Methods of attaching with VIOS	POWER system or Blade	Minimal VIOS level for connection	Minimal VIOS level for multipath
VIOS NPIV	POWER7	V2.1.2	V2.1.2
	POWER6	V2.1.2	V2.1.2
	POWER Blade <sup>a</sup>	V2.2	V2.1.2 <sup>b</sup>
VIOS VSCSI	POWER7	V1.5.0	V2.1.2
	POWER6	V1.5.0	V2.1.2
	POWER5	V1.5.0	V2.1.2
	POWER Blade	V1.5.0	V2.1.2 <sup>b</sup>

a. BladeCenters in chassis S support only VIOS VSCSI connection of DS8000 to IBM i

b. Two VIOS in a BladeCenter are supported providing that the BladeCenter is managed by Systems Director Management Console and that there are enough Fibre Channel and Ethernet adapters for each VIOS.

### 11.1.4 Useful websites

The following websites provide up-to-date information about the environments used when connecting DS8800 to IBM i:

- ▶ System i Storage Solutions:

<http://www.ibm.com/systems/i/hardware/storage/index.html>

- ▶ Virtualization with IBM i, PowerVM, and Power Systems:

<http://www.ibm.com/systems/i/os/>

- ▶ IBM Power Systems Hardware Information Center:

[http://publib.boulder.ibm.com/infocenter/powersys/v3r1m5/index.jsp?topic=/iphdx/550\\_m50\\_landing.htm](http://publib.boulder.ibm.com/infocenter/powersys/v3r1m5/index.jsp?topic=/iphdx/550_m50_landing.htm)

- ▶ IBM Power Blade servers:  
<http://www.ibm.com/systems/power/hardware/blades/index.html>
- ▶ IBM i and System i Information Center:  
<http://publib.boulder.ibm.com/iseriess/>
- ▶ IBM Support Portal:  
<http://www.ibm.com/support/entry/portal/>
- ▶ System Storage Interoperation Center (SSIC):  
<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>
- ▶ BladeCenter Interoperability Guide (BIG), accessible on the following website:  
<http://www-947.ibm.com/support/entry/portal/docdisplay?lnocid=MIGR-5073016>
- ▶ To check for the latest IBM i program temporary fix (PTF):  
<http://www.ibm.com/support/entry/portal/>

## 11.2 Using Fibre Channel adapters

This section presents an overview of currently supported adapters in POWER systems or POWER based BladeCenters when connecting IBM i to a DS8000. Also provided in this section is the number of LUNs that you can assign to an adapter and the queue depth on a LUN with SCSI command tag queuing.

For more detailed, up-to-date information, see the websites listed in 11.1.4, “Useful websites” on page 204.

**Tip:** A Fibre Channel adapter in IBM i is also known as an input output adapter (IOA).

### 11.2.1 Native attachment

The following Fibre Channel adapters are supported in IBM i system to connect DS8000 series in native mode:

- ▶ #2787 PCI-X Fibre Channel disk controller
- ▶ #5760 4 GB Fibre Channel disk controller PCI-x
- ▶ #5749 4 GB Fibre Channel disk controller PCI-x
- ▶ #5774 4 GB Fibre Channel disk controller PCI Express
- ▶ #5735 8 GB Fibre Channel disk controller PCI Express
- ▶ #5273 PCIe LP 8 Gb 2-Port Fibre Channel adapter
- ▶ #5276 PCIe LP 4 Gb 2-Port Fibre Channel adapter

Adapters with feature numbers #2787 and #5760 are 1-port, input output processor (IOP) based adapters that can address up to 32 logical volumes. The other adapters are 2-port IOP-less adapters that can address up to 64 volumes per port.

The Low Profile adapters feature number #5273 and #5276 are supported in POWER models that support Low Profile PCI slots. For more information, see “PCI adapter placement rules and slot priorities” in the “IBM Power Systems Hardware Information Center” as listed in 11.1.4, “Useful websites” on page 204.

**Tip:** You can assign up to 32 LUNs for two or more IOP-based adapters in multipath, or up to 64 LUNs for two or more ports in IOP-less adapters. The ports in multipath must belong to separate IOP-less adapters.

IBM i V6R1 and later, in combination with IOP-less adapters, provides SCSI command tag queuing support on DS8000 systems with the supported queue depth being six input and output operations.

## 11.2.2 Attachment with VIOS NPIV

Connecting DS8000 series to an IBM i with VIOS NPIV in POWER servers requires one of the following adapters in VIOS:

- ▶ #5735 8 GB Fibre Channel disk controller PCI Express
- ▶ #5273 PCIe LP 8 Gb 2-Port Fibre Channel adapter
- ▶ #5729 PCIe2 8 Gb 4-port Fibre Channel adapter

For Fibre Channel over Ethernet connection:

- ▶ #5708 10 Gb FCoE PCIe Dual Port adapter
- ▶ #5270 PCIe LP 10 Gb FCoE 2-port adapter

NPIV connection requires an NPIV enabled SAN switches for connection of DS8000.

Connecting DS8000 series to an IBM i through VIOS NPIV in IBM BladeCenter requires one of the following adapters:

- ▶ #8271 QLogic 8 GB Fibre Channel Expansion Card (CFFh)
- ▶ #8242 QLogic 8 GB Fibre Channel Card (CIOv)
- ▶ #8240 Emulex 8 GB Fibre Channel Expansion Card (CIOv)

For Fibre Channel over Ethernet connection:

- ▶ #8275 QLogic 2-port 10 Gb Converged Network Adapter (CFFh)

With VIOS NPIV, up to 64 LUNs can be assigned to a port in a virtual Fibre Channel adapter in IBM i. The IBM i with virtual Fibre Channel adapters used with VIOS NPIV provides SCSI command tag queuing support of DS8800 system with the queue depth being 6 I/O operations.

## 11.2.3 Attachment with VIOS VSCSI

Connecting DS8000 series to an IBM i with VIOS virtual SCSI in POWER servers requires one of the following adapters in VIOS:

- ▶ #5774 4 GB Fibre Channel Disk Controller PCI Express
- ▶ #5735 8 GB Fibre Channel Disk Controller PCI Express

Attaching DS8000 to an IBM i through VIOS VSCSI in BladeCenter requires one of the following adapters:

- ▶ Any of the 8 GB adapters and FCoE adapter listed under the VIOS NPIV in BladeCenter requirements
- ▶ 8252 QLogic Ethernet and 4 GB Fibre Channel Expansion Card (CFFh)
- ▶ 8251 Emulex 4 GB Fibre Channel Expansion Card (CFFv)
- ▶ 8248 QLogic 4 GB Fibre Channel Expansion Card (CFFv)

**CFFv, CFFh, and CIOv:** CFFv stands for combination form factor vertical. CFFh stands for combination form factor horizontal, and CIOv stands for combination I/O form factor vertical.

Up to 16 LUNs can be assigned to a virtual SCSI adapter in IBM i when connecting DS8000 series to an IBM i client. SCSI command tag queuing is supported by IBM i with a VIOS VSCSI connection and queue depth 32 input and output operations.

## 11.2.4 Overview of the number of LUNs per adapter and the queue depth

Table 11-3 provides an overview of the number of LUNs and supported queue depth on adapters at different connection methods:

Table 11-3 Maximal number of LUNs per port and Queue depth

Connection method	Max number of LUNs per port, or max number per two or more ports in multipath	Max number of LUNs per virtual FC adapter	Max number of LUNs per virtual SCSI adapter	Queue depth to a LUN
Native with IOP-based adapters	32			0
Native with IOP-less adapters	64			6 <sup>a</sup>
VIOS_NPIV		64		6
VIOS Virtual SCSI			16	32

a. IBM i level v6.1 or later is needed to support SCSI command tag queuing

## 11.3 Sizing and implementation guidelines

When sizing DS8000 resources for use with IBM i, use the Disk Magic tool to model the response times and utilization values. Also, apply the sizing guidelines, such as the number of DS8000 ranks, the number of IBM i adapters, size of LUNs, and so on. It is best to apply the sizing guidelines before modelling with Disk Magic, so that you can specify the values obtained by sizing, such as number of ranks, size of LUNs, and so on, when you start modelling your DS8000.

In order to model with Disk Magic and to apply the sizing before modelling, you need to collect performance data on IBM i. The data must be collected with IBM i Collection Services in 5 minutes intervals for a few consecutive days. After the data is collected, use Performance tools (licensed product number 5761-PT1) to create the following reports to use for sizing and Disk Magic modelling:

- ▶ System report, sections Disk utilization
- ▶ Resource report, section Disk utilization
- ▶ Component report, section Disk activity

These reports give size for the peaks in I/O per second (IOPS), writes per second, and MBps.

### 11.3.1 Sizing for natively connected and VIOS connected DS8000

The IBM i capability, *skip operations*, is the capability of efficiently reading or writing a non-contiguous block of data. An example of skip operations is a block of data that spans over 128 KB pages, and some of the 4 KB pages in the 128 KB pages are not populated with data. The skip operations function creates a bit mask that indicates which pages contain data and must be written. Only the pages that are populated with data are written to the storage system.

IBM i native attaches to the DS8000 supports skip operations. However, VIOS requires that they are repeated by breaking up the non-contiguous I/O block into multiple discrete contiguous I/O blocks. Consequently, you see more writes per second and smaller transfer sizes in the environment with VIOS, as compared to native attachment.

There is no difference in the overall workload throughput in MBps, whether the DS8000 is connected natively to IBM i or through VIOS. This point is valid for both VIOS NPIV and VIOS VSCSI. Therefore, our advice is to use the same sizing guidelines for both natively and VIOS connected DS8000.

**Sizing:** Although the characteristics of the IBM i workload slightly change when attaching the DS8000 through VIOS, use the same sizing guidelines that apply for connecting natively to IBM i.

### 11.3.2 Planning for arrays and DDMs

For IBM i production workloads, use 146 GB 15,000 revolutions per minute (RPM) disk drive modules (DDMs) or 300 GB 15,000 RPM DDMs. The larger, slower drives might be suitable for less I/O intensive work, or for those workloads that do not require critical response times, for example, archived data, or data that is high in volume, but low in use, such as scanned images.

Typical IBM i workloads will benefit from using solid state drives (SSD) with 300GB or 400 GB capacities. From a performance point of view, they can be the best option for heavy I/O applications.

**Tip:** SSD can only be used with RAID 5, and there are other restrictions regarding the use of SSDs. For more information about implementing SSD with IBM i, see *DS8000: Introducing Solid State Drives*, REDP-4522.

Consider using RAID 10 for an IBM i workload, because it helps provide significant high availability and performance improvements, compared to RAID 5 or RAID 6.

### 11.3.3 Cache

The DS8000 cache size is significant for an IBM i workload. Use the Disk Magic tool to model the disk response times and DS8000 utilizations for a particular IBM i workload with certain cache sizes, and determine the best size of DS8000 cache.

As a rough guideline for the cache size based on the DS8000 capacity, see “Table 6.88, Cache guidelines” in the Redbooks publication, *DS8800 Performance Monitoring and Tuning*, SG24-8013-00.



### 11.3.4 Number of ranks

When considering the number of ranks, take into account the maximum disk operations per second, including RAID penalty, and per rank. These values are measured at 100% DDM utilization with no cache benefit and with an average I/O of 4 KB. Larger transfer sizes reduce the number of operations per second.

The maximum disk operations per second depends on the type of disks being used in the DS8000 series, however they do not depend on the disk capacity, for example a rank of 300 GB 15 K RPM Fibre Channel DDMs supports the same maximal number of disk operations per second, as a rank of 146 GB 15 K RPM Fibre Channel DDMs.

The following measured maximum disk operations per second are for other DDM types:

- ▶ One 6+P RAID-5 rank of 15 K RPM Fibre Channel disks in DS8100, DS8300 and DS8700 can handle maximum 1800 disk operations per second for 100% DDM utilization.
- ▶ One 6+P RAID-5 rank of 15 K RPM SAS disks in DS8800 can support a maximum of 2047 disk operations per second for 100% DDM utilization.
- ▶ One 6+P RAID-5 rank of *large form factor* (LFF) SSD in DS8100, DS8300 and DS8700 is capable of supporting a maximum of 25 200 disk operations per second, at 100% SSD utilization.
- ▶ One 6+P RAID-5 rank of *small form factor* (SFF) SSD in DS8800 can support maximum of 43 000 disk operations per second, at 100% disk SSD utilization.

Based on these values, it is possible to calculate how many host IOPS each rank can handle at the preferred utilization of 40% for IOP-based adapters or 60% for IOP-less adapters. Examples of maximal host I/Os per hard disk drive (HDD) rank for the workloads with 70% reads per second, and 50% reads per second, as shown in Table 11-4. This calculation is done for an average of maximal supported disk operations in a rank with spares, and a rank without spares. It calculates 20% of read cache hit and 30% of write cache efficiency.

Table 11-4 Maximal host IOPS for HDD ranks

RAID rank type	Host IOPS at 70% reads		Host IOPS at 50% read	
	IOP-less	IOP-based	IOP-less	IOP-based
<b>Fibre Channel DDM</b>				
RAID-5 15 K RPM	827	551	643	428
RAID-10 15 K RPM	1102	706	982	655
RAID-6 15 K RPM	636	424	463	308
<b>SAS DDM</b>				
RAID-5 15 K RPM	940	627	731	487
RAID-10 15 K RPM	1253	835	1116	744
RAID-6 15 K RPM	723	482	526	351

Table 11-4 shows RAID 10 can support higher host I/O rates than RAID 5. However, you must consider this against the reduced effective capacity of a RAID 10 rank, as compared to RAID 5. RAID 6 is also performs slightly slower than RAID 5.

Table 11-5 shows the maximal number of host I/Os per rank of SSDs for 60% disk utilization. The calculation is done for the average of maximum IOPS to an 6+P and 7+P rank, taking into account 20% read cache hit and 30% write cache efficiency. Most of the implementations will be done with IOP-less adapters, so this calculation applies to 60% utilization of SSD valid for IOP-less connection.

Table 11-5 Maximal host IOPS for SSD ranks

RAID 5 rank type	Host IOPS at 70% reads	Host IOPS at 50% read
6+P LFF SSD	10,800	8,400
7+P LFF SSD	12,343	9,900
6+P SFF SSD	18,429	14,333
7+P SFF SSD	21,061	16,381

### 11.3.5 Sizing for SSD and hot data management with IBM i

The following sections provide sizing considerations.

#### Using the Storage Tier Advisory Tool

To size the number of Solid State Drives for an IBM i workload consider using Storage Tier Advisor Tool (STAT) for IBM i LUNs. STAT provides you the heat distribution of on the IBM i data. Figure 11-1 shows an example of heat distribution on IBM i.

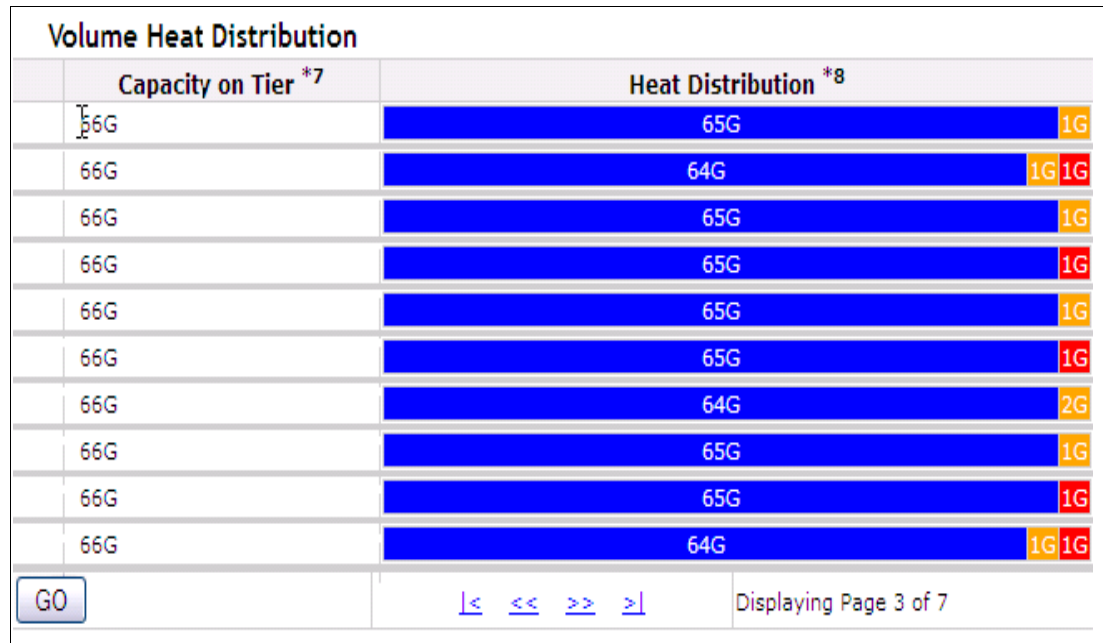


Figure 11-1 STAT output for an IBM i workload

For more information how to use STAT, see the Redpaper publication, *IBM System Storage DS8000 Easy Tier*, REDP-4667-02.

#### Using guidelines

For sizing SSDs at the installations who do not have Easy Tier installed and therefore cannot use STAT, you might want to use the following guidelines:

- ▶ Consider a rough guideline that one SSD replaces 5 \* 15 K RPM HDD as far as performance of IBM i workload are concerned.

- ▶ Consider the skew level shown in Figure 11-2, it presents the percentage of workload; (small size I/O) on the percentage of active data. It is a skew level of an IBM i benchmark workload that is a good emulation of a general IBM i customer's workload.

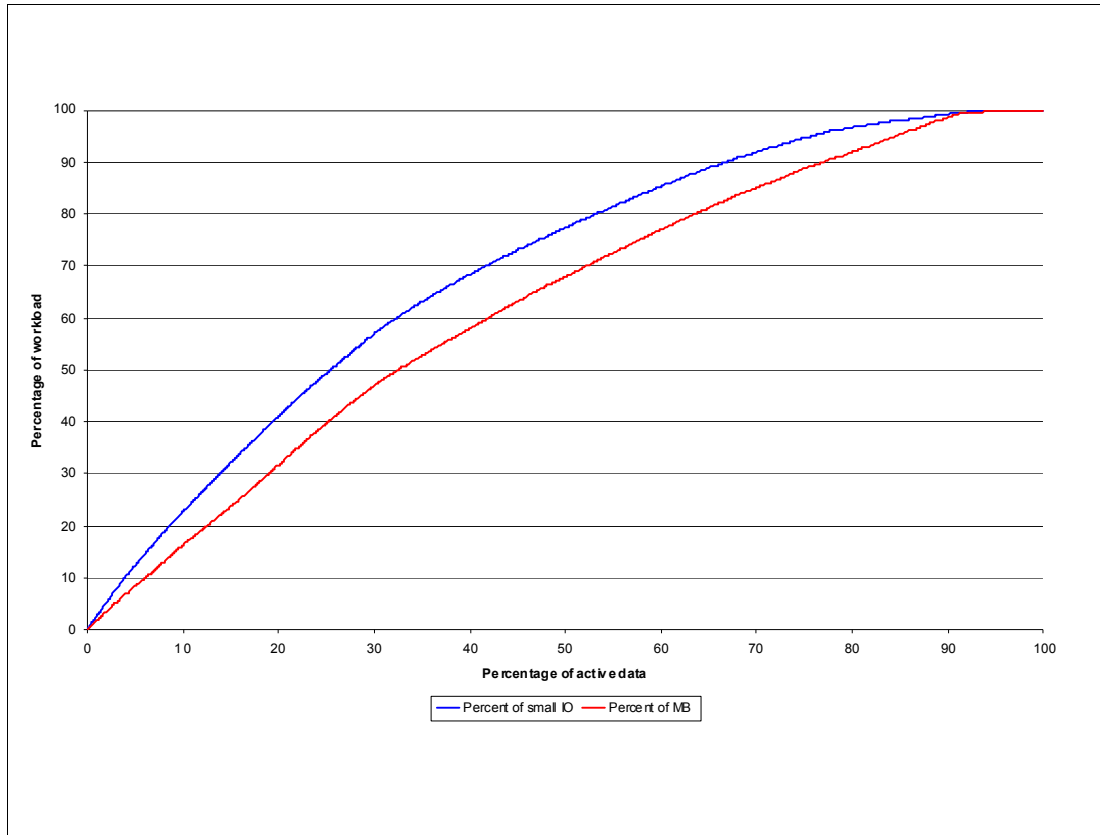


Figure 11-2 Skew level of an IBM i benchmark workload

**Example:** You are planning the configuration of 2 \* RAID-5 ranks of 300 GB SSD and 22 \* RAID-5 ranks of 146 GB 15 K RPM HDD for an IBM i workload.

The usable capacity on SSD ranks is about 3.4 TB and the capacity on the HDD ranks is about 19 TB, therefore, about 15% of the overall capacity will reside on SSD. Assuming that the active data covers all the capacity, and based on the skew level graph in Figure 11-2, about 33% of I/O with small transfer sizes will be done to SSD.

To model performance of the IBM i workload with the described DS8000 configuration and by using Easy Tier to relocate the data, you might want to use Disk Magic. Since the modelling of Easy Tier is presently not supported for IBM i, you can use the following workaround, assuming that the workload experiences smaller transfer sizes: Split the present peak I/O in the ratio 33% / 67% and model the 33% part on the configuration with 2 \* RAID-5 ranks of 300 GB SSD, and the 67% part on 22 \* RAID-5 ranks of 146 GB 15 K RPM HDD. To get an estimation of the response time in the peak, calculate the weighted average of response times of the two parts.

### Hot-spot management with Easy Tier

Using Easy Tier is an option to relocate hot data to SSD for an IBM i workload. For more information about using Easy Tier, see the Redpaper publication, *IBM System Storage DS8000 Easy Tier*, REDP-4667-02. Besides using Easy Tier, the IBM i based options for hot-spot management for an IBM i workload are described in the following section.

## Using IBM i tools and methods

Alternatively to Easy Tier and STAT tool, you can use the IBM i tools for hot-spot management. IBM i data relocation methods can be used with natively and VIOS\_NPIV connected DS8000, because IBM i recognizes the LUNs that reside on SSD in DS8000 by the LUN Vital Product Data (VPD).

The following IBM i tools and methods are available to you:

- ▶ *Performance Explorer (PEX)* is a data collection tool in IBM i which collects information about a specific system process or resource to provide detailed insight. PEX can be used to identify the IBM i objects that are the most suitable to be relocated to solid-state drives. For this, usually a PEX collection of IBM i disk events is run, such as synchronous and asynchronous reads, synchronous and asynchronous writes, page faults, and page outs.
- ▶ *iDoctor* is a suite of tools used to manage the collection, investigate, and analyze performance data on IBM i. The data collected by PEX can be analyzed by the PEX-Analyzer tool of iDoctor. An example of iDoctor showing read operations by IBM i objects, can be seen in Figure 11-3. You might notice that big majority of read operations goes to the three IBM i objects, so these objects are good candidates to relocate to SSD.

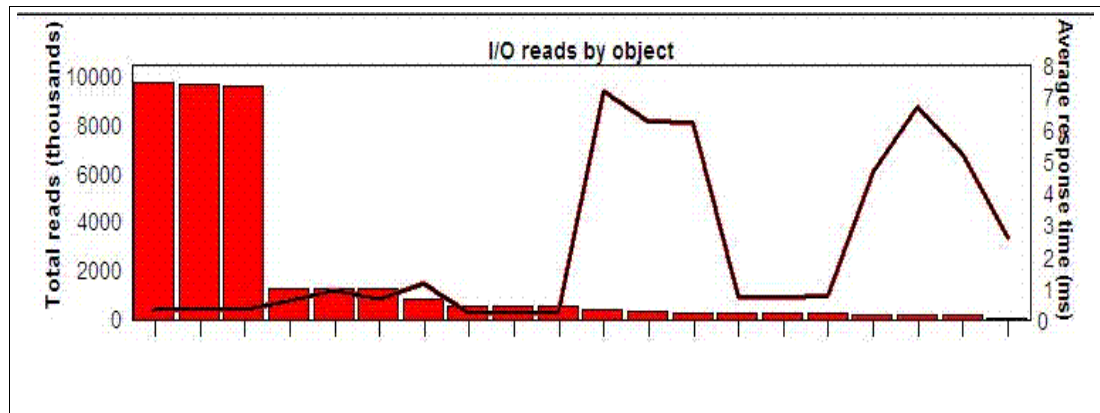


Figure 11-3 Read operations on IBM i objects by iDoctor

- ▶ *IBM i Media Preference Data Migration* is the method to migrate specific IBM i objects to SSD by using IBM i commands, such as Change Physical File (CHGPF) with parameter **UNIT(\*SSD)**.
- ▶ *ASP balancing* is the IBM i method based on the data movement within a system disk pool or base disk pool Auxiliary Storage Pool - ASP in IBM i. Two hot spot management methods are available with ASP balancing:
  - *Hierarchical Storage Management (HSM) balancing*. In this way of data relocation you first trace the ASP by command **TRCASPBAL** which collects the data statistics, then you use the command **STRASPBAL** with parameter **TYPE(\*HSM)** to actually relocate the hot data to high performing LUNs.
  - *Media Preference (MP) balancing*. This ASP balancer function helps correct any issues with media preference flagged database objects which are not on their preferred media type, which is either SSDs or HDDs. You start it with the command **STRASPBAL TYPE(\*MP)** with the parameter **SUBTYPE** set to **\*CALC**, **\*SSD** or **\*HDD**, depending which way of data migration that you want to use. MP balancing is supported starting with IBM i V7.1.

For more information about using IBM i based tools and methods, see the Redbooks publication, *DS8800 Performance Monitoring and Tuning*, SG24-8013-00, and the “IBM i Information Center” as listed in 11.1.4, “Useful websites” on page 204.

### 11.3.6 Number of Fibre Channel adapters in IBM i and in VIOS

When deciding on the number of Fibre Channel adapters in an IBM i and DS8000 configuration, in addition to the maximal number of LUNs per port specified in Table 11-3 on page 207, also consider the performance aspect of the configuration. To achieve good performance of DS8000 with IBM i, in some cases, you will need to implement less than the maximum number of LUNs per port. Here, the most important factor to consider is the adapter throughput capacity. This section provides the measured maximal I/O rate (in IOPS) and maximal data rate (MB per second or MBps) per an IOP-based adapter, or per port in an IOP-less adapter. Based on these values, information is provided about the sizing guidelines for the capacity (number of LUNs) per port or per adapter, to achieve good performance.

Also provided are guidelines for sizing Fibre Channel adapters in IBM i for native attachment to the DS8000, and for sizing Fibre Channel adapters in VIOS for VIOS VSCSI or VIOS NPIV attachment.

#### Fibre Channel adapters in native attachment

For the measured values of maximal I/O rates and data rates per port in other adapters, see the following rates:

- ▶ A 4 GB IOP-based Fibre Channel adapter, feature number #5760, connected to the IOP, feature number #2844, can support these maximums:
  - Maximum 3,900 IOPS
  - Maximum 140 MBps of sequential throughput
  - Maximum 45 - 54 MBps of transaction throughput
- ▶ A port in 4 GB IOP-less Fibre Channel adapter, feature number #5749, or #5774, can support these maximums:
  - Maximum 15,000 IOPS
  - Maximum 310 MBps of sequential throughput
  - Maximum 250 MBps of transaction throughput
- ▶ A port in 8 GB IOP-less adapter, feature number #5735, can support these maximums:
  - Maximum 16,000 IOPS
  - Maximum 400 MBps of sequential throughput
  - Maximum 260 MBps of transaction throughput

When sizing the number of adapters for an IBM i workload, consider the guidelines by IOPS, per port, by disk capacity per ports, and by MBps, per ports, which are shown in Table 11-6. The guidelines by IOPS and by MBps are to be used for the peak of a workload, because they consider 70% utilization of adapters per ports.

The guideline for the disk capacity per port is based on the number of IOPS that one port can sustain at 70% utilization, on the *access density* (AD) of an IBM i workload, and on the preferred maximum 40% utilization of a LUN.

**Tip:** Access density, measured in IOPS/GB, is the ratio that results from dividing the average IOs/sec by the occupied disk space. These values can be obtained from the IBM i system, component, and resource interval performance reports. Based on experiences with IBM i workloads, it is assumed that AD = 1.5 IOPS/GB for the sizing guideline.

The guideline for MBps per port is based on the measurements of maximum sequential and transactional throughput of a port. Based on DS8000 reports from typical workloads, a rough estimation is made that about 20% of the IBM i I/Os are sequential. Accordingly, the calculation is made for the weighted average between transaction MBps and sequential MBps, and applied 70% for port utilization.

When planning for multipath, consider that two or more ports are used for I/O. Therefore, you can implement two times or more times the preferred capacity per port.

Table 11-6 Sizing for Fibre Channel adapters in IBM i

IOP	Adapter / port	IOPS at 70% utilization	Disk capacity per port (GB)	MBps per port
2844	5760 4 Gb IOP-based Fibre Channel adapter	3,200	853	45 MBps
-	Port is 5749/5774 4 Gb IOP-less Fibre Channel adapters	10,500	2,800	183 MBps
-	Port in 5735 8 Gb IOP-less Fibre Channel adapter	12,250	3,267	202 MBps

**Example:** You plan to implement DS8000 connected to IBM i with 8 Gb adapters feature number 5735. You plan for the LUNs of capacity 141 GB. By the performance guideline in Table 11-6, you consider 3267 GB per port, so you might want to connect  $3267 / 141 = 23 *$  LUNs per port, or  $46 * \text{LUNs per 2 ports}$  in 2 different adapters for multipath.

### Fibre Channel adapters in VIOS

When connecting the DS8000 series to IBM i with VIOS NPIV or VIOS VSCSI, follow the guidelines in Table 11-7 to determine the necessary number of physical Fibre Channel adapters in VIOS. The guidelines are obtained by modeling port utilization with Disk Magic for the DS8700 connected to an open server with 70% reads and 50% reads, and with 16 KB transfer sizes and 120 KB transfer sizes. The maximum amount of IOPS to keep port utilization under 60% is modeled.

Table 11-7 shows the maximum amount of IOPS and MBps at separate read and write ratios, and separate transfer sizes to keep port utilization in VIOS under the preferred 60%.

Table 11-7 Sizing the Fibre Channel adapters in VIOS

Percentage of reads per second	Transfer size (blocksize)	Maximum IOPS, per one port in VIOS		Maximum MBps, per one port in VIOS	
		8 GB adapter	4 GB adapter	8 GB adapter	4 GB adapter
70%	16		18,500		289
70%	120		2,900		340
50%	16		26,000		406
50%	120		3,900		457

**Port in an IOP-less adapter:** Avoid using one port in an IOP-less adapter to connect disk storage and the other port to connect to tape.

## 11.4 Sizing and numbering of LUNs

This section provides information about the sizes of DS8000 LUNs for IBM i, which sizes of LUNs to use, and how many LUNs to assign to a port, to help achieve good performance.

### 11.4.1 Logical volume sizes

IBM i connected natively or with VIOS NPIV is supported on DS8000 as fixed block (FB) storage. Unlike other open systems using the FB architecture, IBM i only supports specific volume sizes, and these volume sizes might not be an exact number of extents. In general, these volume sizes relate to the volume sizes available with internal devices, although certain larger sizes are supported for external storage only. IBM i volumes are defined in decimal gigabytes, for example, 10<sup>9</sup> bytes.

When creating the logical volumes for use with IBM i, you can see in almost every case that the IBM i device size does not match a whole number of extents, and therefore, certain space is wasted. Table 11-8 provides the IBM i volumes sizes and the number of extents required for IBM i volume sizes.

Table 11-8 IBM i logical volume sizes

Model type		IBM i device size (GB)	Number of logical block addresses (LBAs)	Extents	Unusable space (GiB <sup>1</sup> )	Usable space%
Unprotected	Protected					
2107-A81	2107-A01	8.5	16,777,216	8	0.00	100.00
2107-A82	2107-A02	17.5	34,275,328	17	0.66	96.14
2107-A85	2107-A05	35.1	68,681,728	33	0.25	99.24
2107-A84	2107-A04	70.5	137,822,208	66	0.28	99.57
2107-A86	2107-A06	141.1	275,644,416	132	0.56	99.57
2107-A87	2107-A07	282.2	551,288,832	263	0.13	99.95

1. GiB represents “binary gigabytes” (2<sup>30</sup> bytes), and GB represents “decimal gigabytes” (10<sup>9</sup> bytes).

**Tip:** IBM i levels 5.4 and 6.1 do not support logical volumes of size 8.59 and 282.2 as an IBM i load source unit or boot disk, where the load source unit is located in the external storage server. IBM i 7.1 does not support logical volumes of size 8.59 as a load source unit.

### LUN sizes

IBM i can only use fixed logical volume sizes, therefore, configure more logical volumes than actual DDMs. At a minimum, use a 2:1 ratio. For example, with 146 GB DDMs, use a maximum size of 70.56 GB LUNs.

When connecting DS8000 natively to IBM i, it is important to consider using a smaller size of LUNs with IOP-based adapters, because they do not support SCSI command tag queuing. Using smaller LUNs can reduce I/O queues and wait times by allowing IBM i to support more parallel I/Os. In general, 35 GB LUNs can be a good number to define with IOP-based adapters. Unlike IOP-based adapters, IOP-less adapters support SCSI command tag queuing. Therefore, you can have bigger LUNs defined, such as 70.56 GB LUNs or 141.1 GB LUNs.

Connection of DS8000 series to IBM i with VIOS NPIV or VIOS VSCSI supports SCSI command tag queuing, therefore, use 70.56 GB LUNs or 141.1 GB LUNs with this type of connection.

### Number of LUNs per port

IOP-based adapters can support up to 32 LUNs, but for performance reasons, do not use the maximum number. Instead, define less than 32 LUNs per adapter. Table 11-6 on page 214 can help you to determine the number of LUNs per adapter, because it shows the preferred capacity per adapter. With multipath, consider using double capacity per two adapters.

With *native attachment* using IOP-less adapters, consider the capacities per adapter that are shown in Table 11-6 on page 214, and calculate the number of LUNs from them.

Table 11-9 gives you an overview of the preferred number of LUNs per port for different adapters and different sizes of LUNs. The calculation for the table assumes workload Access Density = 1.5.

Table 11-9 Preferred number of LUNs per port

Preferred number of LUNs per port	IOP 2844 / IOA 5760	4 Gb adapters 5749 / 5774	8 Gb adapter 5735
35 GB LUNs	24	64	64
70 GB LUNs	Size not advisable	39	46
141 GB LUNs	Size not advisable	20	23

For two ports in multipath, use the 2-times preferred number. If it is bigger than the maximal number of LUNs per port, use the maximum number per port.

When connecting DS8000 series to IBM i with **VIOS NPIV**, you can assign 64 LUNs per port in virtual Fibre Channel adapters, or 64 LUNs per multiple ports in multipath, without a performance impact.

When connecting DS8000 series to IBM i with **VIOS VSCSI**, you can define up to 16 LUNs per virtual port, or up to 16 LUNs per multiple ports in multipath, without a performance impact.

**Tip:** IBM i multipath with VIOS NPIV or VIOS VSCSI is a multipath with two or more VIOS with each path using a separate VIOS.

## 11.4.2 Sharing or dedicating ranks for an IBM i workload

Consider using separate extent pools in DS8000 series for IBM i workload and other workloads. This method isolates the I/O for each server and helps provide better control of performance of IBM i workload.

However, you might consider sharing ranks when the other servers' workloads have a sustained low disk I/O rate, compared to the System i I/O rate. Generally, System i has a relatively high I/O rate, where that of other servers might be lower, often under one I/O per GBps.

As an example, a Windows file server with a large data capacity can typically have a low I/O rate with fewer peaks and can be shared with IBM i ranks. However, Microsoft SQL Server, IBM DB2 Server, or other application servers, might show higher rates with peaks, therefore, consider using other ranks for these servers.



When sharing DS8000 among many IBM i workloads, dedicate the ranks to the most important IBM i systems to help provide stable performance. Also consider sharing the ranks among less important or smaller IBM i workloads. However, the decision to mix platforms or mix IBM i workloads on a DS8000 array, rank, or extent pool is based on your IBM i performance requirements.

### 11.4.3 Connecting using SAN switches

When connecting DS8000 series to IBM i using switches, consider the following points for your SAN configuration:

- ▶ Zone the SAN switches so that one zone contains one or multiple IBM i ports, and one DS8000 port. The reason for this advice is that IBM i establishes only one path from an IBM i port to a DS8000 port; even if IBM i ports is in zone with multiple DS8000 ports IBM i will still use only one of them for I/O traffic.

If multiple IBM i ports are in the same zone with multiple DS8000 ports, each IBM i port might use the same DS8000 port for the IO traffic. Thus, one of the DS8000 ports becomes overloaded and the others are idle, which causes unbalanced usage of DS8000 ports and consequently impact on DS8000 performance.

DS8000 host adapters can be shared between System i and other platforms.

- ▶ For performance reasons, it is best to limit the number of IBM i adapters or ports connected to one Host attachment (HA) card in DS8000. Based on the available measurements and experiences, consider planning the following maximal number of IBM i ports for one port of HA card in DS8000:
  - With IOP-based adapters:
    - Four adapters per HA card port in DS8100, DS8300, and DS8700
    - Up to 12 adapters per HA card in DS8800
  - With 4 GB IOP-less adapters:
    - Two to four ports per HA card in DS8100, DS8300 and DS8700
    - Six to ten ports per HA card in DS8800
  - With 8 GB IOP-less adapters:
    - Two ports per HA card in DS8100, DS8300 and DS8700
    - Four to eight ports per HA card in DS8800

For a current list of switches supported under IBM i, see the web pages “System Storage Interoperation Center” and “IBM i Storage Solutions” as listed in 11.1.4, “Useful websites” on page 204.

## 11.5 Using multipath

With IBM i, multipath is part of the base operating system. You can define up to eight connections from multiple I/O adapters on an IBM i server to a single logical volume in the DS8000. Each connection for a multipath disk unit functions independently. Several connections provide availability by allowing disk storage to be used even if a single path fails.

If you have a configuration where the logical units are only assigned to one I/O adapter, you can easily change to multipath by assigning the logical units in the DS8700 to another I/O adapter. The existing DDxxx drives change to DMPxxx and new DMPxxx resources are created for the new path. It might be possible in your environment to re-assign logical volumes to other I/O adapters, but careful planning and implementation are required.

Multipath is important for IBM i, because it provides greater resilience to SAN failures, which can be critical for IBM i, due to the single level storage architecture. Multipath is not available for IBM i internal disk units, but the possibility of path failure is much less with internal drives. It is because there are fewer interference points where problems can occur, such as long fiber cables and SAN switches, increased possibility of human error when configuring switches and external storage, and the concurrent maintenance on the DS8000, which can make certain paths temporarily unavailable.

Many IBM i clients still have their entire environment on the server auxiliary storage pool (ASP) and loss of access to any disk will cause the system to fail. Even with user ASPs, loss of a user ASP disk eventually causes the system to stop. Independent auxiliary storage pools (IASPs) provide isolation, so that loss of disks in the IASP only affects users that are accessing that IASP, but the rest of the system is unaffected. However, with multipath, even loss of a path to a disk in an IASP does not cause an outage.

With the combination of multipath and RAID 5, RAID 6, or RAID 10 protection in the DS8000, you can take advantage of full protection of the data paths and the data itself without the requirement for additional disks.

Besides providing resiliency, multipath improves performance because IBM i uses all available paths to a LUN in DS8000. The I/O operations are balanced among the available paths in a round robin algorithm.

### **11.5.1 Avoiding single points of failure**

When implementing multipath, provide as much redundancy as possible. With native attachment to the DS8000, multipath requires two Fibre Channel IOAs, at a minimum, when connecting the same logical volumes. With IOP-less adapters, connect the LUNs to two ports from separate adapters. Ideally, place them on separate buses or loops and in separate I/O racks in the Power System. If a SAN is included, use separate switches for each path. Also use host adapters in separate I/O drawer pairs in the DS8000.

When connecting DS8000 to IBM i with VIOS, implement multipath so that each path to logical volumes uses a separate VIOS, separate SAN switches, and host adapters, in separate I/O drawer pairs in the DS8000. With such implementation, IBM i will keep resiliency, even if a VIOS fails.

### **11.5.2 Configuring the multipath**

To achieve multipathing, when DS8000 attaches natively to IBM i, assign the volumes to two or more ports from Fibre Channel adapters or IBM i host adapters.

When connecting with VIOS NPIV, attach DS8000 to an IBM i LPAR with two VIOS in NPIV, and assign the LUNs to the virtual Fibre Channel adapter that connect to both of the VIO servers, to provide availability in case of maintenance or failure of one VIO servers. You can also connect IBM i with more than two VIOS and setup one path to the LUNs through each of them, to provide even better resiliency. Up to eight paths can be established to a LUN for IBM i.

In certain installations, each VIOS connects to a separate SAN, such as DS8000 attaching to both SANs. In this case, consider connecting two virtual Fibre Channel adapters to each VIOS. Each virtual Fibre Channel adapter is assigned to a separate physical adapter and communicates to a separate SAN. This way, you help ensure resiliency in SANs during outage of one VIOS. In the DS8000, assign the LUNs to all four virtual Fibre Channel adapters in IBM i.

For example, a client is performing maintenance on one VIOS. If the other, active VIOS is connected only with one SAN and a switch in that SAN fails, IBM i misses the LUNs. However, if each VIOS is connected to both SANs, resiliency occurs.

Figure 11-4 shows an example of connecting two VIOS through two SANs.

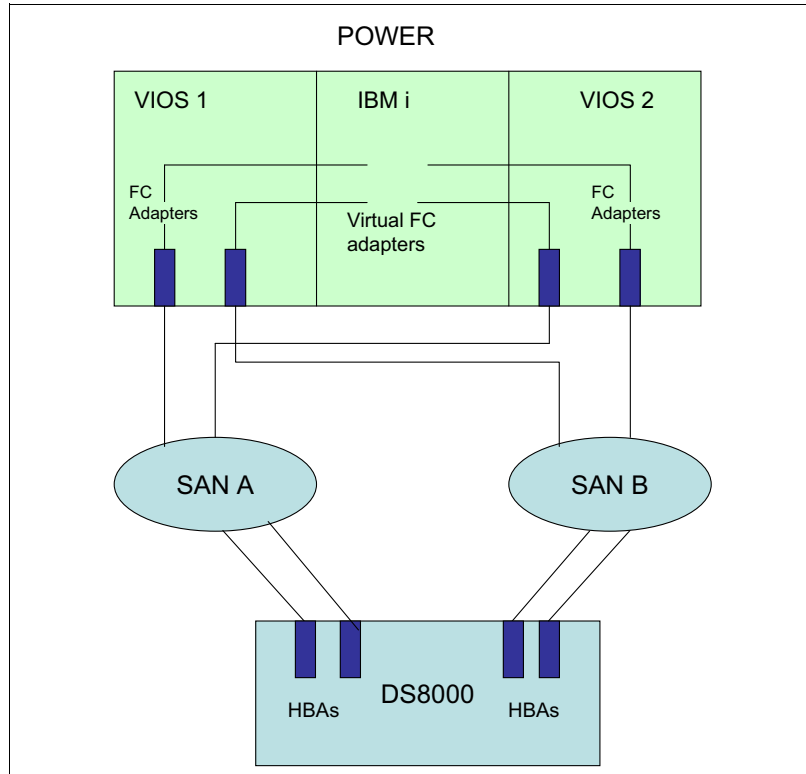


Figure 11-4 Connecting with two VIOS and two SANs

VIOS VSCSI assigns the volumes to two or more Fibre Channel adapters, each of them in a separate VIOS. The volumes in VIOS or hdisks can then be mapped to the separate virtual SCSI adapters in IBM i. As specified in 11.6.7, “Setting up VIOS” on page 223, change the SCSI reservation attribute for the LUNs in each VIOS to non-reserve.

After the LUNs are assigned to IBM i, IBM i recognizes them as multipathed LUNs and establishes two or more paths to them. To achieve multipathing, you do not need to set up any driver in IBM i or perform any further tasks.

**Native attachment multipathing:** With native attachment to the DS8000, multipathing can be established for the Load Source LUN (Boot LUN) when connecting with IOP-based or IOP-less adapters on IBM i levels V6R1 or V7R1. However, multipathing is not possible for the Load Source LUN on IBM i V5R4.

### 11.5.3 Multipathing rules for multiple System i hosts or partitions

When you use multipath disk units, consider the implications of moving IOPs and multipath connections between nodes. Do not split multipath connections between nodes, either by moving IOPs between logical partitions (LPAR), or by switching expansion units between systems. If two separate nodes have both connections to the same LUN in the DS8000, both nodes might overwrite data from the other node.

The multiple system environment enforces the following rules when using multipath disk units:

- ▶ If you move an IOP with a multipath connection to a separate LPAR, you must also move all other IOPs with connections to the same disk unit to the same LPAR.
- ▶ When you make an expansion unit a switchable entity, make sure that all multipath connections to a disk unit switch with the expansion unit.
- ▶ When you configure a switchable entity for a independent disk pool, make sure that all of the required IOPs for multipath disk units switch with the independent disk pool.

If a multipath configuration rule is violated, the system issues warnings or errors to alert you about the condition. It is important to pay attention when disk unit connections are reported missing. You want to prevent a situation where a node might overwrite data on a LUN that belongs to another node.

Disk unit connections might be missing for a variety of reasons, but especially if one of the preceding rules was violated. If a connection for a multipath disk unit in any disk pool is found to be missing during an initial program load or vary on, a message is sent to the System Operator message queue (QSYSOPR).

If a connection is missing and you confirm that the connection was removed, you can update hardware service manager (HSM) to remove that resource. The HSM tool can be used to display and work with system hardware from both a logical and a packaging viewpoint. This tool also aids in debugging I/O processors and devices, and fixing failing or missing hardware. You can access HSM in SST and DST by selecting the option to start a service tool.

## 11.6 Configuration guidelines

This section contains information about configuring DS8000 for IBM i, connecting through a SAN, connecting through a VIOS, and for defining the DS8000 volumes to IBM i.

**IBM i specific:** This section does not provide step-by-step procedures on how to set up the DS8000 for IBM i, just the tasks that are specific or different when connecting IBM i.

### 11.6.1 Creating extent pools for IBM i LUNs

Create two large extent pools for IBM i LUNs, each of them assigned to one DS8000 server with the rank group of 0 for workload balancing. When creating the LUNs, consider defining them in rotate extents mode, also known as storage pool striping.

### 11.6.2 Defining the LUNs for IBM i

For DS8000 natively attached to IBM i, or attached through VIOS NPIV, select the following GUI options when defining LUNs for IBM i:

- ▶ Volume type:
  - iSeries protected or iSeries unprotected

For more information about protected and unprotected LUNs, see 11.6.3, “Protected versus unprotected volumes” on page 221.
- ▶ Select the standard allocation method and rotate extents.

- ▶ Select the volume sizes for IBM i:  
For more information about IBM i volume sizes, see 11.4, “Sizing and numbering of LUNs” on page 215.

Figure 11-5 shows an example of creating an IBM i LUN.

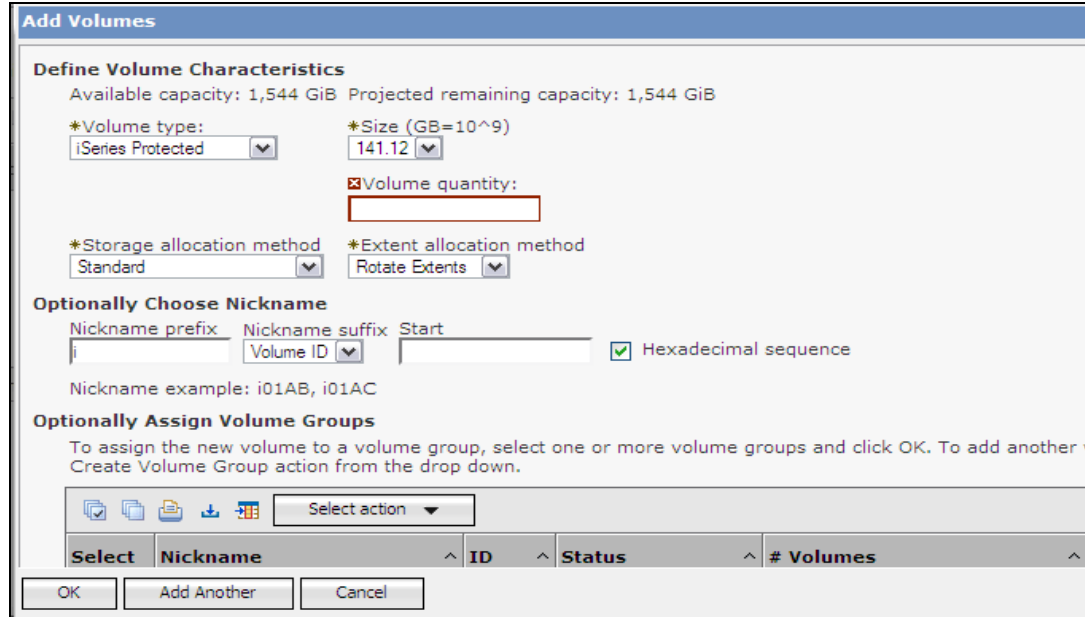


Figure 11-5 Defining an IBM i LUN

If you use the `crtfbvo1` DS CLI command to define IBM i LUNs, specify the parameter `-OS400` and the model of LUN that corresponds to a particular LUN size and protection (Table 11-8 on page 215). For more information about protected and unprotected volumes, see 11.6.3, “Protected versus unprotected volumes” on page 221.

On DS8000 models prior to DS8800, specify parameter `-eam rotateext` to achieve storage pool striping for IBM i LUNs. For the DS8800, rotate extents is the default extent allocation method. For example, you can use a DS CLI command to create an IBM i volume on a DS8800:

```
mkfbvo1 -extpool p6 -os400 A04 -name ITS0_i_#h 2000-2003
```

This command defines a protected volume of size 70.56 GB with default rotate extents method. If the DS8000 is to be attached to IBM i with VIOS VSCSI, create LUNs as for open systems. Such LUNs can be of any size and you do not need to define them in one of the fixed sizes for IBM i. However, consider keeping the size of volumes within the advisable limits, as is explained in 11.4, “Sizing and numbering of LUNs” on page 215.

### 11.6.3 Protected versus unprotected volumes

When defining IBM i logical volumes, you must decide whether these are to be protected or unprotected. It is simply a notification to IBM i and does not mean that the volume is protected or unprotected. In fact, all DS8700 LUNs are protected by either RAID 5, RAID 6, or RAID 10.

Defining a volume as unprotected means that is available for IBM i to perform a mirroring of that volume to another of equal capacity, either internal or external. If you do not intend to use IBM i host-based mirroring, define your logical volumes as protected.

## 11.6.4 Changing LUN protection

Although it is possible to change a volume from protected to unprotected, or unprotected to protected, use care when using the DS CLI to accomplish this task. If the volume is not assigned to any IBM i or is non-configured, you can change the protection.

However, if volume is configured, do not change the protection. If you do, you must first delete the logical volume, which returns the extents used for that volume to the extent pool. You can then create a new logical volume with the correct protection after a short period of time, depending on the number of extents returned to the extent pool.

Before deleting the logical volume on the DS8000, you must first remove it from the IBM i configuration, assuming it is configured. This removal is an IBM i task that is disruptive, if the disk is in the system ASP or user ASPs 2-32, because it requires an IPL of IBM i to completely remove the volume from the IBM i configuration. This task is the same as removing an internal disk from an IBM i configuration.

Deleting a logical volume on the DS8000 is similar to physically removing a disk drive from an System i. You can remove disks from an IASP with the IASP varied off without performing an IPL on the system.

## 11.6.5 Setting the ports and defining host connections for IBM i

When defining a host connection for a DS8000 connection to IBM i natively or with VIOS NPIV, specify the hosttype `iSeries` in the DS8000 GUI or DS CLI command. When connecting DS8000 natively, specify the WWPN of the port in IBM i. When connecting with VIOS NPIV, specify the WWPN of the port in virtual Fibre Channel adapter in IBM i.

When connecting DS8000 natively with IOP-less or IOP-based adapters, set up the ports in DS8000 as follows:

- ▶ With point to point connection, define the ports as `Fibre Channel_AL`. An exception is to use SCSI Fibre Channel protocol (SCSI-FCP) for ports that connect the load source LUN with IOP-based adapters through point to point connection.
- ▶ If the connection is through switches, define the ports as `SCSI-FCP`.

Figure 11-6 shows how to define the host connection for IBM i attached with SAN switches.

**Create New Host Connection**

**Define Host Ports**  
Define one or more host ports that you will use to map hosts to a volume group in the next step. After you add table will be mapped to the same volume group when you create the connection.

\*Host Connection Nickname: Host 1

\*Port Type: Fibre Channel Point-to-Point/Switched (FcSf)

\*Host Type: IBM pSeries, RS/6000 and RS/6000 SP Servers (AIX)(pSeries)

Enter the 16-digit WWPN manually, or select it from the list, and then add it to the table.

\*Host WWPN: 1000000c9831fa5 [Add]

Select	WWPN	Nickname
--------	------	----------

Figure 11-6 Create a host connection for IBM i in DS8000 GUI

The following example shows how to define a host connection for IBM i, using the DSCLI:

```
mkhostconnet -wwname 1000000c9831fa5 -hosttype iseries -volgrp v1 i_port1
```

## 11.6.6 Zoning the switches

When zoning the switches that connect the DS8000 natively to IBM i, consider the following guidelines:

- ▶ Zone the switches so that each port in IBM i Fibre Channel adapters accesses one port in DS8000. Multiple ports from IBM i adapters can access one port in DS8000. You can share a DS8000 with I/O traffic from multiple IBM i systems or other servers, providing that the aggregated I/O traffic from attached systems do not overload the port.
- ▶ Do not zone the switches so that multiple IBM i ports can access multiple DS8000 ports, because IBM i establishes only one path from one port to the disk. If you do, all IBM i ports establish the path through the same DS8000 port, but leave other DS8000 ports unused, causing performance bottlenecks on the used port.

## 11.6.7 Setting up VIOS

As opposed to connecting with VIOS VSCSI, with VIOS NPIV, you do not need to map the LUNs in VIOS to the ports in virtual Fibre Channel adapters. Instead, you assign the LUNs to virtual Fibre Channel ports when defining host connections in the DS8000.

## VIOS VSCSI

To implement multipath with two VIOS, complete the following steps:

1. Remove the SCSI reservation attribute from the LUNs or hdisks that are to be connected through two VIOS. To do this, enter the following command for each hdisk that will connect to the IBM i in multipath:

```
chdev -dev hdiskX -attr reserve_policy=no_reserve
```

2. Set the attributes of Fibre Channel adapters in the VIOS to `fc_err_recov=fast_fail` and `dyntrk=yes`. When the attributes are set to these values, the error handling in Fibre Channel adapter enables faster transfer to the alternate paths, in case of problems with one Fibre Channel path. To help make multipath within one VIOS work more efficient, specify these values by entering the following command:

```
chdev -dev fscsix -attr fc_err_recov=fast_fail dyntrk=yes-perm
```

3. Enter the following command for each hdisk (hdiskX) to get more bandwidth by using multiple paths:

```
chdev -dev hdiskX -perm -attr algorithm=round_robin
```

Ensure that the queue depth in VIOS and the queue depth in the VIOS matches the IBM i queue depth of 32 by using the following commands:

1. Enter the following VIOS command to check the queue depth on physical disks:

```
lsdev -dev hdiskxx -attr queue_depth
```

2. Set the queue depth to 32, if needed, by using the following command:

```
chdev -dev hdiskxx -attr queue_depth=32
```

## VIOS NPIV

Set the attribute of Fibre Channel adapters in VIOS to `fc-err-recov=fast` and `dyntrk=yes-perm`, as is explained in “VIOS VSCSI” on page 224.

### 11.6.8 Adding volumes to the System i configuration

After the logical volumes are created and assigned to the host, they appear as *non-configured units* to IBM i. When assigning a large quantity of LUNs, it can take time for all of them to appear in IBM i. At this stage, they are used in exactly the same method as for non-configured internal units. There is nothing particular to external logical volumes as far as IBM i is concerned.

You use the same functions for adding the logical units to an ASP, as you use for internal disks.

When connecting DS8000 natively or with VIOS NPIV, the volumes are shown in IBM i as disk units with the following characteristics:

- ▶ Type 2107.

- ▶ Model A0x or A8x:

The model depends on the size and protection.

- ▶ Serial number:

The serial number is in the form 50-yyyzzz. Where yyy stands for the LUN ID, and zzz stands for the last three characters of the DS8000 WWPN.

- ▶ Resource name:

The resource name starts with DMP when using multipath and with DD when using single path.



When connecting DS8000 to IBM i with VIOS VSCSI, the volumes appear in IBM i with the following attributes:

- ▶ Type 6B22
- ▶ Model 050
- ▶ Serial number with 12 characters

You can add volumes to IBM i configuration by using the IBM i 5250 interface, Operations Navigator GUI, or the web GUI, Systems Director Navigator for IBM i. These methods can be used for adding LUNs to IBM i System ASP, user ASPs, or IASPs.

**Tip:** The web GUI Systems Director Navigator for i is available with IBM i V6R1 and later.

### 11.6.9 Using the 5250 interface

You can add disk units to the configuration either by using the text (5250 terminal mode) interface with dedicated service tools (DST) or system service tools (SST), or with the System i Navigator GUI.

To add a logical volume in the DS8000 to the System ASP by using green screen SST, complete the following steps:

1. Run the **STRSST** command and sign on to start SST.
2. Select option 3, Work with disk units (Figure 11-7).

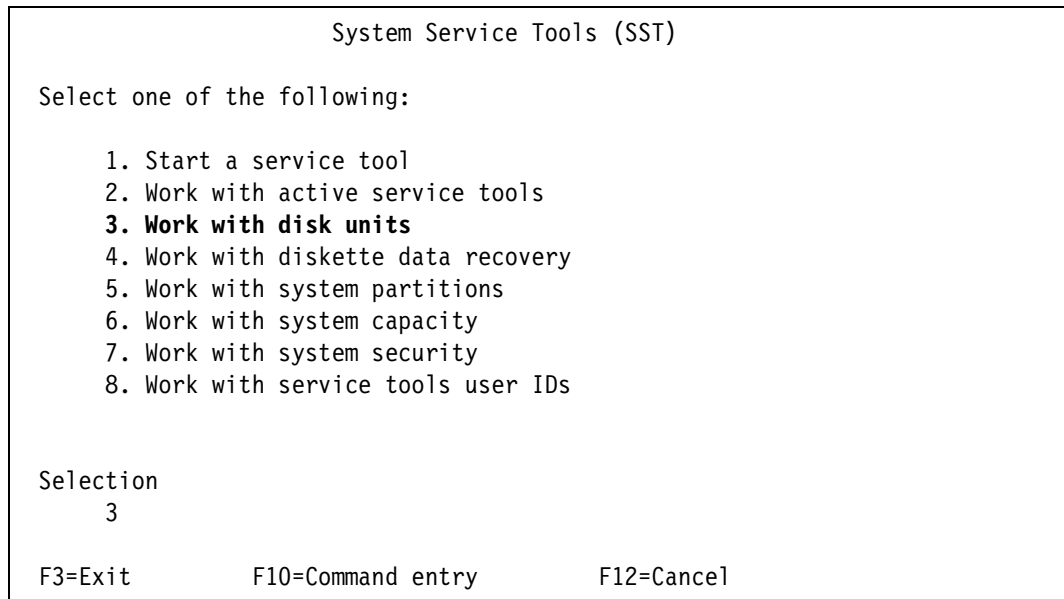


Figure 11-7 System Service Tools menu

3. Select option 2, Work with disk configuration (Figure 11-8).

```
Work with Disk Units

Select one of the following:

    1. Display disk configuration
    2. Work with disk configuration
    3. Work with disk unit recovery

Selection
    2

F3=Exit      F12=Cancel
```

Figure 11-8 Working with Disk Units menu

When adding disk units to a configuration, you can add them as empty units by selecting option 2, or you can select option 4 to allow IBM i to balance the data across all the disk units. Balancing the data offers a more efficient use of the available resources.

4. Select option 4, Add units to ASPs and balance data (Figure 11-9).

```
Work with Disk Configuration

Select one of the following:

    1. Display disk configuration
    2. Add units to ASPs
    3. Work with ASP threshold
    4. Add units to ASPs and balance data
    5. Enable remote load source mirroring
    6. Disable remote load source mirroring
    7. Start compression on non-configured units
    8. Work with device parity protection
    9. Start hot spare
    10. Stop hot spare
    11. Work with encryption
    12. Work with removing units from configuration

Selection
    4

F3=Exit      F12=Cancel
```

Figure 11-9 Working with Disk Configuration menu

- In the Add Units to ASPs panel (Figure 11-10), select option 1 or 2 to create a new ASP, or select option 3 to add LUNs to an existing ASP. In this example, use option 3 to add the LUNs to the existing ASP1.

```

Add Units to ASPs

Select one of the following:

    1. Create unencrypted ASPs
    2. Create encrypted ASPs
    3. Add units to existing ASPs

Selection
    3

F3=Exit      F12=Cancel
  
```

Figure 11-10 Creating a new ASP or using existing ASP

- In the Specify ASPs to Add Units panel (Figure 11-11), specify the ASP number to the left of the desired units and press Enter. In this example, ASP1 was specified for the system ASP.

```

Specify ASPs to Add Unit

Specify the existing ASP to add each unit to.

Specify  Serial                                     Resource
ASP      Number          Type Model Capacity Name
  1      50-210629F      2107 A04   70564 DMP015
  1      50-210529F      2107 A04   70564 DMP014
  1      50-210429F      2107 A04   70564 DMP013
  1      50-210329F      2107 A04   70564 DMP012

F3=Exit      F5=Refresh      F11=Display disk configuration capacity
F12=Cancel
  
```

Figure 11-11 Specifying the ASPs to add units

- In the Confirm Add Units panel (Figure 11-12), press Enter to continue if everything is correct. Depending on the number of units you are adding, this step can take time. When it completes, open your disk configuration to verify the capacity and data protection.

Confirm Add Units

Add will take several minutes for each unit. The system will have the displayed protection after the unit(s) are added.

**Press Enter to confirm your choice for Add units.**  
 Press F9=Capacity Information to display the resulting capacity.  
 Press F10=Confirm Add and Balance data on units.  
 Press F12=Cancel to return and change your choice.

ASP Unit	Serial Number	Type	Model	Resource Name	Protection	HotSpare protection
1	50-210729F	2107	A04	ccDMP016	RAID 5	N
2	50-200029F	2107	A04	ccDMP001	RAID 5	N
3	50-200129F	2107	A04	ccDMP002	RAID 5	N
4	50-200229F	2107	A04	ccDMP003	RAID 5	N
5	50-200329F	2107	A04	ccDMP004	RAID 5	N
6	50-200429F	2107	A04	ccDMP005	RAID 5	N

F9=Resulting Capacity                      F10=Add and Balance  
 F11=Display Encryption Status          F12=Cancel

Figure 11-12 Confirming add units

### 11.6.10 Adding volumes to an independent auxiliary storage pool

IASPs can be switchable or private entities. To add LUNs or create private (non-switchable) IASPs, using the Operations Navigator GUI, complete the following steps:

- Start System i Navigator. Figure 11-13 shows the initial window.

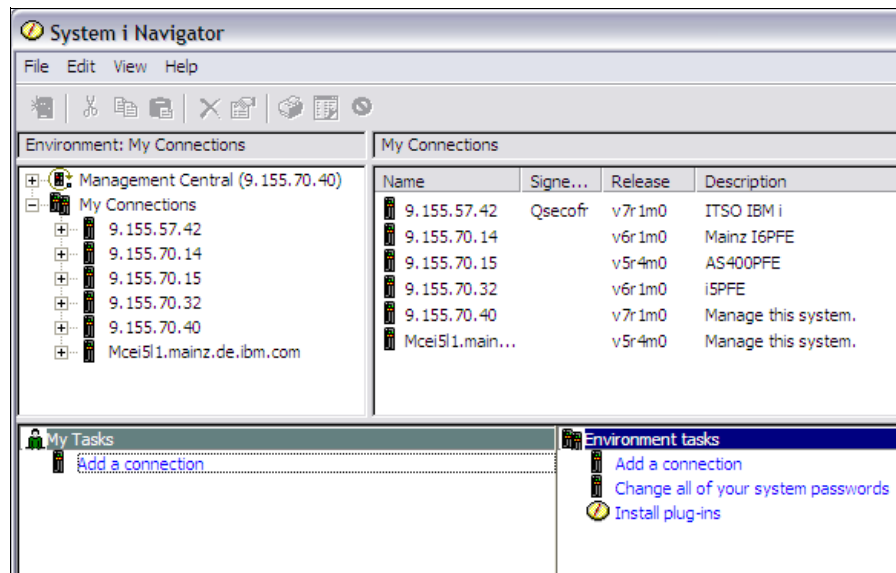


Figure 11-13 System i Navigator initial window

2. Click the expand icon to the server that you want to add the logical volume to, and sign on to that server, as shown in Figure 11-14.



Figure 11-14 System i Navigator Signon to System i window

3. As shown in Figure 11-15, expand **Configuration and Service** → **Hardware** → **Disk Units**.

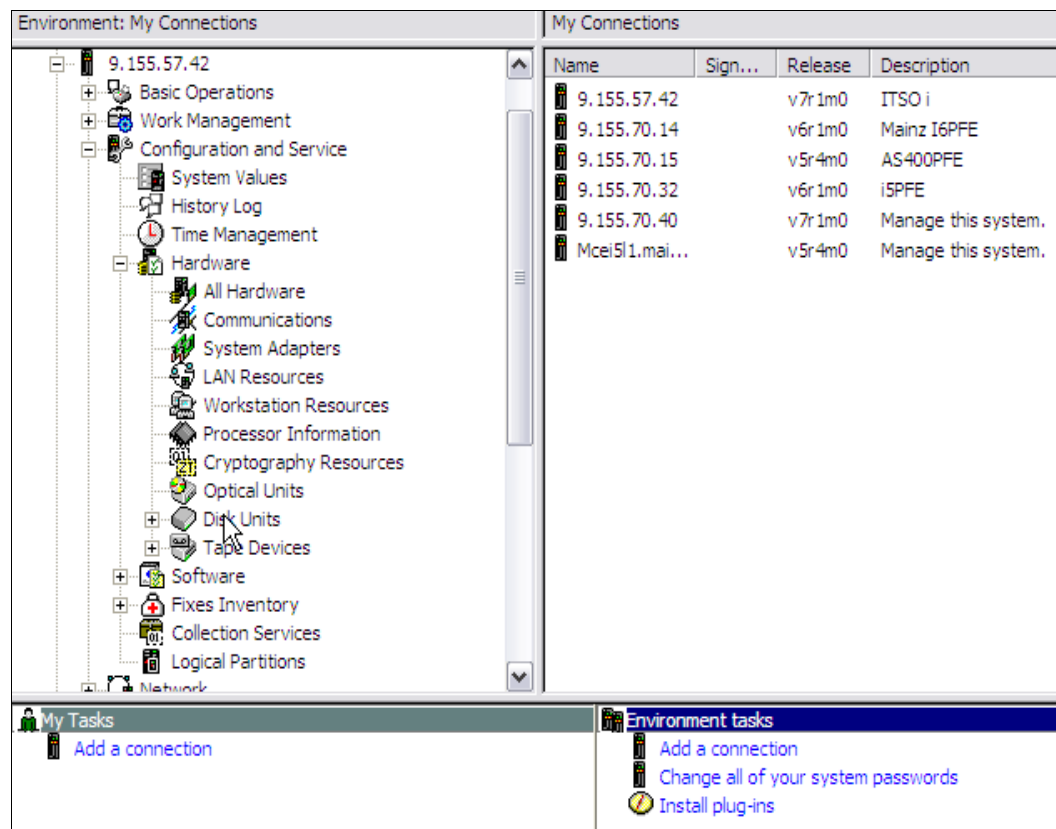


Figure 11-15 System i Navigator Disk Units

4. In the Service Tools Sign-on window (Figure 11-16), enter your Service tools ID and password. Then click **OK**.

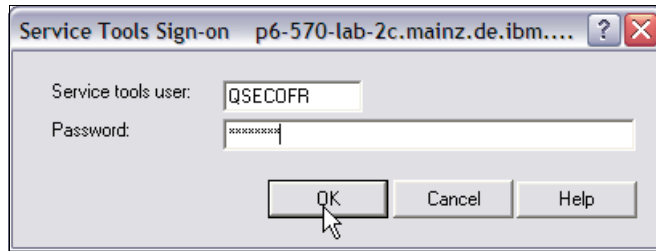


Figure 11-16 SST Sign-on

5. Right-click **Disk Pools** and select **New Disk Pool**.
6. In the New Disk Pool wizard, on the window that opens, click **Next**.
7. In the New Disk Pool window (Figure 11-17), complete these steps:
  - a. For Type of disk pool, select **Primary**.
  - b. Enter a name for the new disk pool.
  - c. For Database, leave the default setting **Generated by the system**.
  - d. Select the method that matches the type of logical volume you are adding. If you do not select one of these methods, you see all available disks. In this example, select **Protect the data in this disk pool**.
  - e. Click **OK**.



Figure 11-17 Defining a new disk pool

- In the New Disk Pool - Select Disk Pool window (Figure 9-15), click **Next**.

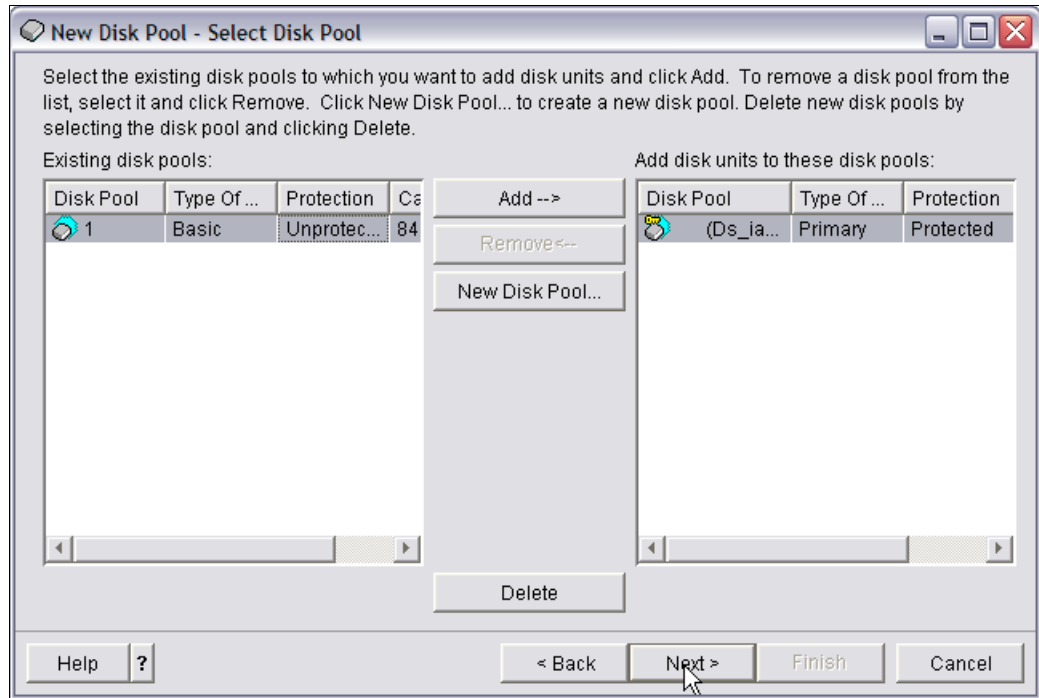


Figure 11-18 New Disk Pool - Select Disk Pool panel

- In the Disk Pool New Disk Pool - Add Disks Units window (Figure 11-19), click **Add Parity-Protected Disks**. The disks or LUNs to be added can be partially protected.

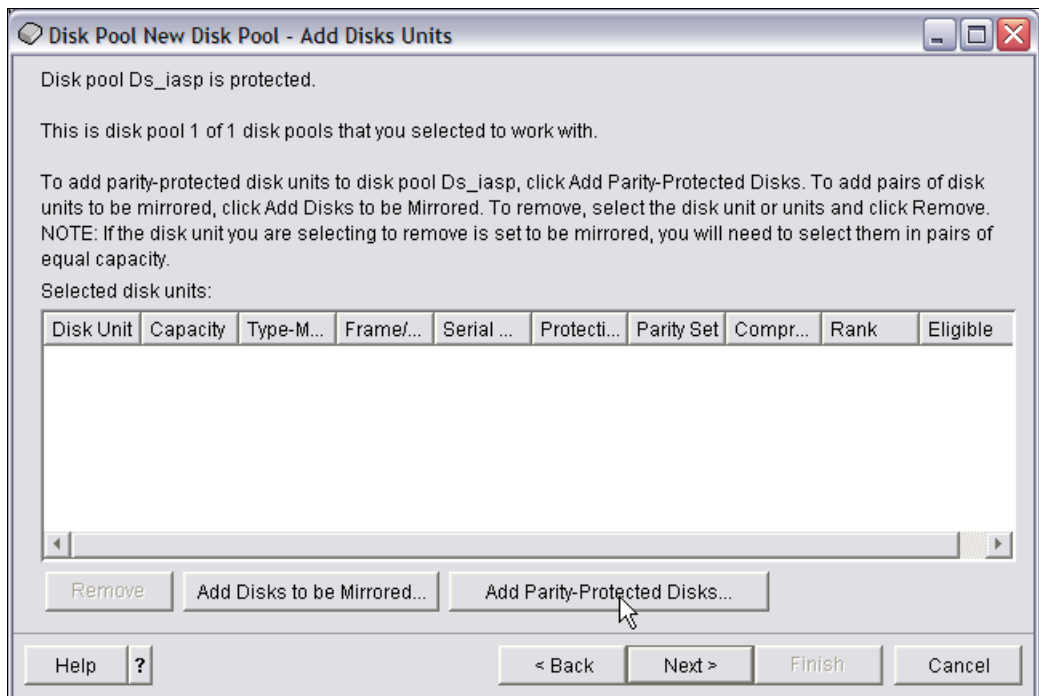


Figure 11-19 Add disks to disk pool

10. Highlight the disks you want to add to the disk pool and click **Add** when a list of non-configured units appears, as shown in Figure 11-20.

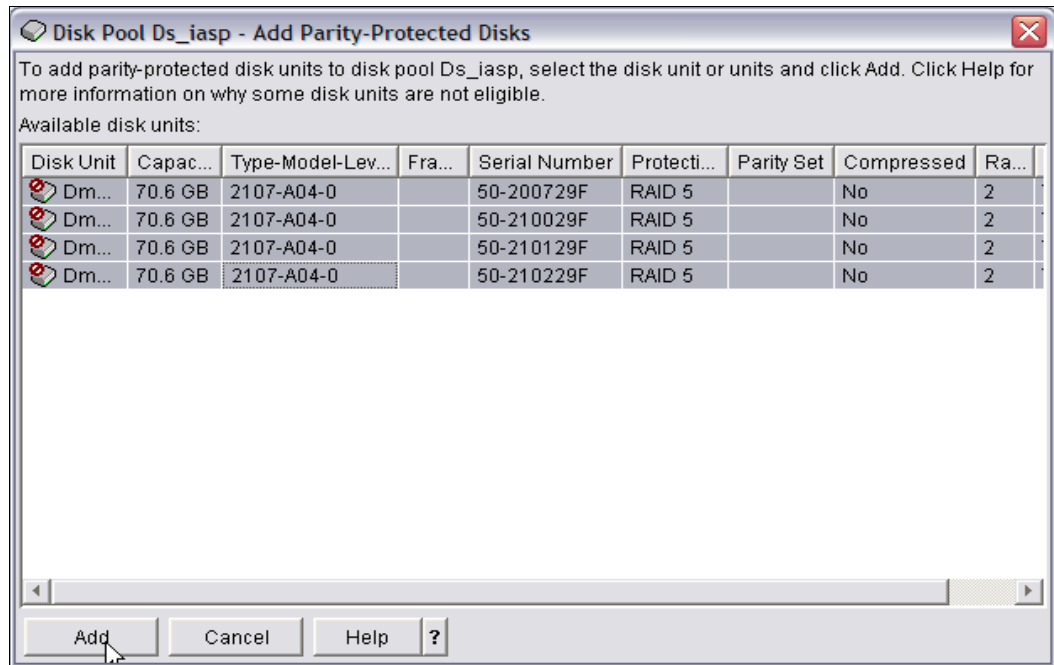


Figure 11-20 Select the disks to add to the disk pool

11. In the Disk Pool New Disk Pool - Add Disk Units window (Figure 11-21), click **Next**.

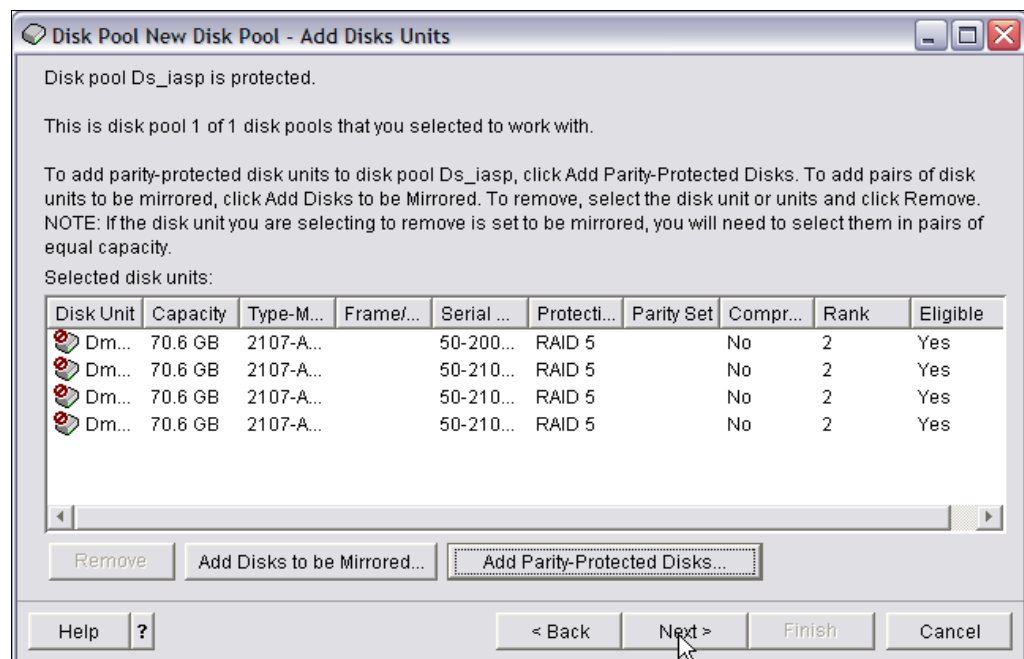


Figure 11-21 Confirming disks to add to disk pool



12. In the New Disk Pool - Summary window (Figure 11-22), click **Finish** to add disks to the disk pool.

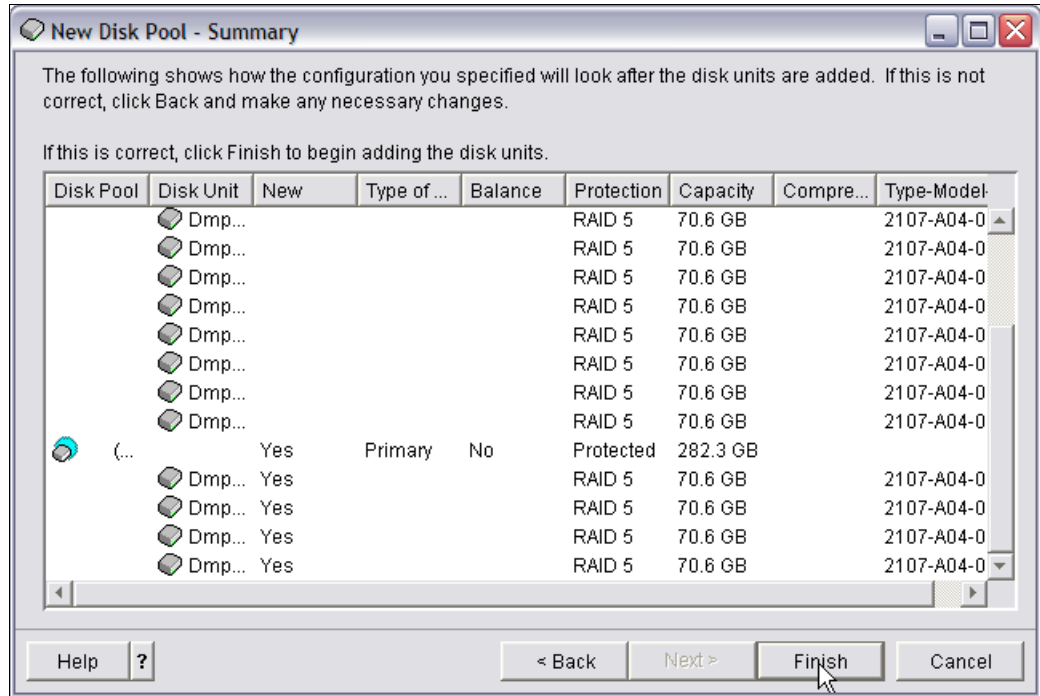


Figure 11-22 New Disk Pool - Summary

13. Respond to any message windows that appear. After you take action on any messages, the New Disk Pool Status window shown in Figure 11-23 appears and shows progress.

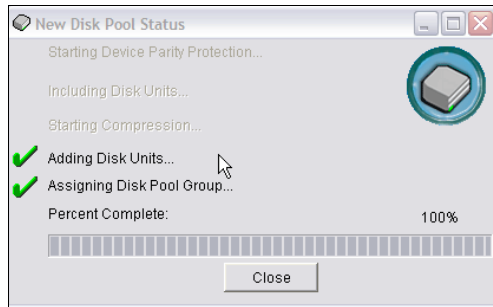


Figure 11-23 New Disk Pool Status

**Tip:** This step might take time, depending on the number and size of the logical units that you add.

14. When complete, click **OK** in the action window shown in Figure 11-24.

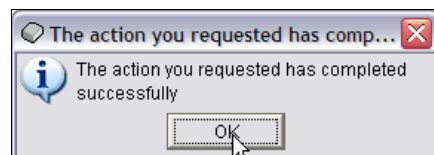


Figure 11-24 Disks added successfully to disk pool

15. Click **OK**, if you see the message that the created IASP is ready to start mirroring. The System i Navigator Disk Pools window (Figure 11-25) shows the new disk pool.

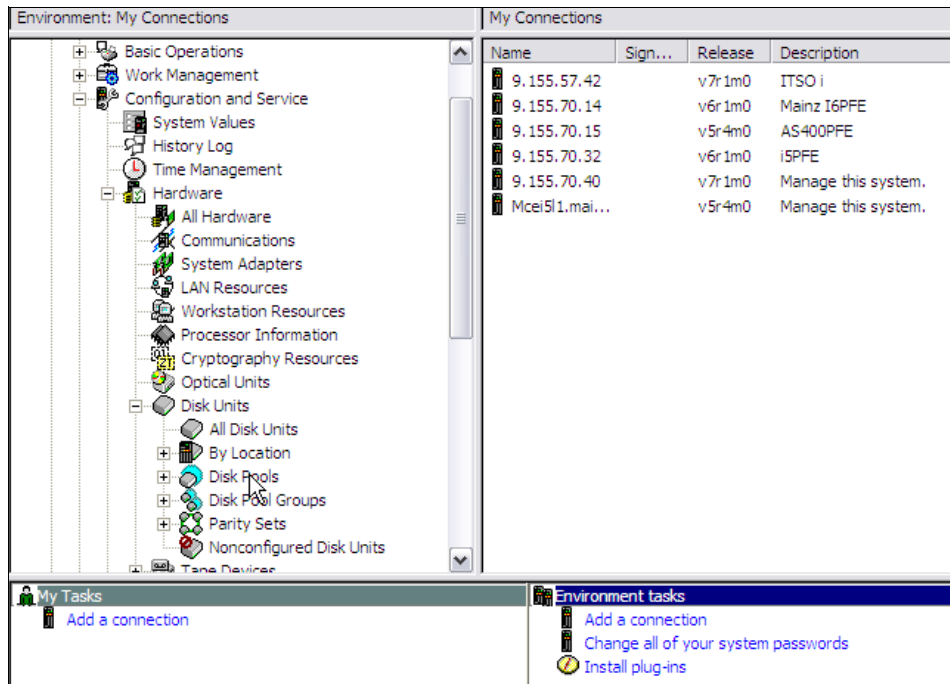


Figure 11-25 New disk pool shown in System i Navigator

To see the logical volume, as shown in Figure 11-26, expand **Configuration and Service** → **Hardware** → **Disk Pools**, and click the disk pool that you have just created.

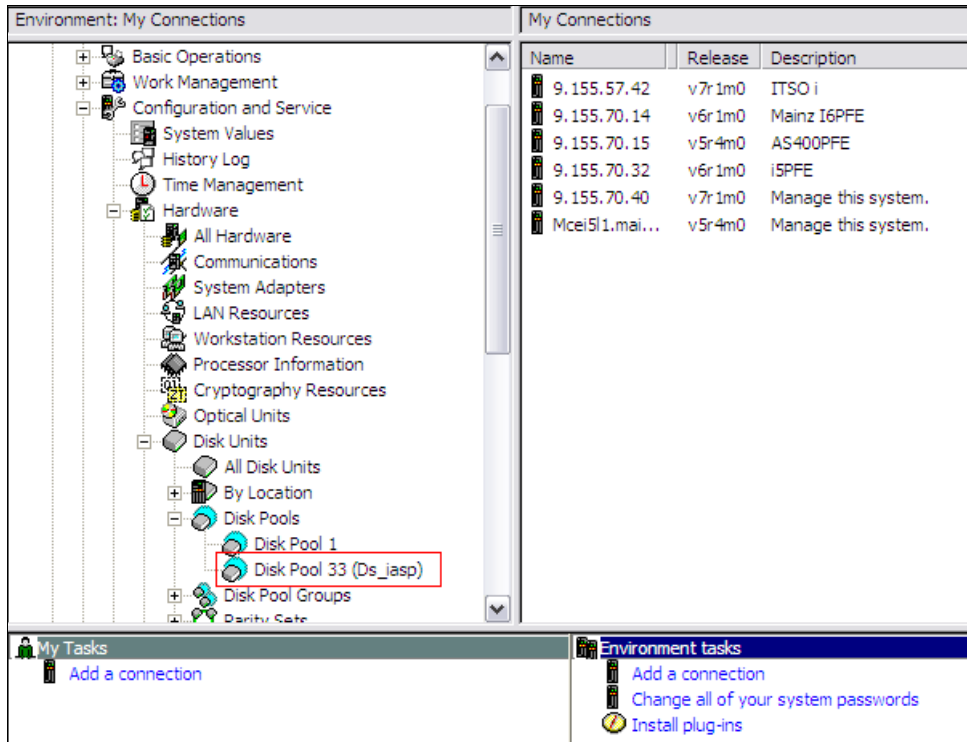


Figure 11-26 New logical volumes shown in System i Navigator

## 11.7 Booting from SAN

Traditionally, IBM i, and formerly System i, hosts required the use of an internal disk as a boot drive or load source unit (LSU or LS). The boot from SAN support was added in i5/OS V5R3M5 and later.

### 11.7.1 Requirements for boot from SAN

You can implement the external load source unit with Power5, Power6, Power7 servers, and with Power6 and Power7 technology-based Blade servers. You can also use it with System p servers 9117-570, 9119-590, and 9119-595 that allow IBM i and a partition.

Boot from SAN is supported on natively attached DS8000, and on DS8000 attached through VIOS NPIV or VIOS VSCSI.

With natively attached DS8000, the external load source requires IBM i level V5R3M5 or later.

**Important:** Not all of the IBM i software levels are supported on a given hardware. Table 11-1 on page 203 gives an overview of IBM i levels that can run on a particular hardware type. For a complete and up-to-date list, see the IBM SSIC website:

<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

The external load source unit can be connected with any of the IOAs listed in 11.2, “Using Fibre Channel adapters” on page 205 for native attachment. When implementing boot from SAN with IOP-based adapters, attach to the boot from SAN, IOP feature number #2847 to the connect the load source unit. The IOAs that connect the other LUNs can be attached to IOPs feature number #2844.

With native attachment to DS8000, multipath can be established for the load source LUN when connecting with IOP-based or IOP-less adapters on IBM i levels V6R1 or V7R1. However, multipath is not possible for the load source LUN on IBM i V5R4.

Boot from SAN is supported on any attachment to DS8000 with VIOS NPIV or VIOS VSCSI to IBM i.

When DS8000 is connected through VIOS to IBM i on POWER servers, multipath will be established for the external load source, which is connected to two or more virtual adapters, each virtual adapter assigned to a separate VIOS. This way, resiliency for the boot disk is provided even if one VIOS fails. To achieve such multipathing, you need VIOS level 2.1.2 or later.

### 11.7.2 Tagging the load source LUN

To enable IBM i to boot from an external load source, the adapter that is to be used for this external LUN must be tagged as the *load source I/O device*.

For IBM i in a partition of a POWER server or System p server, the external load source is tagged in the HMC, in the profile of the IBM i LPAR. With natively connected DS8000, tag the physical adapter connecting the load source LUN. With DS8000 connected through VIOS, tag the virtual adapter to the boot LUN that is assigned or mapped.

To tag an adapter as the load source I/O device in HMC, complete the following steps:

1. In HMC, expand the pop-up menu at the IBM i partition and select **Configuration** → **Manage profiles** (Figure 11-27).

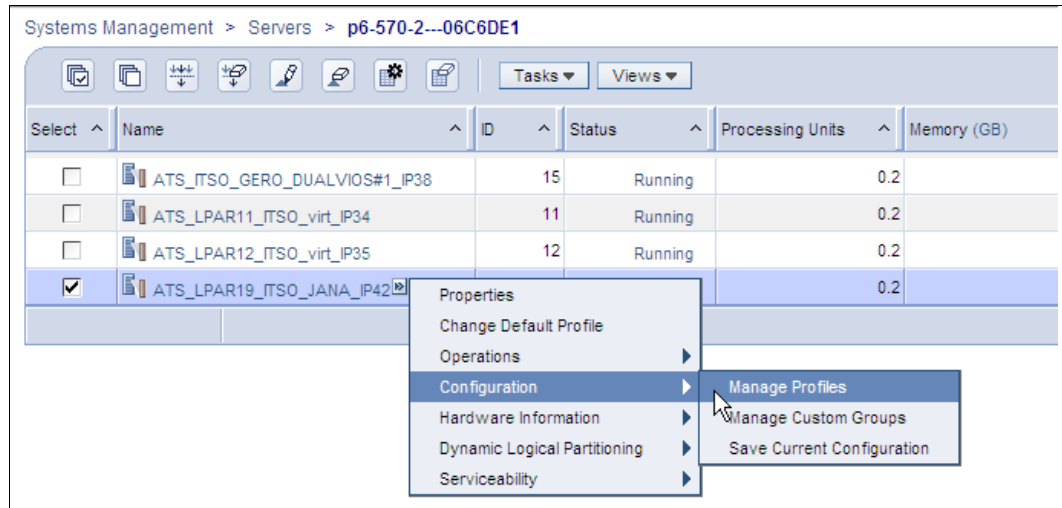


Figure 11-27 HMC: Manage Profiles

2. In the Manage Profiles window, select the IBM i LPAR, expand the Actions menu and select **Edit**, as shown in Figure 11-28.

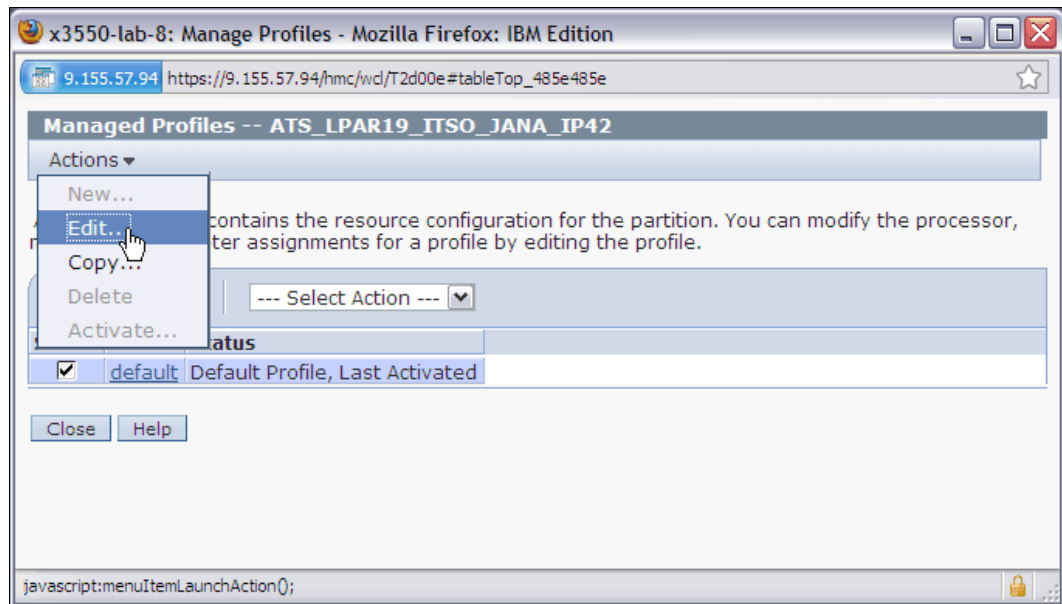


Figure 11-28 HMC: Edit profile

3. Select the table **Tagged I/O** and expand the pull-down menu under Load source (Figure 11-29). Select the adapter that connects the boot from SAN volume. This figure shows an example of tagging an 8 GB IOP-less Fibre Channel adapter through the load source that is connected.

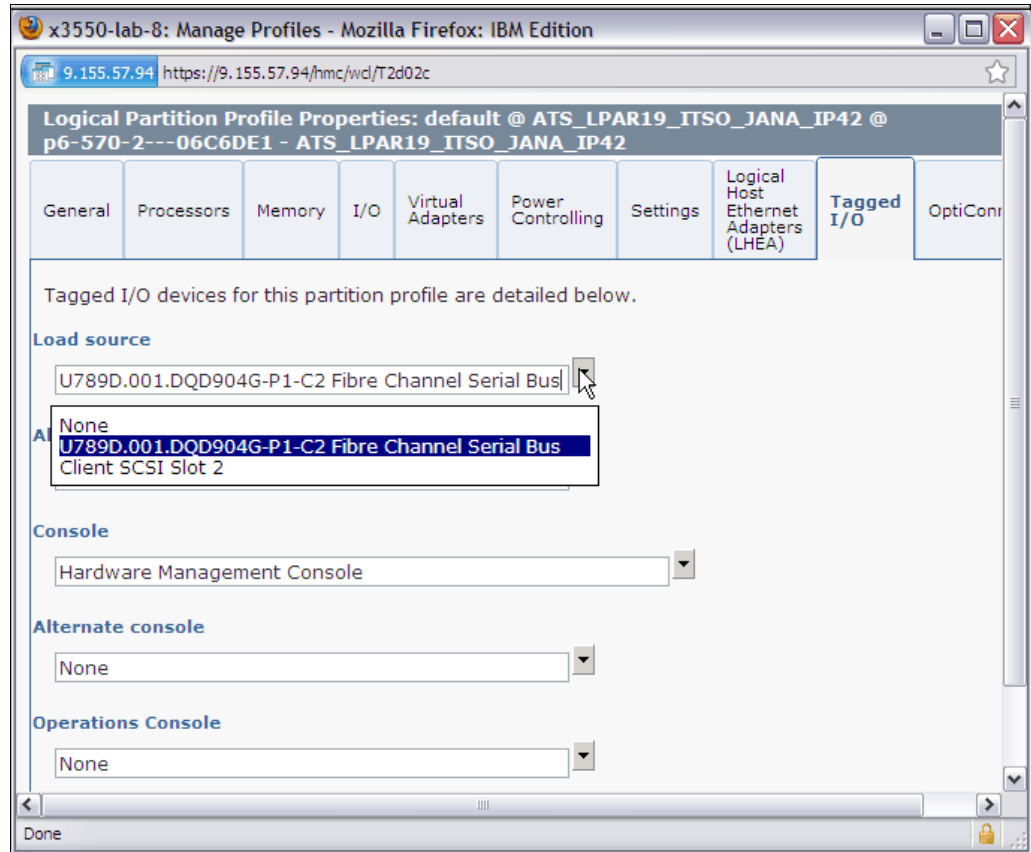


Figure 11-29 HMC: Select the adapter that connects external Load Source

When IBM i resides on a Blade server and the DS8000 is connected with VIOS NPIV or Virtual SCSI, the load source adapter defaults to the single client adapter that is initially in the IBM i partition.

Figure 11-30 shows a partition to be created with Integrated Virtualization Manager (IVM) on a Blade server. After the partition is created, you can add other virtual adapters and change the tagging of the load source on the partition properties page in IVM.

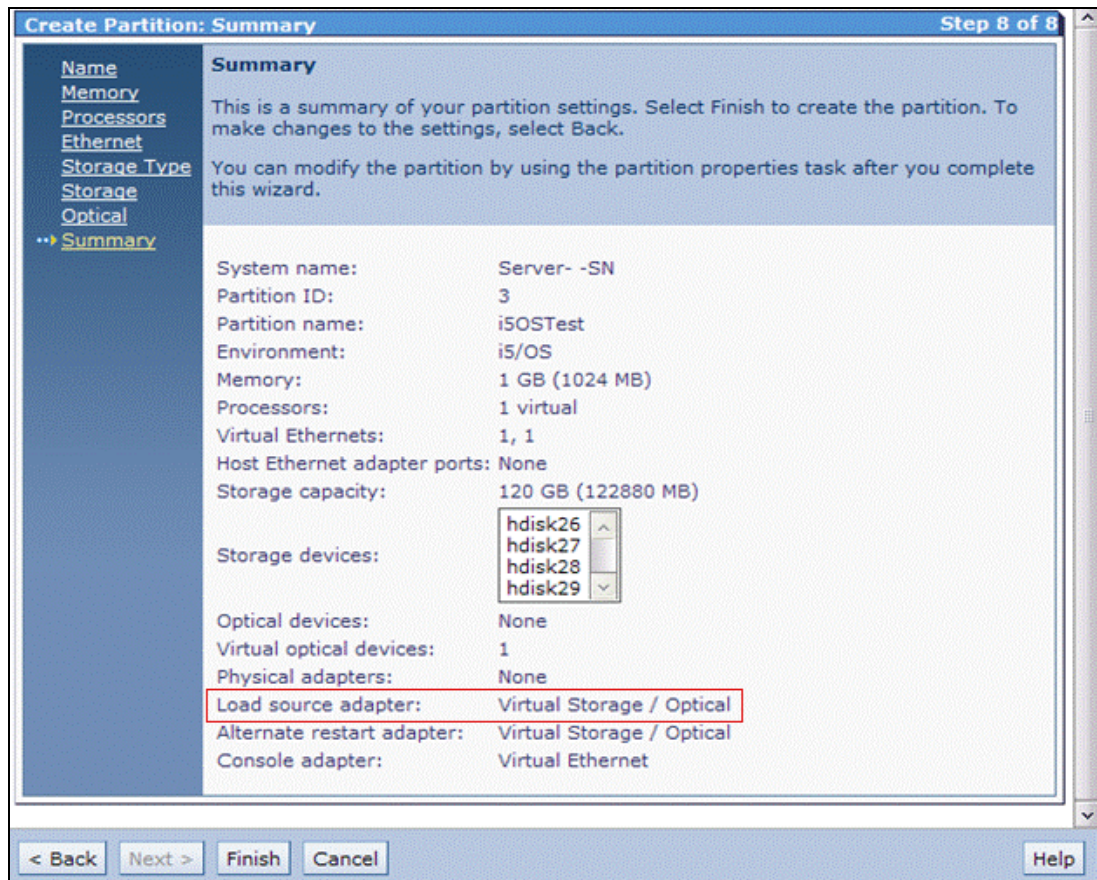


Figure 11-30 Load source with IVM

## 11.8 Installing IBM i with boot from SAN through VIOS NPIV

This section provides information about installing IBM i on an LPAR of the POWER server to the DS8000 that is to boot from SAN, and connected through VIOS NPIV.

### 11.8.1 Scenario configuration

The best practice is to connect IBM i with two VIOS NPIV in multipath. However, for the purpose of the scenario in this section, IBM i is connected through two virtual Fibre Channel adapters, each of them assigned to a port from a separate physical adapter in the same VIOS.

Figure 11-31 shows that in IBM i, the virtual Fibre Channel adapter ID 14 is connected to the Fibre Channel adapter ID 34 in VIOS, and the virtual adapter 15 is connected to the VIOS adapter 35.

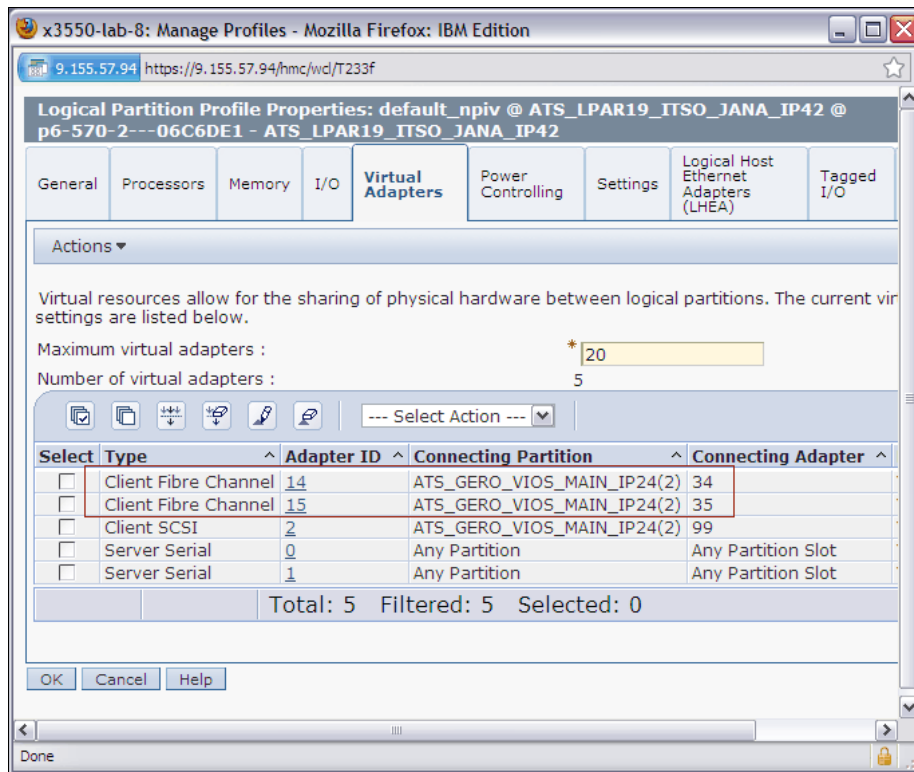


Figure 11-31 Assigning virtual Fibre Channel adapters

Figure 11-32 shows both WWPNs for the virtual Fibre Channel adapter 14 in the Virtual Fibre Channel Adapter Properties window.

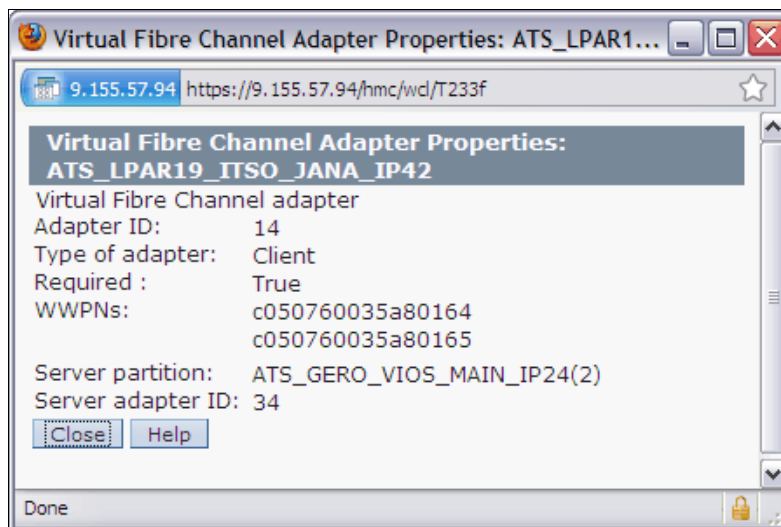


Figure 11-32 WWPNs of the virtual Fibre Channel adapter

In the DS8000, the load source and other LUNs are assigned to both WWPN from each virtual Fibre Channel adapter.

In VIOS, the virtual Fibre Channel adapters from IBM i correspond to virtual devices vfchost12 and vfchost13 (Figure 11-33).

```

$ lsdev -slots
# Slot                Description          Device(s)
HEA 1                 Logical I/O Slot    lhea0 ent0
U789D.001.DQD904G-P1-T3 Logical I/O Slot    pci2 sissas0
U9117.MMA.06C6DE1-V2-C0 Virtual I/O Slot    vsa0
U9117.MMA.06C6DE1-V2-C2 Virtual I/O Slot    vasi0
U9117.MMA.06C6DE1-V2-C11 Virtual I/O Slot    ent1
U9117.MMA.06C6DE1-V2-C20 Virtual I/O Slot    vfchost0
U9117.MMA.06C6DE1-V2-C21 Virtual I/O Slot    vfchost1
U9117.MMA.06C6DE1-V2-C22 Virtual I/O Slot    vfchost2
U9117.MMA.06C6DE1-V2-C23 Virtual I/O Slot    vfchost3
U9117.MMA.06C6DE1-V2-C24 Virtual I/O Slot    vfchost10
U9117.MMA.06C6DE1-V2-C25 Virtual I/O Slot    vfchost11
U9117.MMA.06C6DE1-V2-C30 Virtual I/O Slot    vfchost4
U9117.MMA.06C6DE1-V2-C31 Virtual I/O Slot    vfchost5
U9117.MMA.06C6DE1-V2-C32 Virtual I/O Slot    vfchost6
U9117.MMA.06C6DE1-V2-C33 Virtual I/O Slot    vfchost7
U9117.MMA.06C6DE1-V2-C34 Virtual I/O Slot vfchost12
U9117.MMA.06C6DE1-V2-C35 Virtual I/O Slot vfchost13
U9117.MMA.06C6DE1-V2-C70 Virtual I/O Slot    vfchost8
U9117.MMA.06C6DE1-V2-C71 Virtual I/O Slot    vfchost9

```

Figure 11-33 Virtual devices in VIOS

Before installation, the virtual adapters still report a status of NOT\_LOGGED\_IN (Figure 11-34).

```

$ lsmmap -vadapter vfchost12 -npiv
Name                Physloc                CIntID CIntName          CIntOS
-----
vfchost12          U9117.MMA.06C6DE1-V2-C34          19

Status:NOT_LOGGED_IN
FC name:fcs2                FC loc code:U789D.001.DQD904G-P1-C6-T1
Ports logged in:0
Flags:4<NOT_LOGGED>
VFC client name:                VFC client DRC:

$ lsmmap -vadapter vfchost13 -npiv
Name                Physloc                CIntID CIntName          CIntOS
-----
vfchost13          U9117.MMA.06C6DE1-V2-C35          19

Status:NOT_LOGGED_IN
FC name:fcs3                FC loc code:U789D.001.DQD904G-P1-C6-T2
Ports logged in:0
Flags:4<NOT_LOGGED>
VFC client name:                VFC client DRC:

```

Figure 11-34 Reporting of virtual Fibre Channel adapters before installation



In the IBM i profile, the virtual Fibre Channel adapter ID 14 is tagged as a load source. Because the installation is done from the images of DVD drives in the VIOS virtual repository, the virtual SCSI adapter connecting the repository is tagged as the alternate restart device, as shown in Figure 11-35.

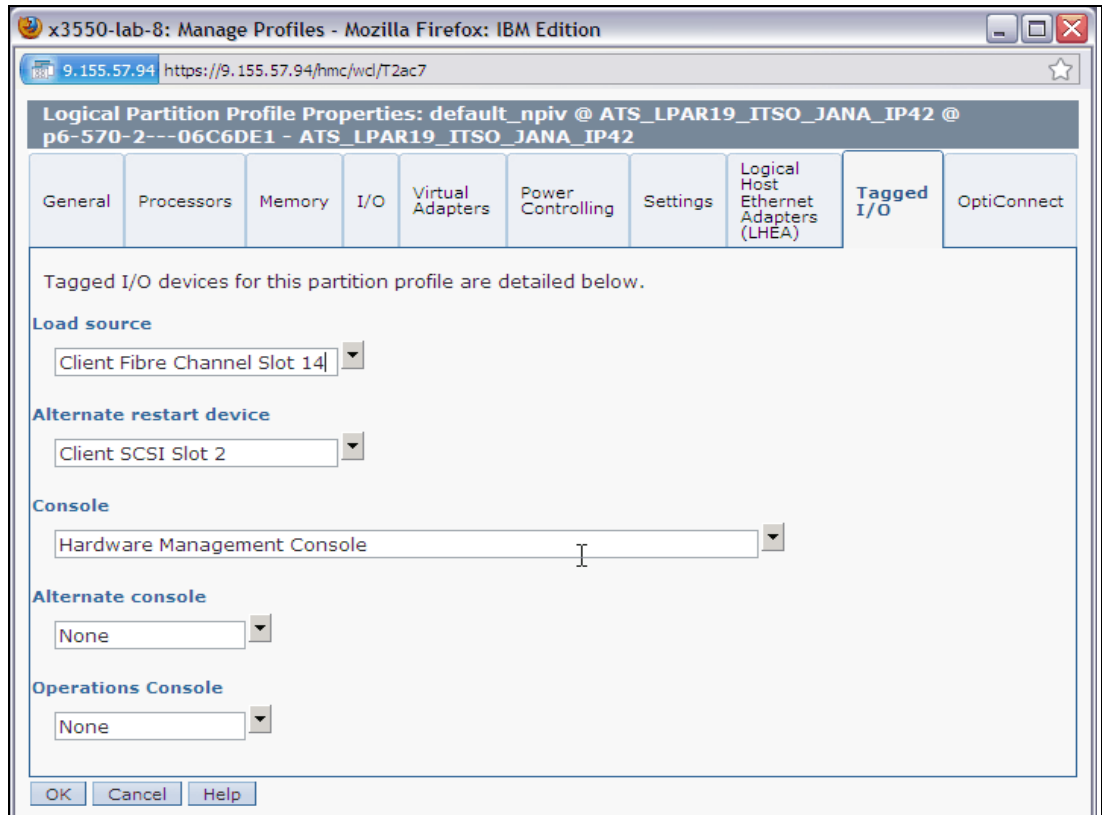


Figure 11-35 Tagged virtual adapters

The IPL source must be specified as D, under the **Settings** tab in the Partition Properties window (Figure 11-36).

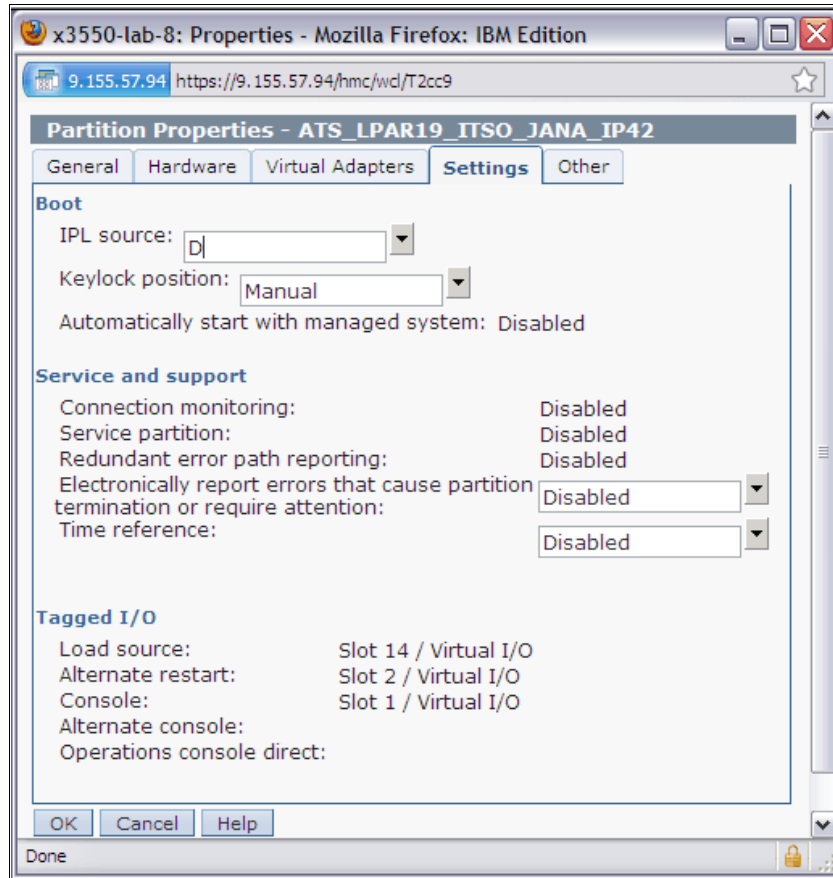


Figure 11-36 IPL settings

To start the installation, activate the IBM i LPAR, and then complete the following steps:

1. Select the language group for IBM i, and then select option 1, Install Licensed Internal Code (Figure 11-37).

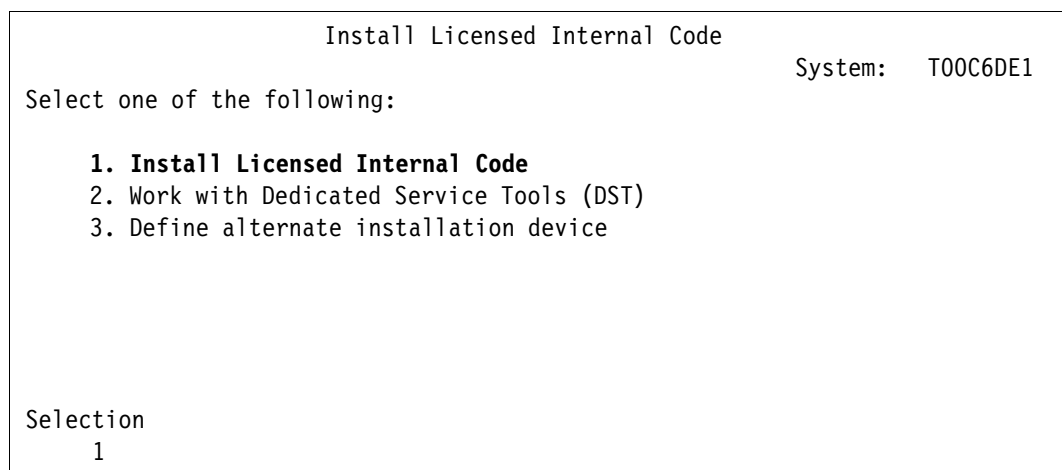


Figure 11-37 Installing License Internal Code

2. Select the LUN to be the load source device in the Select Load Source Device window, as shown in Figure 11-38.

Select Load Source Device									
Type 1 to select, press Enter.									
Opt	Serial Number	Type	Model	Sys Bus	Sys Card	I/O Adapter	I/O Bus	Ctl	Dev
1	50-210A29F	2107	A04	255	14	0	0	0	3
	50-210929F	2107	A04	255	15	0	0	0	2
	50-200929F	2107	A04	255	14	0	0	0	1
	50-200829F	2107	A04	255	15	0	0	0	0

F3=Exit                      F5=Refresh                      F12=Cancel

Figure 11-38 Selecting the load source device

3. After confirming the selection of the load source disk, select option 2, Install Licensed Internal Code and Initialize system (Figure 11-39) to proceed with IBM i installation.

Install Licensed Internal Code (LIC)						
Disk selected to write the Licensed Internal Code to:						
	Serial Number	Type	Model	I/O Bus	Controller	Device
	50-210A29F	2107	A04	0	0	3
Select one of the following:						
1. Restore Licensed Internal Code						
2. <b>Install Licensed Internal Code and Initialize system</b>						
3. Install Licensed Internal Code and Recover Configuration						
4. Install Licensed Internal Code and Restore Disk Unit Data						
5. Install Licensed Internal Code and Upgrade Load Source						
Selection						
2						
F3=Exit                      F12=Cancel						

Figure 11-39 Installing Licensed Internal Code and initialize system

## 11.8.2 Booting from SAN and cloning

When booting from SAN support, you can take advantage of certain advanced features that are available with the DS8000 series and copy services functions. These functions provide the ability to perform a rapid copy of the data held on DS8000 LUNs. Therefore, when you have a system that only has external LUNs with no internal drives, you can create a *clone* of your system.

**Important:** For the instances in this section, a clone is a copy of a system that only uses external LUNs. Therefore, boot, or IPL, from SAN, is a prerequisite for this function.

## Why consider cloning

By using the cloning capability, you can create a complete copy of your entire system in minutes. You can then use this copy in any way you want, for example, you can use it to minimize your backup windows, or protect yourself from a failure during an upgrade, or even use it as a fast way to provide yourself with a backup or test system. You can do all of these tasks using cloning with minimal impact to your production operations.

## When to use cloning

You might want to use cloning in the following circumstances:

- ▶ You need enough free capacity on your external storage unit to accommodate the clone. If Metro Mirror or Global Mirror are used, you need enough bandwidth on the links between the DS8000 primary and secondary site.
- ▶ Consider that the copy services need enough resources on the primary and secondary DS8000 to ensure that there will be only a minimal performance impact on the production IBM i system.
- ▶ Do not attach a clone to your network, until you have resolved any potential conflicts that the clone has with the parent system.

## 11.9 Migrating

For many System i clients, migrating to the DS8000 is best achieved by using traditional save and restore techniques. However, several alternatives are available to consider. For more information, see 11.9.1, “Metro Mirror and Global Copy” on page 244, or 11.9.2, “IBM i data migration” on page 244.

### 11.9.1 Metro Mirror and Global Copy

Depending on the existing configuration, it might be possible to use Metro Mirror or Global Copy to migrate from an IBM TotalStorage Enterprise Storage Server (ESS) or any other DS8000 family model to a newer DS8000 model. It might also be possible to use any combination of external storage units that support Metro Mirror and Global Copy. For more information about Metro Mirror and Global Copy, see *IBM System Storage DS8000: Copy Services in Open Environments*, SG24-6788.

You can use DS8000 copy services for migration when the IBM i has the load source unit and all the other disks on a DS8000, and you plan to migrate the IBM i workload to another DS8000 on 1000 km remote location, with the two storage systems connected through dark fiber. To do this, you must establish a Global Copy relationship with IBM i LUNs at the remote DS8000.

When ready to migrate to the remote site, you must also do a complete shutdown of the IBM i to ensure all data is written to disk, and wait until Global Copy transfers all the updates to the remote DS8000 with no out of synch tracks. Additionally, unassign the DS8000 LUNs on local site and assign the DS8000 LUNs to the remote IBM i LPAR.

After the IPL of the IBM i, the new DS8000 LUNs are recognized by IBM i.

### 11.9.2 IBM i data migration

It is also possible to use native IBM i functions to migrate data from existing disks to the DS8000, whether the existing disks are internal or external. When you assign the new DS8000 logical volumes to the System i, initially they are non-configured. For more information, see 11.6.8, “Adding volumes to the System i configuration” on page 224.

If you add the new units and choose to spread data, IBM i automatically migrates data from the existing disks onto the new logical units.

You can then use the **STRASPBAL TYPE(\*ENDALC)** IBM i command to mark the units to remove from the configuration, as shown in Figure 11-40. This command can help reduce the downtime associated with removing a disk unit and keeps new allocations away from the marked units.

```

                                Start ASP Balance (STRASPBAL)

Type choices, press Enter.

Balance type . . . . . > *ENDALC          *CAPACITY, *USAGE, *HSM...
Storage unit . . . . .                    1-4094
      + for more values

                                                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
  
```

Figure 11-40 Ending allocation for existing disk units

When you later run the **STRASPBAL TYPE(\*MOVDTA)** IBM i command, all data is moved from the marked units to other units in the same ASP, as shown in Figure 11-41. You must have sufficient new capacity to allow the data to be migrated.

```

                                Start ASP Balance (STRASPBAL)

Type choices, press Enter.

Balance type . . . . . > *MOVDTA          *CAPACITY, *USAGE, *HSM...
Time limit . . . . .                    1-9999 minutes, *NOMAX

                                                                Bottom
F3=Exit  F4=Prompt  F5=Refresh  F12=Cancel  F13=How to use this display
F24=More keys
  
```

Figure 11-41 Moving data from units marked \*ENDALC

You can specify a time limit that the function is to run for each ASP being balanced, or the balance can be set to run to completion. If you must end the balance function prior to this, use the End ASP Balance (**ENDASPBAL**) command. A message is sent to the system history (QHST) log when the balancing function is started for each ASP. A message is also sent to the QHST log when the balancing function completes or is ended. If the balance function is run for a few hours and then stopped, it continues from where it left off when the balance function restarts so that the balancing can be run during off-hours over several days.

To remove the old units from the configuration, you must use **DST** and reload the system or partition with **IPL**. With this method, you can remove the existing storage units over a period of time. This method requires that both the old and new units are attached to the system at the same time. Therefore, additional IOPs and IOAs might be required if you are migrating from an ESS to a DS8000. It might be possible in your environment to reallocate logical volumes to other IOAs, but careful planning and implementation are required.





## IBM System z considerations

This chapter provides the specifics of attaching the IBM System Storage DS8000 series system to System z hosts.

The following topics are covered:

- ▶ Connectivity considerations
- ▶ Operating system prerequisites and enhancements
- ▶ Considerations for z/OS
- ▶ DS8000 device definition and zDAC - z/OS FICON Discovery and Auto-Configuration feature
- ▶ Extended Address Volume (EAV) support
- ▶ FICON specifics for a z/OS environment
- ▶ z/VM considerations
- ▶ VSE/ESA and z/VSE considerations
- ▶ I/O Priority Manager for z/OS
- ▶ TPC-R V5.1 in a z/OS environment
- ▶ Full Disk Encryption (FDE)

## 12.1 Connectivity considerations

The DS8000 storage system connects to System z hosts using FICON channels, with the addition of FCP connectivity for Linux for System z hosts. For more information, see the *System z Connectivity Handbook*, SG24-5444:

<http://www.redbooks.ibm.com/abstracts/sg245444.html>

### 12.1.1 FICON

Check for dependencies on the host hardware driver level and the supported feature codes. Your IBM service representative can help you determine your current hardware driver level on your mainframe processor complex. For the IBM z9® and z10 servers, FICON Express4 LX (FC3321 at 10 km and FC3324 at 4 km) and FICON Express4 SX (FC3322) are available. In addition, z10 supports the new FICON Express8 LX (FC3325 at 10 km) and FICON Express8 SX (FC3326).

### 12.1.2 LINUX FCP connectivity

You can use either direct or switched attachment to attach a storage unit to a System z host system that runs Novell SUSE Linux Enterprise Server10 SP1 or Red Hat Enterprise Linux 5.1 with current maintenance updates for FICON.

FCP attachment to Linux on System z systems can be done through a switched-fabric or direct attachment configuration. For more information, see Chapter 6, “Linux considerations” on page 81.

## 12.2 Operating system prerequisites and enhancements

The following minimum software levels are required to support the DS8800:

- ▶ z/OS V1.8
- ▶ z/VM V5.4
- ▶ IBM z/VSE® V4.2
- ▶ Transaction Processing Facility (TPF) V4.1 with PTF
- ▶ Novell SUSE Linux Enterprise Server10 SP1
- ▶ Red Hat Enterprise Linux 5.1

Certain functions of the DS8000 require later software levels than the minimum levels listed here. See the IBM System Storage Interoperation Center (SSIC) at the following website:

<http://www.ibm.com/systems/support/storage/ssic/interoperability.wss>

**Tip:** In addition to the IBM System Storage Interoperation Center (SSIC), see the Preventive Service Planning (PSP) bucket of the 2107 devices for software updates.

The PSP information can be found in the IBM Resource Link® website:

<http://www.ibm.com/servers/resourceLink/svc03100.nsf?OpenDatabase>

You need to register for an IBM registration ID (IBM ID) before you can sign in to the website. Under the “Planning” section, go to “Tools” to download the relevant PSP package.



## 12.3 Considerations for z/OS

This section describes program enhancements that z/OS has implemented to support the characteristics of the DS8000, and provides guidelines for the definition of parallel access volumes (PAVs).

### 12.3.1 Program enhancements for z/OS

These relevant Data Facility Storage Management Subsystem (DFSMS) small program enhancements (SPEs) were introduced in z/OS for support of all DS8000 models:

- ▶ Scalability support
- ▶ Large Volume support (LV):  
With LV support, it is possible to address a capacity up to 3.964 PB. To accommodate the needs of installations that require super large volumes, IBM developed a volume named Extended Address Volume (EAV), recognized by the host as 3390 Model A (3390A).
- ▶ Read availability mask support
- ▶ Initial program load enhancements
- ▶ DS8000 device definition
- ▶ Read control unit and device recognition for DS8000
- ▶ Performance statistics
- ▶ Resource Measurement Facility
- ▶ System Management Facilities
- ▶ Migration considerations
- ▶ Coexistence considerations

Many of these program enhancements are initially available as authorized program analysis reports (APARs) and Program Temporary Fix (PTF) for the current releases of z/OS, and are later integrated into follow-on releases of z/OS. For this reason, see the DS8000 Preventive Service Planner (PSP) bucket for your current release of z/OS.

#### Scalability support

The I/O supervisor (IOS) recovery was designed to support a small number of devices per control unit, and a unit check was presented on all devices at failover. It does not scale well with a DS8000, which has the capability to scale up to 65,280 devices. Under these circumstances, you can have CPU or spinlock contention, or exhausted storage under the 16 M line at device failover, or both.

Starting with z/OS V1.4 and higher for DS8000 software support, the IOS recovery is improved by consolidating unit checks at an LSS level instead of each disconnected device. This consolidation shortens the recovery time as a result of I/O errors. This enhancement is particularly important, because the DS8000 can have up to 65,280 devices in a storage facility.

#### Scalability benefits

With enhanced scalability support, the following benefits are possible:

- ▶ Common storage area (CSA) usage above and under the 16 M line is reduced.
- ▶ The IOS large block pool for error recovery processing and attention, and the state change interrupt processing are located above the 16 M line, thus reducing the storage demand under the 16 M line.
- ▶ Unit control blocks (UCB) are pinned during event notification facility signalling during channel path recovery.

- ▶ Scalability enhancements provide additional performance improvements as follows:
  - Bypassing dynamic pathing validation in channel recovery for reduced recovery I/Os
  - Reducing elapsed time by reducing the wait time in channel path recovery

### **Read availability mask support**

The dynamic channel path identifier management (DCM) provides you with the ability to define a pool of channels that are managed by the system. The channels are added and deleted from control units based on workload importance and availability needs. DCM attempts to avoid single points of failure when adding or deleting a managed channel by not selecting an interface on the control unit on the same I/O card.

### **Initial program load enhancements**

During the Initial Program Load (IPL) sequence, the channel subsystem selects a channel path to read from the system residence volume SYSRES device. Certain types of I/O errors on a channel path cause the IPL to fail even though there are alternate channel paths that might work. For example, consider a situation where you have a bad switch link on the first path but good links on the other paths. In this case, you cannot IPL, because the same faulty path is always chosen.

The channel subsystem and z/OS are enhanced to retry I/O over an alternate channel path. This circumvents IPL failures that are caused by the selection of the same faulty path to read from the SYSRES device.

## **12.3.2 DS8000 device definition**

To exploit the increase in the number of LSSs that can be added in the DS8000 (255 LSSs), the unit must be defined as 2107 in the Hardware Configuration Definition (HCD) or input/output configuration program. The host supports 255 logical control units when the DS8000 is defined as UNIT=2107. You must install the appropriate software to support this setup. If you do not have the required software support installed, you can define the DS8000 as UNIT=2105, but then, only the 16 logical control units (LCUs) of address group 0 can be used.

The Control Unit (CU) device is a 2107/242x for any DS8000 device type. In HCD you can use 3990 as a control-unit type for 3380 and 3390 devices. However, you must use 2107/242x as the control-unit type for 3380B, 3380A, 3390B, and 3390A devices. For more detailed information about how to configure DS8000 in z/OS environment, see the Redbooks publication, *I/O Configuration Using z/OS HCD and HCM*, SG24-7804, and the *z/OS V1R12 HCD User's Guide*, SC33-7988.

### **Defining additional subchannels**

Starting with z9 processors, you can define an additional subchannel set with ID 1, class 1 (SS 1) on top of the existing subchannel set class 0 (SS 0) in a channel subsystem. With this additional subchannel set, you can configure more than 2 x 63 K devices for a channel subsystem. With z/OS V1.7+, you can define PAV alias devices of type 3380A and 3390A, of the DS8000 (2107) DASD control units to SS 1. Device numbers can be duplicated across channel subsystems and subchannel sets. For more information about subchannel configuration on the I/O Definition File (IODF) and HCD, see the Redpaper publication, *Multiple Subchannel Sets: An Implementation View*, REDP-4387.

New Channel Subsystems (CSS) were also implemented on z10, z196, and z114. See the related installation and planning guide to verify total CSS availability on your host. For more details about the IBM zEnterprise® System, see the Redbooks publication, *IBM zEnterprise System Technical Introduction*, SG24-7832.

**Important:**

- ▶ The logical configuration on the host (HCD) must match with the device logical configuration in order to be able to use the devices.
- ▶ A layout of correspondence within UCBs host and device LUNs also needs to be kept updated to expedite each analysis to be done on your environment.

### **Read control unit and device recognition for DS8000**

The host system informs the attached DS8000 of its capabilities, so that it supports native DS8000 control unit and devices. The DS8000 then only returns information that is supported by the attached host system, using the self-description data, such as read data characteristics, sense ID, and read configuration data.

The following commands are able to recognize device type 2107 in their output:

- ▶ DEVSERV QDASD and PATHS command responses
- ▶ IDCAMS LISTDATA COUNTS, DSTATUS, STATUS, and IDCAMS SETCACHE

### **12.3.3 zDAC - z/OS FICON Discovery and Auto-Configuration feature**

The z/OS FICON Discovery and Auto-Configuration (zDAC) feature, which is deployed by the new z/Enterprise z196 and z114, is supported by DS8700 and DS8800 starting at LMC level 5.1. This function was developed to reduce the complexity and skills needed in a complex FICON production environment for changing the I/O configuration.

With zDAC, you can add storage subsystems to an existing I/O configuration in less time, depending on the policy that you defined. zDAC proposes new configurations that incorporate the current contents of your I/O Definition File (IODF) with additions for new and changed subsystems and their devices based on the policy that you defined in the Hardware Configuration Definition (HCD).

The following requirements must be met for using zDAC:

- ▶ Your System z must be a z/Enterprise (z196 or z114) running z/OS V1 R12 with PTFs or higher.
- ▶ LPAR must be authorized to make dynamic I/O Configuration (zDCM) changes on each processor hosting a discovery system.

Hardware Configuration Definition (HCD) and Hardware Configuration Management (HCM) users need to have authority for making dynamic I/O configuration changes.

As the name of this feature implies, zDAC provides two capabilities:

- ▶ **Discovery:**
  - Provides capability to discover attached disk connected to FICON fabrics
  - Detects new and older storage subsystems
  - Detects new control units on existing storage subsystems
  - Proposes control units and device numbering
  - Proposes paths for all discovery systems to newly discovered control units including the Sysplex scope

- ▶ Auto-configuration:
  - For high availability reasons, when zDAC proposes channel paths, it looks at single points of failure only. It does not consider any channel or port speed, or any current performance information.
  - After a storage subsystem is explored, the discovered information is compared against the target IODF, paths are proposed to new control units, and devices are displayed to the user. With that scope of discovery and auto configuration, the target work IODF is being updated.

When using zDAC, keep in mind the following considerations:

- ▶ Physical planning is still your responsibility.
- ▶ Logical configurations of the Storage Subsystem are still done by you.
- ▶ Decide what z/OS image should be allowed to use the new devices.
- ▶ Determine how the new devices must be numbered.
- ▶ Decide how many paths to the new control units need to be configured.

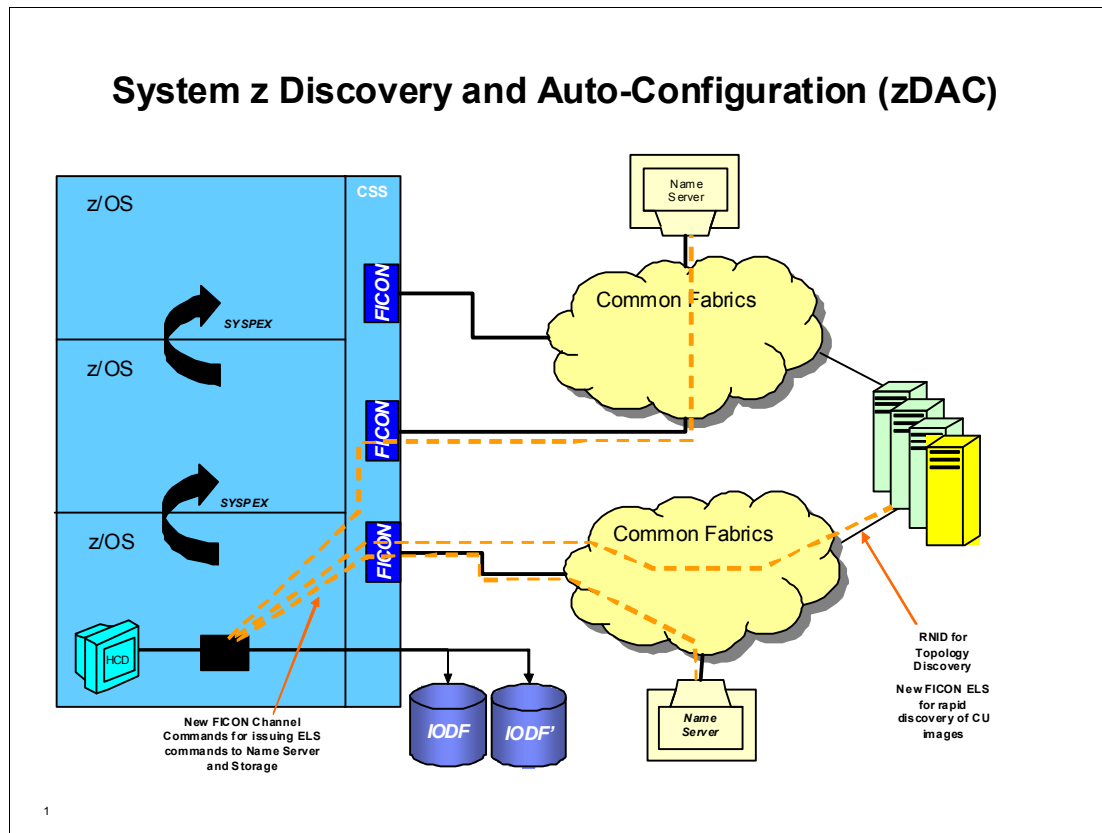


Figure 12-1 zDAC concept

For more information about zDAC, see the Redbooks publication, *z/OS V1R12 HCD User's Guide*, SC33-7988.

**Attention:** In z/OS environments, make sure that SAN switch zoning is enabled and configured properly, before start to use zDAC discovery.

## 12.3.4 Performance statistics

Because a logical volume is no longer allocated on a single RAID rank or single device adapter pair, the performance data is now provided with a set of rank performance statistics and extent pool statistics. The RAID rank reports are no longer reported by IBM Resource Measurement Facility™ (IBM RMF™) and IDCAMS LISTDATA batch reports. RMF and IDCAMS LISTDATA are enhanced to report logical volume statistics that are provided on the DS8000.

These reports consist of back-end counters that capture the activity between the cache and the ranks in the DS8000 for each individual logical volume. These rank and extent pool statistics are disk system-wide instead of volume-wide only.

## 12.3.5 Resource Measurement Facility

The Resource Measurement Facility (RMF) supports DS8000 from z/OS V1.4 and later with APAR number OA06476 and PTFs UA90079 and UA90080. RMF was enhanced to provide monitor I and III support for the DS8000 storage system.

### Statistics

The Disk Systems Postprocessor report contains two DS8000 sections, extent pool statistics, and rank statistics. These statistics are generated from SMF record 74 subtype 8:

- ▶ The extent pool statistics section provides capacity and performance information about allocated disk space. For each extent pool, it shows the real capacity and the number of real extents.
- ▶ The rank statistics section provides measurements about read and write operations in each rank of an extent pool. It also shows the number of arrays and the array width of all ranks. These values show the current configuration. The wider the rank, the more performance capability it has. By changing these values in your configuration, you can influence the throughput of your work.

Also, response and transfer statistics are available with the Postprocessor Cache Activity report generated from SMF record 74 subtype 5. These statistics are provided at the subsystem level in the Cache Subsystem Activity report and at the volume level in the Cache Device Activity report. In detail, RMF provides the average response time and byte transfer rate per read and write requests. These statistics are shown for the I/O activity, which is called *host adapter activity*, and for the transfer activity from hard disk to cache, which is called *disk activity*.

Reports were designed for reporting FICON channel utilization. RMF also provides support for Remote Mirror and Copy link utilization statistics. This support was delivered by APAR OA04877. PTFs are available for the release of z/OS V1R4 and later.

### Cache reporting

RMF cache reporting and the results of a LISTDATA STATUS command report a cache size that is half the actual size, because the information returned represents only the cluster to which the logical control unit is attached. Each LSS on the cluster reflects the cache and nonvolatile storage (NVS) size of that cluster. z/OS users will find that only the SETCACHE CFW ON | OFF command is supported but other SETCACHE command options. For example, DEVICE, SUBSYSTEM, DFW, NVS, are not accepted.

The cache and NVS size reported by the LISTDATA command is somewhat less than the installed processor memory size. The DS8000 licensed internal code uses part of the processor memory and it is not reported by LISTDATA.

## 12.3.6 System Management Facilities

System Management Facilities (SMF) is a component of IBM Multiple Virtual Storage (IBM MVS™) Enterprise Systems Architecture (ESA) System Product (SP) that collects I/O statistics, provided at the data set and storage class levels. It helps monitor the performance of the direct access storage subsystem.

SMF collects disconnect time statistics that are summarized and reported at the data set level. To support Solid State Drives (SSD), SMF is enhanced to separate DISC time for READ operations from WRITE operations. Here are the two subtypes:

▶ SMF 42 Subtype 6:

This subtype records DASD data set level I/O statistics. I/O response and service time components are recorded in multiples of 128 microseconds for the data set I/O statistics section:

- S42DSRDD is the average disconnect time for reads.
- S42DSRDT is the total number of read operations.

▶ SMF 74 Subtype 5:

The DS8000 provides the ability to obtain cache statistics for every volume in the storage subsystem. These measurements include the count of the number of operations from DASD cache to the back-end storage, the number of random operations, the number of sequential reads and sequential writes, the time to execute those operations, and the number of bytes transferred. These statistics are placed in the SMF 74 subtype 5 record.

For more information, see the Redbooks publication, *MVS System Management Facilities (SMF)*, SA22-7630.

### Migration considerations

The DS8000 is supported as an IBM 2105 for z/OS systems without the DFSMS and z/OS small program enhancements (SPEs) installed. It allows clients to roll the SPE to each system in a sysplex without having to take a sysplex-wide outage. You must perform an IPL to activate the DFSMS and z/OS portions of this support.

### Coexistence considerations

IBM provides support for the DS8000 running in 2105 mode on systems that do not have this SPE installed. The support consists of the recognition of the DS8000 real control unit type and device codes when it runs in 2105 emulation on these down-level systems.

IODF files created by HCD can be shared on systems that do not have this SPE installed. Additionally, you can use existing IODF files that define IBM 2105 control unit records for a 2107 subsystem as long as 16 or fewer logical systems are configured in the DS8000.

## 12.4 Extended Address Volume (EAV) support

As storage facilities tend to expand to larger capacities, the UCB's 64 K limitation is approaching at a rapid rate. Therefore, large volume support must be planned for the needs of installations that require super large volumes, IBM has developed a volume named Extended Address Volume (EAV) recognized by host as 3390 Model A (3390A).

Full support for EAV volume was provided starting from z/Os V1R12. Previous levels can have the same restrictions on sequential (basic, large), partitioned (PDS/PDSE), catalogs, and BDAM data sets in the extended addressing space (EAS).

Support is enhanced to expand volumes to 65,520 cylinders, using existing 16-bit cylinder addressing. It is often referred to as *64 K cylinder volumes*. Components and products that previously shipped with 32,760 cylinders, now also support 65,520 cylinders, for example:

- ▶ Direct access device space management (DADSM) or common volume table of contents (VTOC) access facility (CVAF)
- ▶ DFSMS functional component (DFSMSdss)
- ▶ ICKDSF, a device support facility
- ▶ Data Facility Sort (DFSORT), a high-speed data-processing utility

Checkpoint restart processing now supports a checkpoint data set that resides partially or wholly above the 32,760 cylinder boundary.

With logical volumes (LVs), the VTOC has the potential to grow large. Callers such as DFSMSdss need to read the entire VTOC to find the last allocated data set control block (DSCB). In cases where the VTOC is large, you can experience performance degradation. An interface is implemented to return the highly allocated DSCB on volumes initialized with an indexed VTOC. DFSMSdss uses this interface to limit VTOC searches and to improve performance. The VTOC must be within the first 64 K-1 tracks, but the index can be anywhere on the volume.

With LVs support, it is possible to address a capacity of up to 3.964 PB, where 64 K subchannels multiplied by 55.6 GB per volume, equals to 3.64 PB. To accommodate the needs of installations that require super large volumes, IBM has developed an volume, an extended address volume (EAV) called the *3390 Model A*, shown in Figure 12-2.

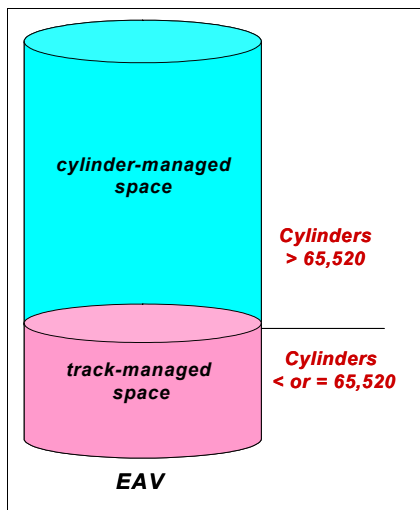


Figure 12-2 Entire new EAV

For the 3390 Model A, support is enhanced to expand the volumes to 1,182,006 cylinders, using 28-bit cylinder addressing, which currently has a limit of 256 M tracks. The existing CCHH track address became CCCcccH x, where the cylinder address is cccCCCC x and the head address is H x, H=0-14.

Starting with LMC level 7.6.30.xx, the DS8000 supports up to Mod 1062 = 1,182,006 cylinders = 1.004+TB.

**Attention:** Copy Services are only allowed for 3390A with maximum size of Mod 236 x 3390 Model 1 = 262,668 cylinders

The partial change from track to cylinder addressing creates two address areas on EAV volumes:

- ▶ Track managed space is the area on an EAV located within the first 65,520 cylinders. Using the 16-bit cylinder addressing enables a theoretical maximum address of 65,535 cylinders. In order to allocate more cylinders, you need to have a new format to address the area of more than 65,520 cylinders. With 16-bit cylinder numbers, the existing track address format is CCCCHHHH:
  - The 16-bit track number is represented with HHHH.
  - The 16-bit track cylinder is represented with CCCC.
- ▶ Cylinder managed space is the area on an EAV located after the first 65,520 cylinders. This space is allocated in multicylinder units (MCU), which have a size of 21 cylinders. To address the extended capacity on an EAV, the new cylinder-track address format for 28-bit cylinder numbers is CCCcCccH:
  - The H represents a 4-bit track number from zero to 14.
  - The high order 12 bits of a 28-bit cylinder number is represented with ccc.
  - The low order 16 bits of a 28-bit cylinder number is represented with CCCC.

Components and products, such as DADSM/CVAF, DFSMSdss, ICKDSF, and DFSORT, also now support 1,182,006 cylinders. Consider the following examples:

- ▶ DS8000 and z/OS limit CKD EAV volume size:
  - For 3390 Model A, one to 1,182,006 cylinders or about 1 TB of addressable storage
  - For 3390 Model A, up to 1062 times the size of 3390 Model 1
- ▶ Configuration granularity:
  - For 1 cylinder boundary sizes, 1 to 56 520 cylinders
  - For 1113 cylinders boundary sizes; 56,763 (51 x 1113) to 1,182,006 (1062 x 1113) cylinders

The size of an existing Model 3/9/A volume can be increased to its maximum supported size using dynamic volume expansion. It can be done with the DS CLI command, as shown in Example 12-1. or by using the web GUI.

*Example 12-1 Dynamically expand CKD volume*

---

```
dsccli> chckdvol -cap 1182006 -capttype cyl -quiet 90f1
Date/Time: 8 September 2010 1:54:02 PM IBM DSCLI Version: 6.6.0.284 DS:
IBM.2107-75TV181
CMUC00022I chckdvol: CKD Volume 90f1 successfully modified.
```

---

**DVE:** Dynamic Volume Expansion can be done as the volume remains online with the host system, and is configured in the following increments:

- ▶ Mod1 (1,113 cylinders) on volumes > than 65,520 cylinders
- ▶ One cylinder on volumes with =< 65,520 cylinders

A VTOC refresh through ICKDSF is a preferred practice, use ICKDSF REFORMAT REFVTOC to rebuild a VTOC index to be full compatible with the new cylinder addressing, even if the volume will remain online showing the new number of cylinders using the DEVSERV command as shown in Example 12-4 on page 258.



**Volume initialization:** Now, with the new function “QUICK INIT,” volume initialization is performed dynamically after volume expansion, with no delay for initialization before going online to the host

The VTOC allocation method for an EAV volume is changed, compared to the VTOC used for LVs. The size of an EAV VTOC index is increased four times, and now has 8192 blocks instead of 2048 blocks. Because no space is left inside the Format 1 DSCB, new DSCB formats, Format 8 and 9 were created to protect existing programs from encountering unexpected track addresses. These DSCBs are called *extended attribute DSCBs*. Format 8 and 9 DSCBs are new for EAV. The existing Format 4 DSCB was changed also to point to the new Format 8 DSCB.

**Tip:** VTOC size should be increased not only for INDEX size. It is possible only if there is free space behind the VTOC location. The VTOC and VTOC index must not have extends.

**Volume size:** When formatting a volume with ICKDSF where the VTOC index size was omitted, the volume takes the default size of 15 tracks.

## 12.4.1 Identifying an EAV

When a volume has more than 65,520 cylinders, the modified Format 4 DSCB is updated to x'FFFE', which has a size of 65,534 cylinders. It will identify the volume as being an EAV. An easy way to identify an EAV is to list the VTOC Summary in Time Sharing Option (TSO) of the Interactive System Productivity Facility (ISPF), version 3.4. Example 12-2 shows the VTOC summary of a 3390 Model 3 volume.

Example 12-2 VTOC information for 3390-3

```

+----- VTOC Summary Information -----+
| Volume . : RE77BE
| Command ==>
|
| Unit . . : 3390
|
| Volume Data          VTOC Data          Free Space   Tracks   Cyls
| Tracks . . : 16,695   Tracks . . :      150   Size . . : 16,530   1,102
| %Used . . :         0   %Used . . :         1   Largest . . : 16,530   1,102
| Trks/Cyls:         15   Free DSCBs:   7,497   Free
|                                     Extents . . :         1
|
| F1=Help   F2=Split   F3=Exit   F9=Swap   F12=Cancel
+-----+

```

Example 12-3 shows a typical VTOC summary for a 3390 Model A EAV volume. It is divided into two parts: the new *Tracks managed*, and the *Total* space.

Example 12-3 VTOC information for 3390-A 1TB

```

+----- VTOC Summary Information -----+
| Volume . : RE77BF
| Command ==>
|
| Unit . . : 3390
|
| Free Space

```

VTOC Data		Total	Tracks	Cyls
Tracks . . :	150	Size . . :	17,729,925	<b>1,181,995</b>
%Used . . :	1	Largest . . :	16,747,290	1,116,486
Free DSCBS:	7,497	Free		
		Extents . . :	2	
Volume Data		Track Managed	Tracks	Cyls
Tracks . . :	17,730,090	Size . . :	982,635	65,509
%Used . . :	0	Largest . . :	982,635	65,509
Trks/Cyls:	15	Free		
F1=Help	F2=Split	F3=Exit	F9=Swap	F12=Cancel

+-----

---

The DEVSERV command can also be used to identify an EAV volume. Example 12-4 shows the new value for the DTYPE field by running the DEVSERV PATHS command.

*Example 12-4 DEVSERV command for 3390 mod3 and 3390 mod A 1TB*

```
ds p,77be,2
IEE459I 13.00.47 DEVSERV PATHS 646
UNIT  DTYPE  M CNT VOLSER  CHPID=PATH STATUS
      RTYPE  SSID CFW TC   DFW PIN DC-STATE CCA DDC      CYL CU-TYPE
077BE,33903 ,0,000,RE77BE,80=+ 81=+ 85=+ 86=+ 82=+ 83=- 84=+ 87=+
      2107  7701  Y  YY.  YY.  N SIMPLEX  BE  BE      1113 2107
077BF,3390A ,0,000,RE77BF,80=+ 81=+ 85=+ 86=+ 82=+ 83=- 84=+ 87=+
      2107  7701  Y  YY.  YY.  N SIMPLEX  BF  BF      1182006 2107
***** SYMBOL DEFINITIONS *****
O = ONLINE          + = PATH AVAILABLE
- = LOGICALLY OFF, PHYSICALLY OFF
```

### Data set type dependencies on an EAV

For EAV, all Virtual Storage Access Method (VSAM) data set types are eligible to be placed on the extended addressing space, or cylinder managed space, of an EAV volume running on z/OS V1.11. This includes all VSAM data types, such as:

- ▶ Key-sequenced data set
- ▶ Relative record data set
- ▶ Entry-sequenced data set
- ▶ Linear data set
- ▶ DB2, IBM IMS™, IBM CICS®, and IBM zSeries® file system (zFS) data sets

The VSAM data sets placed on an EAV volume can either be storage management subsystem (SMS) or non-SMS managed.

Starting from z/OS V1.12 also following Data set types are now supported: sequential (basic, large), partitioned (PDS/PDSE), Catalogs, and BDAM data sets in the extended addressing space (EAS).

### EAV and data set placement dependencies

The EAV volume can be theoretically divided into two parts:

- ▶ The track managed space from cylinder number 1 to cylinder number 65,520
- ▶ The cylinder managed space from number 65,521 to 1,182,006 Cylinders

The EAV track managed area supports all type of data sets. The EAV cylinder managed area has restrictions, which are explained in “Data set type dependencies on an EAV”.

For example, if you allocate a 3390 Model 9 with 65,520 cylinders and place sequential data sets on it (Figure 12-3) perform a dynamic volume expansion, because of a volume full condition, to make the volume into an EAV as 3390 Model A. The required VTOC reformat is successfully performed.

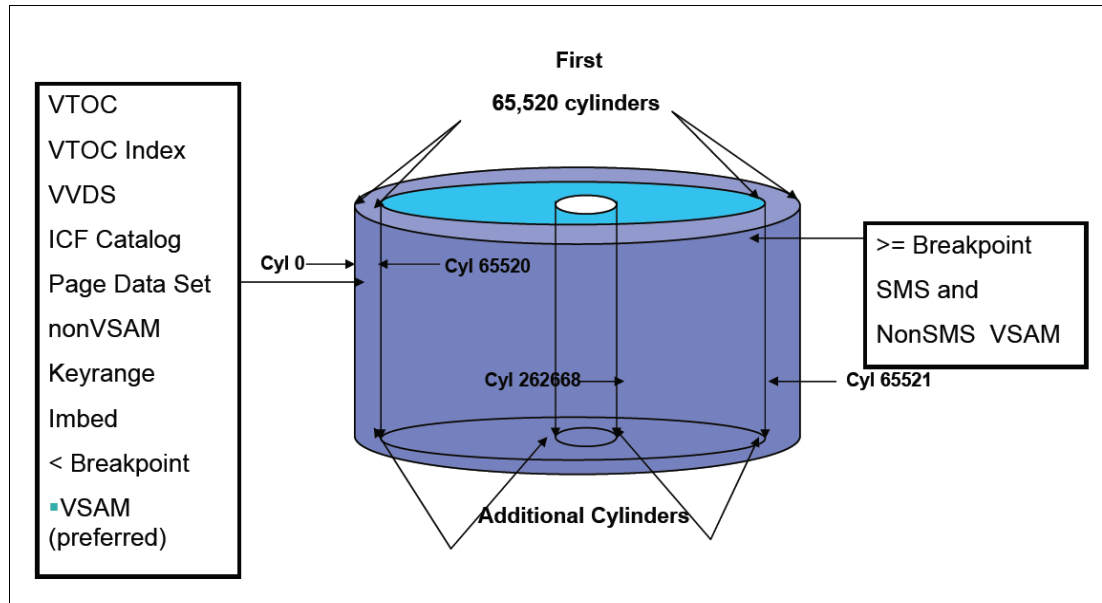


Figure 12-3 Data set placement on EAV

For software, trying to allocate an extent of sequential data sets will fail and produce the error IEF257I - Space requested not available, even if there is a surplus of space physically available in the cylinder managed area. Data sets with extents on the cylinder managed area are not supported with z/OS V1.10. Placing any kind of VSAM data set on the same volume will not cause allocations errors.

## 12.4.2 z/OS prerequisites for EAV

EAV volumes have the following prerequisites:

- ▶ EAV volumes are only supported on z/OS V1.10 and later. If you try to bring an EAV volume online for a system with a release older than z/OS V1.10, the EAV volume does not come online.
- ▶ The 3390 Model A definitions have no additional HCD considerations.
- ▶ On parameter library (parmlib) member IGDSMSxx, the parameter USEEAV(YES) must be set to allow data set allocations on EAV volumes. The default value is NO and prevents allocating data sets to an EAV volume. Example 12-5 shows a sample message that you receive when trying to allocate a data set on EAV volume and USEEAV(NO) is set.

Example 12-5 Message IEF021I with USEEVA set to NO

---

```
IEF021I TEAM142 STEP1 DD1 EXTENDED ADDRESS VOLUME USE PREVENTED DUE TO SMS
USEEAV (NO)SPECIFICATION.
```

---

A new parameter called *break point value* (BPV) determines the size of a data set that can be allocated on a cylinder-managed area. The default for the parameter is 10 cylinders and it can be set on PARMLIB member IGDSMSxx, and in the storage group definition. Storage group BPV overrides system-level BPV. The BPV value can be 0 - 65520. The 0 value means that the cylinder-managed area is always preferred, and the 65520 value means that a track-managed area is always preferred.

**Prerequisite:** Before implementing EAV volumes, apply the latest maintenance, z/OS V1.10 and higher levels coexisting maintenance levels.

### Space considerations for data set allocation

For data set allocation and space requirements, keep in mind the following considerations:

- ▶ When allocating a data set extent on an EAV volume, the space requested is rounded up to the next MCU, if the entire extent is allocated in cylinder-managed space. Individual extents always start and end on an MCU boundary.
- ▶ A given extent is contained in a single managed space, which means an extent cannot straddle where the cylinder-managed area begins.
- ▶ Exact space will be obtained if allocated on a track-managed area, exactly how it was before EAV implementation.
- ▶ If the requested space is not available from the preferred managed space, as determined by BPV, the system can allocate the space from both cylinder-managed and track-managed spaces.
- ▶ Because VSAM striped data sets require that all the stripes be the same size, the system will attempt to process exact requested space using a combination of cylinder-managed and track-managed spaces.

### VSAM control area considerations

For VSAM control area (CA), keep in mind the following considerations:

- ▶ VSAM data sets created in z/OS V1.10 for EAV and non-EAV, might have separate CA sizes from what was received in prior releases. The reason for it is that a CA must be compatible with an MCU for a cylinder-managed area.
- ▶ VSAM data sets allocated with compatible CAs on a non-EAV are eligible to be extended to additional volumes that support cylinder-managed space. VSAM data sets physically copied from a non-EAV to an EAV might have an incompatible CA and might not be eligible for EAV. It means that extents for additional space do not use cylinder-managed space.

**Migration consideration:** This consideration is for a migration to z/OS V1.10.

## 12.4.3 EAV migration considerations

For EAV migration, keep in mind the following considerations:

- ▶ Assistance for migration will be provided through the Application Migration Assistance Tracker. For more details about Assistance Tracker, see APAR II13752, at this website: <http://www.ibm.com/support/docview.wss?uid=isg1II13752>
- ▶ Actions advised:
  - Review your programs and take a look at the calls for the macros OBTAIN, REALLOC, CVAFDIR, CVAFSEQ, CVAFDSM, and CVAFFILT. These macros were changed and you need to update your program to reflect those changes.

- Look for programs that calculate volume or data set size by any means, including reading a VTOC or VTOC index directly with a basic sequential access method or execute channel program (EXCP) data control block (DCB). This task is important because now you will have new values returning for the volume size.
  - Review your programs and look for EXCP and Start I/O (STARTIO) macros for DASD channel programs and other programs that examine DASD channel programs or track addresses. Now that you have the new addressing mode, your programs must be updated.
  - Look for programs that examine any of the many operator messages that contain a DASD track, block address, data set, or volume size. The messages now show new values.
- ▶ Migrating data:
- Define new EAVs by creating them on the DS8000 or expanding existing volumes using dynamic volume expansion.
  - Add new EAV volumes to storage groups and storage pools, and update automatic class selection (ACS) routines.
  - Copy data at the volume level with either IBM TDMF®, DFSMSdss, Peer-to-Peer Remote Copy (PPRC), DFSMS, Copy Services Global Mirror, Metro Mirror, Global Copy, or FlashCopy.
  - Copy data at the data set level with SMS attrition, zDMF - z/OS Dataset Mobility Facility, DFSMSdss, and DFSMSHsm.

Good volume candidates for EAV include DB2, zFS, CICS VSAM, record-level sharing (RLS), and Information Management System (IMS) VSAM. Poor EAV candidates are DFSMSHsm migration level 1, Backup, or migration level 2, although small data set packaging is eligible for cylinder managed space. Other poor EAV candidates are work volumes, TSO, generation data group, batch, load libraries, and system volumes.

DFSMSdss and DFSMSHsm considerations when migrating to EAV are described in the Redbooks publication, *DFSMS V1.10 and EAV Technical Guide*, SG24-7617.

## 12.5 FICON specifics for a z/OS environment

With FC adapters that are configured for FICON, the DS8000 series provides the following configuration capabilities:

- ▶ Either fabric or point-to-point topologies
- ▶ A maximum of 128 host adapter ports, depending on the DS8800 processor feature
- ▶ A maximum of 509 logins per Fibre Channel port
- ▶ A maximum of 8192 logins per storage unit
- ▶ A maximum of 1280 logical paths on each Fibre Channel port
- ▶ Access to all control-unit images over each FICON port
- ▶ A maximum of 512 logical paths per control unit image

### 12.5.1 Overview

FICON host channels limit the number of devices per channel to 16,384. To fully access 65,280 devices on a storage unit, it is necessary to connect a minimum of four FICON host channels to the storage unit. This way, by using a switched configuration, you can expose 64 control-unit images (16,384 devices) to each host channel.

## Quick Init for z/OS

Starting from release 6.2 (LMC level 7.6.20.xx), volume initialization is performed dynamically after volume creation or expansion. There is no delay for initialization before going online to the host. The customer can initiate copy services operations immediately after volume creation or expansion without needing to wait for full volume initialization. It addresses special requirements of accounts that re-provision their volumes regularly.

## Multiple Allegiance

Normally, if any System z host image (server or LPAR) does an I/O request to a device address for which the storage disk subsystem is already processing an I/O that came from another System z host image, the storage disk subsystem will send back a *device busy* indication. This delays the new request and adds to the overall response time of the I/O; this delay is shown in the Device Busy Delay (AVG DB DLY) column in the RMF DASD Activity Report. Device Busy Delay is part of the Pend time.

The DS8000 series accepts multiple I/O requests from different hosts to the same device address, increasing parallelism and reducing channel impact. In older storage disk systems, a device had an implicit allegiance, that is, a relationship created in the control unit between the device and a channel path group when an I/O operation is accepted by the device. The allegiance causes the control unit to guarantee access (no busy status presented) to the device for the remainder of the channel program over the set of paths associated with the allegiance.

With Multiple Allegiance, the I/O requests are accepted by the DS8000 and all requests are processed in parallel, unless there is a conflict when writing to the same data portion of the CKD logical volume.

In systems without Multiple Allegiance, all except the first I/O request to a shared volume are rejected, and the I/Os are queued in the System z channel subsystem, showing up in Device Busy Delay and PEND time in the RMF DASD Activity reports. However, a device busy condition can still happen. It will occur when an active I/O is writing a certain data portion on the volume and another I/O request comes in and tries to either read or write to that same data. To ensure data integrity, those subsequent I/Os will get a busy condition until that previous I/O is finished with the write operation.

Multiple Allegiance provides significant benefits for environments running a sysplex, or System z systems sharing access to data volumes. Multiple Allegiance and PAV (Parallel Access Volumes) can operate together to handle multiple requests from multiple hosts.

## 12.5.2 Parallel access volumes (PAV) definition

For EAV volumes, do not consider using static Parallel Access Volumes (PAV) for future planning. Instead, for performance reasons, use HyperPAV and you will be able to increase the amount of real device addresses within an LCU.

Two base concepts are basic in PAV functionality:

- ▶ **Base address:** The base device address in the conventional unit address of a logical volume. There is only one base address associated with any volume.
- ▶ **Alias address:** An alias device address is mapped to a base address. I/O operations to an alias run against the associate's base address storage space. There is no physical space associated with an alias address; you can define more than one alias per base.

Different usages of these aliases are implemented in different PAV typologies:

- ▶ Static PAV means that the hosts can use only the aliases that were defined for that base address, so if you need more parallel accesses to some devices, a new HCD should be done to implement your new needs.
- ▶ Dynamic PAV, using Work Load Manager (WLM) the aliases are dynamically moved between different base addresses as requested by host jobs, WLM also keeps track of the devices utilized by the different workloads, accumulates this information over time, and broadcasts it to the other systems in the same sysplex. If WLM determines that any workload is not meeting its goal due to IOS queue (IOSQ) time, WLM will attempt to find an alias device that can be reallocated to help this workload achieve its goal
- ▶ HyperPAV, Dynamic PAV requires the WLM to monitor the workload and goals. It takes some time until the WLM detects an I/O bottleneck. Then the WLM must coordinate the reassignment of alias addresses within the sysplex and the DS8000. All of this takes time, and if the workload is fluctuating or has a burst character, the job that caused the overload of one volume could have ended before the WLM had reacted. In these cases, the IOSQ time was not eliminated completely.

With HyperPAV, the WLM is no longer involved in managing alias addresses. For each I/O, an alias address can be picked from a pool of alias addresses within the same LCU. This capability also allows different HyperPAV hosts to use one alias to access different bases, which reduces the number of alias addresses required to support a set of bases in an IBM System z environment, with no latency in assigning an alias to a base. This functionality is also designed to enable applications to achieve better performance than is possible with the original PAV feature alone, while also using the same or fewer operating system resources.

You can find more information regarding dynamic PAV at the following website:

<http://www.ibm.com/systems/z/os/zos/features/wlm/>

Again, consider HyperPAV over dynamic PAV management, which allows for less alias device addresses and for more real or base device addresses within a LCU.

## RMF considerations

RMF reports all I/O activity against the base PAV address, not by the base and associated aliases. The performance information for the base includes all base and alias activity.

As illustrated in Example 12-6, on the Data Set Delays window in RMF Monitor III, for Volume MLDC65 with device Number DC65, you can see the identification for an EAV volume (3390A) in the Device field.

The presence of PAV: H means that HyperPAV is used.

*Example 12-6 Data Set Delays: Volume window*

```
RMF V1R10 Data Set Delays - Volume
Command ===>                               Scroll ===> CSR

Samples: 120      System: SC70  Date: 05/10/09 Time: 18.04.00 Range: 120  Sec

----- Volume MLDC65 Device Data -----
Number:   DC65      Active:    0%      Pending:   0%      Average Users
Device:   3390A     Connect:  0%      Delay DB:  0%      Delayed
Shared:   Yes       Disconnect: 0%      Delay CM:  0%      0.0
PAV:     1.0H
```

----- Data Set Name ----- Jobname ASID DUSG% DDLY%

No I/O activity detected.

---

### Missing-interrupt handler values considerations

The DS8000 provides a preferred interval of 30 seconds as part of the RCD. The z/OS uses this information to set its missing-interrupt handler (MIH) value.

The MIH times for PAV alias addresses must not be set. An alias device inherits the MIH of the base address to which it is bound and it is not possible to assign an MIH value to an alias address. Alias devices are not known externally and are only known and accessible by IOS. If an external method is used to attempt to set the MIH on an alias device address, an IOS090I message is generated. For example, the following message is generated for each attempt to set the MIH on an alias device:

IOS090I *alias-device-number* IS AN INVALID DEVICE

**Tip:** When setting MIH times in the IECIO5xx member of SYS1.PARMLIB, do not use device ranges that include alias device numbers.

## 12.5.3 HyperPAV z/OS support and implementation

The DS8000 series users can benefit from enhancements to Parallel Access Volume (PAV) with support for HyperPAV. HyperPAV provides the ability to use an alias address to access any base on the same logical control unit image per I/O base.

### Benefits of HyperPAV

HyperPAV was designed to offer the following capabilities:

- ▶ Provide an even more efficient Parallel Access Volumes (PAV) function.
- ▶ Help clients who implement larger volumes to scale I/O rates without the need for additional PAV alias definitions.
- ▶ Exploit the FICON architecture to reduce impact, improve addressing efficiencies, and provide storage capacity and performance improvements:
  - More dynamic assignment of PAV aliases improves efficiency.
  - The number of PAV aliases needed can be reduced, taking fewer from the 64 K device limitation and leaving more storage for capacity use.
- ▶ Enable a more dynamic response to changing workloads
- ▶ Provide simplified management of aliases
- ▶ Make it easier for users to make a decision to migrate to larger volume sizes

In this section, you see the commands and options that you can use for setup and control of HyperPAV and for the display of HyperPAV status information. The migration to HyperPAV and the system requirements for HyperPAV are provided in this section.



## HyperPAV options

The following SYS1.PARMLIB IECIOSxx options enable HyperPAV at the LPAR level:

**HYPERPAV= YES | NO | BASEONLY**

HyperPAV has the following options and corresponding actions:

<b>YES</b>	Attempt to initialize LSSs in HyperPAV mode.
<b>NO</b>	Do not initialize LSSs in HyperPAV mode.
<b>BASEONLY</b>	Attempt to initialize LSSs in HyperPAV mode, but only start I/Os on base volumes.

The BASEONLY option returns the LSSs with enabled HyperPAV capability to a pre-PAV behavior for this LPAR.

## HyperPAV migration

You can enable HyperPAV dynamically. Because it can take time to initialize all needed LSSs in a DS8000 into HyperPAV mode, planning is important. If many LSSs are involved, then pick a concurrent maintenance window, with low I/O activities, to perform the SETIOS HYPERPAV=YES command and do not schedule concurrent DS8000 microcode changes or IODF activation together with this change.

Example 12-7 shows a command example to dynamically activate HyperPAV. This activation process might take time to complete on all attached disk storage subsystems that support HyperPAV. Verify that HyperPAV is active through a subsequent display command, as shown in Example 12-7.

*Example 12-7 Activate HyperPAV dynamically*

---

```
SETIOS HYPERPAV=YES
```

```
IOS189I HYPERPAV MODE CHANGE INITIATED - CONTROL UNIT CONVERSION WILL  
COMPLETE ASYNCHRONOUSLY
```

```
D IOS, HYPERPAV  
IOS098I 15.55.06 HYPERPAV DATA 457  
HYPERPAV MODE IS SET TO YES
```

---

If you are currently using PAV and FICON, then no HCD or DS8000 logical configuration changes are needed on the existing LSSs.

To stage HyperPAV development, follow these steps:

1. Load or authorize the HyperPAV feature on the DS8000. If necessary, you can run without exploiting this feature by using the z/OS PARMLIB option.
2. Enable the HyperPAV feature on z/OS images that you want to use HyperPAV, using the PARMLIB option or the **SETIOS** command.
3. Enable the HyperPAV feature on all z/OS images in the sysplex and authorize the licensed function on all attached DS8000s.
4. Optional: Reduce the number of aliases defined.

Full coexistence with traditional PAVs, such as static PAV or dynamic PAV, and sharing with z/OS images without HyperPAV enabled, helps migration to HyperPAV to be a flexible procedure.

## HyperPAV definition

The correct number of aliases for your workload can be determined from the analysis of RMF data. The PAV analysis tool, which can be used to analyze PAV usage, is available at the following website:

<http://www.ibm.com/servers/eserver/zseries/zos/unix/bpxalty2.html#pavanalysis>

## HyperPAV commands for setup, control, and status display

Use the following commands for HyperPAV management and status information:

- ▶ SETIOS HYPERPAV= YES | NO | BASEONLY
- ▶ SET IOS=xx
- ▶ D M=DEV
- ▶ D IOS, HYPERPAV
- ▶ DEVSERV QPAV, dddd

In Example 12-8, the `d m=dev` command provides system configuration information for the base address 0710 that belongs to an LSS with enabled HyperPAV.

*Example 12-8 Display information for a base address in an LSS with enabled HyperPAV*

---

```
SY1  d m=dev(0710)
SY1  IEE174I 23.35.49 DISPLAY M 835
DEVICE 0710  STATUS=ONLINE
CHP                10  20  30  40
DEST LINK ADDRESS  10  20  30  40
PATH ONLINE       Y   Y   Y   Y
CHP PHYSICALLY ONLINE Y   Y   Y   Y
PATH OPERATIONAL  Y   Y   Y   Y
MANAGED           N   N   N   N
CU NUMBER         0700 0700 0700 0700
MAXIMUM MANAGED CHPID(S) ALLOWED:  0
DESTINATION CU LOGICAL ADDRESS = 07
SCP CU ND         = 002107.000.IBM.TC.03069A000007.00FF
SCP TOKEN NED     = 002107.900.IBM.TC.03069A000007.0700
SCP DEVICE NED    = 002107.900.IBM.TC.03069A000007.0710
HYPERPAV ALIASES IN POOL  4
```

---

In Example 12-9, address 0718 is an alias address belonging to a HyperPAV LSS. If you see a HyperPAV alias in use or bound, it is displayed as bound.

*Example 12-9 Display information for an alias address belonging to a HyperPAV LSS*

---

```
SY1  D M=DEV(0718)
SY1  IEE174I 23.39.07 DISPLAY M 838
DEVICE 0718  STATUS=POOLED HYPERPAV ALIAS
```

---

The **D M=DEV** command in Example 12-10 shows HyperPAV aliases (HA).

*Example 12-10 The system configuration information shows the HyperPAV aliases*

---

```

SY1  d m=dev
SY1  IEE174I 23.42.09 DISPLAY M 844
DEVICE STATUS: NUMBER OF ONLINE CHANNEL PATHS
      0 1 2 3 4 5 6 7 8 9 A B C D E F
000 DN 4  DN DN DN DN DN DN DN .  DN DN 1 1 1 1
018 DN DN DN DN 4  DN DN DN DN DN DN DN DN DN DN
02E 4  DN 4  DN 4  8  4  4  4  4  4  4  4  DN 4  DN
02F DN 4  4  4  4  4  4  DN 4  4  4  4  4  DN DN 4
030 8  .  .  .  .  .  .  .  .  .  .  .  .  .  .
033 4  .  .  .  .  .  .  .  .  .  .  .  .  .  .
034 4  4  4  4  DN DN DN DN DN DN DN DN DN DN DN DN
03E 1  DN DN DN DN DN DN DN DN DN DN DN DN DN DN DN
041 4  4  4  4  4  4  4  4  AL AL AL AL AL AL AL AL
048 4  4  DN DN DN DN DN DN DN DN DN DN DN DN DN 4
051 4  4  4  4  4  4  4  4  UL UL UL UL UL UL UL UL
061 4  4  4  4  4  4  4  4  AL AL AL AL AL AL AL AL
071 4  4  4  4  DN DN DN DN HA HA DN DN .  .  .
073 DN DN DN .  DN .  DN .  DN .  DN .  HA .  HA .
098 4  4  4  4  DN 8  4  4  4  4  4  DN 4  4  4  4
0E0 DN DN 1  DN DN DN DN DN DN DN DN DN DN DN DN DN
OF1 1  DN DN DN DN DN DN DN DN DN DN DN DN DN DN DN
FFF .  .  .  .  .  .  .  .  .  .  .  .  .  HA HA HA HA
***** SYMBOL EXPLANATIONS *****
@ ONLINE, PHYSICALLY ONLINE, AND OPERATIONAL INDICATORS ARE NOT EQUAL
+ ONLINE                # DEVICE OFFLINE                . DOES NOT EXIST
BX DEVICE IS BOXED      SN SUBCHANNEL NOT AVAILABLE
DN DEVICE NOT AVAILABLE PE SUBCHANNEL IN PERMANENT ERROR
AL DEVICE IS AN ALIAS   UL DEVICE IS AN UNBOUND ALIAS
HA DEVICE IS A HYPERPAV ALIAS

```

---

## HyperPAV system requirements

HyperPAV has the following z/OS requirements:

- ▶ HyperPAV is supported starting with z/OS V1.8.
- ▶ For prior levels of the operating system, the support is provided as an SPE back to z/OS V1.6, as follows:
  - IOS support, APAR OA13915
  - DFSMS support, including DFSMS, SMS, the always-on management (AOM) command, and the DEVSERV command with APARs OA13928, OA13929, OA14002, OA14005, OA17605, and OA17746
  - WLM support, APAR OA12699
  - GRS support, APAR OA14556
  - ASM support, APAR OA14248
- ▶ RMF, APAR OA12865

In addition, DS8000 storage systems require the following licensed functions:

- ▶ FICON attachment
- ▶ PAV
- ▶ HyperPAV

**Important:** Make sure to have the latest z/OS APAR: OA39087 (Media Manager). QSAM/BSAM use of zHPF must remain or be disabled until fixes for OA38939 and OA39130 (and prerequisites OA38966 and OA38925), prior to installing microcode at release 6.2 or higher.

For more details visit the link:

<http://www.ibm.com/support/docview.wss?uid=ssg1S1004024>

## 12.5.4 Extended Distance FICON

For the DS8000, Extended Distance FICON, also known as *Simplified FICON Channel Extension*, is an enhancement to the FICON architecture. It can eliminate performance degradation at extended distances, having no channel extender installed, by implementing a new information unit (IU) pacing protocol. You can use a less complex and cheaper channel extender, which only performs frame forwarding, rather than a channel extender that dissects every channel control word (CCW) to optimize the transfer through the channel extender to get the best performance. These are typically channel extenders that have extended remote copy (XRC) emulation running on them.

An enhancement was implemented on the standard FICON architecture, Fibre Channel single byte command set 3 (FC-SB-3), layer with a new protocol for *persistent* IU pacing. It was achieved for the DS8000 z/OS Global Mirror XRC, SDM read record set (RRS) data transfer from a DS8000 to the SDM host address space. The control units that support the extended distance FICON feature can increase the pacing count. The pacing count is the number of IUs that are *inflight* from a channel to a control unit.

### IU pacing

Standard FICON supports IUs pacing of 16 IUs in flight. Extended Distance FICON now extends the IU pacing for the RRS CCW chain to permit 255 IUs inflight without waiting for an acknowledgement from the control unit, eliminating engagement between the channel and control unit. This support allows the channel to remember the last pacing information and use this information for subsequent operations to avoid performance degradation at the start of a new I/O operation.

Extended Distance FICON reduces the need for channel extenders in DS8000 series 2-site and 3-site z/OS Global Mirror configurations because of the increased number of read commands simultaneously inflight. It provides greater throughput over distance for IBM z/OS Global Mirror (XRC) using the same distance.

### Summary

The Extended Distance FICON channel support produces performance results similar to XRC emulation mode at long distances. It can be used to enable a wider range of choices in channel extension technology, when using z/OS Global Mirroring, because emulation in the extender might not be required. It can help reduce the total cost of ownership and provides comparable performance when mirroring application updates over long distances. For more information, see the Redbooks publication, *DS8000 Copy Services for IBM System z*, SG24-6787.

## 12.5.5 High Performance FICON for System z with multitrack support (zHPF)

To help increase system performance and provide greater System z FICON channel efficiency, the DS8000 provides support for High Performance FICON for System z (zHPF) with multitrack. The zHPF system, with multitrack operations, is an optional licensed feature on DS8000 storage systems.

zHPF is supported starting from DS8000 licensed machine code 5.4.30.248 DS8000 bundle level 64.30.78.0 or higher.

With the introduction of zHPF, the FICON architecture was streamlined by removing significant impact from the storage subsystem and the microprocessor within the FICON channel. A command block is created to chain commands into significantly fewer sequences. The effort required to convert individual commands into FICON format is removed as multiple System z I/O commands are packaged together and passed directly over the fiber optic link.

The zHPF system provides an enhanced FICON protocol and system I/O architecture that results in improvements for small block transfers, with a track or less, to disk using the device independent random access method.

In situations where it is the exclusive access in use, it can help improve FICON I/O throughput on a single DS8000 port by 100%. Realistic workloads with a mix of data set transfer sizes can see 30-70% of FICON I/Os utilizing zHPF, with potential results of up to a 10-30% channel utilization savings.

Although clients can see I/Os complete faster as the result of implementing zHPF, the real benefit is expected to be obtained by using fewer channels to support existing disk volumes, or increasing the number of disk volumes supported by existing channels.

Additionally, the changes in architectures offer end-to-end system enhancements to improve reliability, availability, and serviceability.

The zHPF system uses multitrack operations to allow reading or writing of more than a track's worth of data by a single transport mode operation. DB2, VSAM, zFS, the hierarchical file system, partitioned data set extended (PDSE), striped extended format data sets, and other applications that use large data transfers are expected to benefit from zHPF multitrack function.

In addition, zHPF complements the EAV for System z strategy for growth by increasing the I/O rate capability as the volume sizes expand vertically.

In laboratory measurements, multitrack operations, for example, reading 16 x 4 KB per I/O, converted to the zHPF protocol on a FICON Express8 channel achieved a maximum of up to 40% more MBps than multitrack operations using the native FICON protocol.

IBM laboratory testing and measurements are available at the following website:

[http://www.ibm.com/systems/z/hardware/connectivity/FICON\\_performance.html](http://www.ibm.com/systems/z/hardware/connectivity/FICON_performance.html)

The zHPF, with multitrack operations, is available on z/OS V1.9 or higher, with the PTFs for APARs OA26084 and OA29017.

Currently, zHPF and support for multitrack operations is available for the IBM System z10®, z196 and z114 servers and applies to all FICON Express8S (available only on z196 and z114), FICON Express8, FICON Express4, and FICON Express2 features, CHPID type FC.

The FICON Express adapters are *not* supported.

The zHPF is transparent to applications, however, z/OS configuration changes are required, for example, HCD must have CHPID type FC defined for all the CHPIDs that are defined to the DS8000 control unit.

For the DS8000, installation of the licensed feature key for the High Performance FICON feature is required. When these items are addressed, existing FICON port definitions in the DS8000 will function in either native FICON or zHPF protocols in response to the type of request being performed. These are nondisruptive changes.

For z/OS, after the PTFs are installed in the LPAR, you must then set ZHPF=YES in IECIOSxx in SYS1.PARMLIB or issue the SETIOS ZHPF=YES command. The default setting is ZHPF=NO. Use the ZHPF=YES setting after the required configuration changes and prerequisites are met. For more information about zHPF, see this website:

<http://www.ibm.com/support/techdocs/atsmastr.nsf/fe582a1e48331b5585256de50062ae1c/05d19b2a9bd95d4e8625754c0007d365?OpenDocument>

## 12.5.6 zHPF latest enhancements

Starting with release level 6.2 (LMC code level 7.6.20.xx), DS8000 supports a wide range of new enhancements specific for zHPF channels. The following sections provide a list of the enhancements with a brief description of each.

### Enhancements for QSAM/BSAM

The typical command chain in a command information unit is as follows

- ▶ Prefix - Read... Read - Locate Record - Read... Read
- ▶ Prefix - Write... Write - Locate Record - Write... Write

Original zHPF supported the following read and write commands:

- ▶ Read Track Data
- ▶ Write Track Data

zHPF command support is expanded to include the following options:

- ▶ Format Write
- ▶ Update Key
- ▶ Update Data
- ▶ Read Count
- ▶ Read Data
- ▶ Read Key and Data
- ▶ Read Count, Key and Data

Supported on z/OS V1R11, V1R12, V1R13, and above

### zHPF incorrect transfer length support

Read and write commands indicate a transfer length and the control unit returns a residual count on the last command in the command chain (zero if all data set)

- ▶ Original zHPF delivery (z10) did not store any data in memory on reads if residual transfer count was non-zero.
- ▶ Newer zHPF channels indicate they have zHPF incorrect length support in I/O port login (PERLI)
- ▶ If supported, control unit presents non-zero residual counts without presenting unit check
- ▶ Needed for QSAM and BSAM.
- ▶ Supported on z/OS V1R11, V1R12, V1R13, and above

## zHPF bi-directional data transfer and TCA extension

Original zHPF supported commands transferred limited to about 240 bytes in a command information unit.

Now zHPF support for TCA extension allows more commands per I/O operation:

- ▶ Allows up to about 1000 bytes for commands in beginning of data transfer.
- ▶ Allows for long command chains to be issued in a single command.
- ▶ zHPF channel supports bi-directional data transfer to allow read data after writing TCA extension. See Figure 12-4.

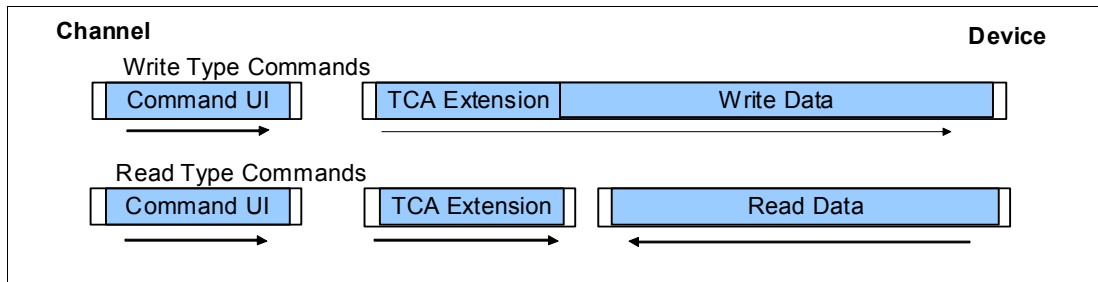


Figure 12-4 zHPF bi-directional data transfer example

- ▶ Needed by DB2, which reads up to 32 pages per I/O operation.
- ▶ Supported on z/OS V1R11, V1R12, V1R13, and above

## zHPF Pre-fetch Optimizer (LPO)

In addition to zHPF list prefetch, R6.2 introduced a new caching algorithm called List Pre-fetch Optimizer (LPO). List Prefetch Optimizer requires zHPF. Unlike zHPF list prefetch, LPO is internal to the DS8000 code. Whereas the objective of zHPF list prefetch is to reduce the I/O connect time, the objective of LPO is to reduce disconnect time. Release level 6.3 went further by fine tuning LPO for some types of list prefetch I/O on zHPF, Prefix and locate commands can provide a list of records (CCHHR) to be read by the next locate operation in the command chain.

- ▶ zHPF allows the control unit to anticipate the next set of records while the host is reading the current set of records.
- ▶ zHPF list of up to 21 records per command.
- ▶ Supported on z/OS V1R11, V1R12, V1R13, and above
- ▶ For more information, see the Redpaper publication, *DB2 for z/OS and List Prefetch Optimizer*, REDP-4682.

## CKD Event Aggregation Summary

Starting from Release 6.2 (LMC level 7.6.20.xx) DS8000 has implemented the CKD Event Aggregation Summary, which works as follows:

- ▶ Reduces host interrupts by up to 1: 256:  
Now the z/OS host receives a single notification of each event that affects all volumes in an LSS.
- ▶ Summary event notification for PPRC suspension events:  
Only if the host supports PPRC suspension summary events, each LSS with a suspended volume presents an unsolicited unit check to each hot spot that is grouped on the LSS indicating PPRC suspension.  
Software initiates a task to issue two commands to get PPRC state of volume on LSS and reset summary event state.

First volume PPRC suspension event on LSS is reported immediately to all grouped hosts.

- ▶ If all hosts accept an event, the next event notification waits at least 150 ms.
- ▶ In case some host does not accept the event in 5 seconds, the state change status on each affected volume is presented:
  - New storage controller health attention message for new volume events
  - Event code, severity code, and LSS device bit map in each message
  - IBM GDPS® uses message severity code to initiate PPRC failover.

CKD Aggregation Summary is supported on z/OS starting from level V1R13.

Figure 12-5 shows an example of the difference of an error that occurred with and without Error Aggregation. As you can see, the difference is conspicuous not only with the total amount of errors reported, but also in how the errors are managed.

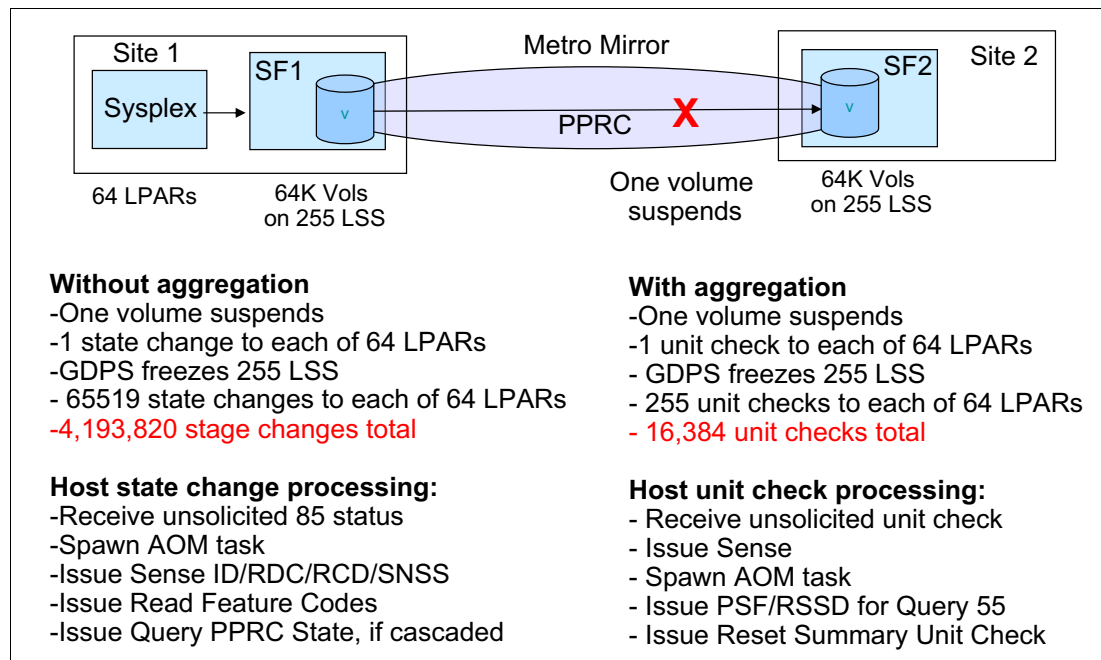


Figure 12-5 PPRC error event comparison



## 12.6 z/VM considerations

This section provides specific considerations that are relevant when attaching a DS8000 series to a z/VM environment.

### 12.6.1 Connectivity

The z/VM operating system provides the following connectivity:

- ▶ z/VM supports FICON attachment as 3990 Model 3 or 6 controller.
- ▶ Native controller modes 2105 and 2107 are supported on z/VM V5.4. The z/VM operating system also simulates controller mode support by each guest.
- ▶ z/VM supports FCP attachment for Linux systems running as a guest.
- ▶ z/VM supports FCP-attached SCSI disks starting with z/VM V5.4 and emulated FBA devices.

### 12.6.2 Supported DASD types and LUNs

The z/VM operating system supports the following extended count key data (IBM ECKD™) DASD types:

- ▶ 3390 Models 2, 3, and 9, including the 32,760 and 65,520 cylinder custom volumes.
- ▶ 3390 Model 2 and 3 in 3380 track compatibility mode.
- ▶ 3390 Model A volumes are not supported at this time. When running z/OS as a virtual machine guest, EAV volumes can be used if they are directly attached to the z/OS guest.

The z/VM operating system also provides the following support when using an FCP attachment:

- ▶ FCP-attached SCSI LUNs as emulated 9336 Model 20 DASD
- ▶ 1 TB SCSI LUNs

### 12.6.3 PAV and HyperPAV z/VM support

The z/VM operating system provides PAV support. This section provides basic support information for implementing PAV in a z/VM environment.

You can find additional z/VM technical information for PAV support at this website:

<http://www.vm.ibm.com/storman/pav/>

#### **z/VM guest support for dedicated PAVs**

The z/VM operating system enables a guest z/OS to use PAV and dynamic PAV tuning as dedicated volumes. The following considerations apply:

- ▶ Alias and base addresses must be attached to the z/OS guest. You need a separate ATTACH command for each alias address. Attach the base address and its aliases address to the same guest.
- ▶ A base address cannot be attached to a system, if one of its alias addresses is attached to that guest. It means that you cannot use PAVs for fullpack minidisks. The QUERY PAV command is available for authorized, class B, users to query base and alias addresses with QUERY PAV rdev and QUERY PAV ALL.

- ▶ To verify that PAV aliases are bound to the correct bases, use the QUERY CHPID xx command combined with QUERY PAV rdev-rdev, where xx is the channel path identifier (CHPID) whose device addresses is displayed, showing the addresses and any aliases, and rdev is the real device address.

### **PAV minidisk support with SPE**

Starting with z/VM V5.2.0 with APAR VM63952, z/VM supports PAV minidisks. With SPE, z/VM provides the following advantages:

- ▶ Support of PAV minidisks
- ▶ Workload balancing for guests that do not exploit PAV, such as Conversational Monitor System
- ▶ A real I/O dispatcher queues minidisk I/O across system-attached alias volumes
- ▶ Minidisks that can be linked for guests that exploit PAV, for example, z/OS and Linux
- ▶ The PAVALIAS parameter of the DASDOPT and MINIOPT user directory statements or the CP DEFINE ALIAS command, which creates alias minidisks, both fullpack and non-fullpack, for using guests
- ▶ Dynamic alias to base reassociation is supported for guests that exploit PAV for dedicated volumes and for minidisks under restricted conditions

### **HyperPAV support**

Starting with z/VM Version 5.4, z/VM supports HyperPAV for dedicated DASD and minidisks.

## **12.6.4 Missing-interrupt handler**

The z/VM operating system sets its MIH value to 1.25 multiplied by what the hardware reports. This way, with the DS8000 setting the MIH value to 30 seconds, z/VM is set to a MIH value of approximately 37.5 seconds. It enables the guest to receive the MIH 30 seconds before z/VM does.

## **12.7 VSE/ESA and z/VSE considerations**

The following considerations apply regarding Virtual Storage Extended, Enterprise Systems Architecture (VSE/ESA), and z/VSE support:

- ▶ An APAR is required for VSE 2.7 to exploit large volume support, but support for EAV is still not provided.
- ▶ VSE has a default MIH timer value to 180 seconds. You can change this setting to the suggested DS8000 value of 30 seconds by using the SIR MIH command, which is documented in *Hints and Tips for VSE/ESA 2.7*, at this website:

<http://www.ibm.com/servers/eserver/zseries/zvse/documentation/>

## 12.8 I/O Priority Manager for z/OS

DS8000 I/O Priority Manager is a licensed function feature introduced for IBM System DS8000 storage systems with DS8000 Licensed Machine Code (LMC) R6.1 or higher. Starting from release 6.2 (LMC level 7.6.20.xx), I/O priority is also working on z/OS environments and it interacts with Workload Manager (eWLM). It enables more effective storage consolidation and performance management and the ability to align quality of service (QoS) levels to separate workloads in the system that are competing for the same shared and possibly constrained storage resources.

DS8000 I/O Priority Manager constantly monitors system resources to help applications meet their performance targets automatically, without operator intervention. The DS8000 storage hardware resources that are monitored by the I/O Priority Manager for possible contention are the RAID ranks and device adapters.

Basically, I/O Priority Manager uses QoS to assign priorities for different volumes and applies network QoS principles to storage by using a particular algorithm called Token Bucket Throttling for traffic control. I/O Priority Manager is designed to understand the load on the system and modify it by using dynamic workload control.

### 12.8.1 Performance groups

For System z, there are 14 performance groups: three performance groups for high-performance policies, four performance groups for medium-performance policies, six performance groups for low-performance policies, and one performance group for the default performance policy.

Only with System z, two operation modes are available for I/O Priority Manager: without software support or with software support.

In the Workload Manager (WLM) Service Definition, the service option “I/O priority management” must be set to Yes, to being able to use it. See also Figure 12-6 on page 277 to verify where it can be enabled on DS8000.

Support is provided for CKD I/O priority with software input:

- ▶ The user assigns a performance policy to each CKD volume that applies in the absence of additional software support.
- ▶ The OS can optionally specify parameters that determine priority of each I/O operation.
- ▶ In general, the priorities will be managed by z/OS through eWLM.
- ▶ It is supported on z/OS V1.11, V1.12, V1.13, and above.

I/O Priority Manager is a DS8000 Licensed Function that must be activated before you can start to use it. For details about how to obtain the necessary information for the DSFA website and how to apply the license keys, see the Redbooks publication, *IBM System Storage DS8000: Architecture and Implementation*, SG24-8886,

In the Workload Manager (WLM) Service Definition, the service option “I/O priority management” must be set to “Yes,” in order to be able to use it. See also Figure 12-6 on page 277 to verify where it can be enabled on the DS8000.

Support is provided for CKD I/O priority with software input:

- ▶ The user assigns a performance policy to each CKD volume that applies in the absence of additional software support.
- ▶ The OS can optionally specify parameters that determine priority of each I/O operation.
- ▶ In general, the priorities will be managed by z/OS through eWLM.
- ▶ It is supported on z/OS V1.11, V1.12, V1.13, and above.

An SMNP trap is automatically generated when a rank enters saturation:

- ▶ *Without z/OS software support:*

On rank saturation volume I/O is managed according to volumes performances groups performance policy defined into HMC

- ▶ *With z/OS software support:*

The user assigns applications priorities through eWLM (WorkLoad Manager).

z/OS assigns an importance value to each I/O based on eWLM output.

z/OS assigns an achievement value to each I/O based on prior history I/O response time for I/O with the same importance based on eWLM expectations for response time.

**Tip:** I/O Priority Manager allows multiple workloads on a single CKD volume with different priorities each other, and permits the High important tasks always to have the best performance

For more details on the use and configuration of I/O Priority Manager, see the Redbooks publication, *IBM System Storage DS8000 Architecture and Implementation*, SG24-8886, and the Redpaper publication, *DS8000 I/O Priority Manager*, REDP-4760.

## 12.8.2 Easy Tier and I/O Priority Manager coexistence

Easy Tier is a workload balance activity on back-end (ranks) and it works to balance workload on same ranks type and also to move high request tracks (hot data) to the more efficient ranks into the machine, usually SSD. For details, see the Redbooks publication, *IBM System Storage DS8000: Architecture and Implementation*, SG24-8886.

I/O Priority Manager works to balance I/O activity on Host Adapter side, limiting the access to the low priority hosts or tasks. It means that while a rank is entering in saturation, to avoid the possibility that all hosts and tasks that are using this rank will have a performance issue.

**Important:** Easy Tier and I/O Priority Manager can coexist because they work on different sides of the machine workload and both can improve machine performance.

Figure 12-6 shows where Easy Tier and I/O Priority Manager can be enabled by web GUI.

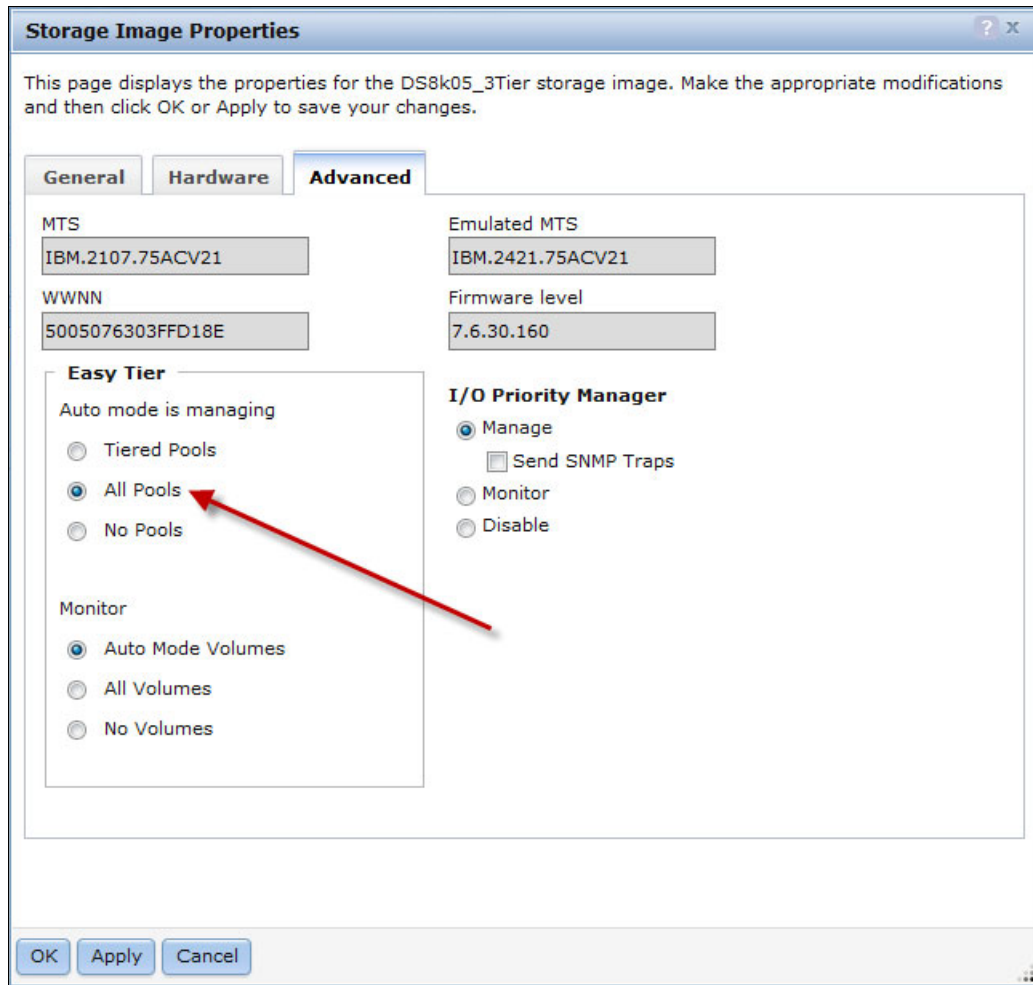


Figure 12-6 Easy Tier and I/O Priority Manager setup

**Tip:** Easy Tier should work in *auto mode* by selecting the **All pools** option to obtain the maximum performance.

## 12.9 TPC-R V5.1 in a z/OS environment

The IBM Tivoli® Storage Productivity Center for Replication for System z products are designed to support hundreds of replication sessions across thousands of volumes for both open and System z attached volumes. These products starting at V5.1 have an intuitive graphical user interface (GUI) and also a command line interface (CLI) to help you to configure pairing more easily.

For more information about system requirements and IBM HyperSwap® activity, see following the Redbooks publications:

- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z V5.1, SC27-4055*
- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z V5.1 Installation and Configuration Guide, SC27-4053*
- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z V5.1 User's Guide, SC27-4054*

### 12.9.1 Tivoli Storage Productivity Center for Replication for System z (TPC-R)

The TPC-R for System z combines the former Two-Site BC and Three-Site BC (Business Continuity) offering into a single license. It provides support for Metro Mirror and Global Mirror configurations as well as three-site recovery management, supporting IBM System Storage DS8000 Metro Global Mirror and Metro Global Mirror with HyperSwap. It is designed to support fast failover and failback, fast reestablishment of three-site mirroring, data currency at the remote site with minimal lag behind the local site, and quick re-synchronization of mirrored sites using incremental changes only.

Tivoli Storage Productivity Center for Replication for System z products support these features:

- ▶ FlashCopy
- ▶ Basic HyperSwap
- ▶ Metro Mirror
- ▶ Metro Mirror with HyperSwap
- ▶ Metro Mirror with Open HyperSwap
- ▶ Global Mirror
- ▶ IBM System Storage DS8000 Metro Global Mirror
- ▶ IBM System Storage DS8000 Metro Global Mirror with HyperSwap

#### **New configuration support**

Tivoli Storage Productivity Center for Replication for System z provides support for IBM DS8000 V6.3 Extent Space Efficient volumes in all Remote Copy Services relationships. Combined with DS8000 V6.3, Users can now use Extent Space Efficient (ESE) volumes in any role of a remote copy services relationship.

#### **Metro Mirror Management improvements**

The new Metro Mirror Management (MGM) improvements require APAR OA37632. Tivoli Storage Productivity Center for Replication for System z provides improved management capability for Metro Mirror sessions containing system volumes. Enablement of HyperSwap is no longer required. This feature also helps ensure that data consistency is still maintained even if Tivoli Storage Productivity Center for Replication for System z is not active at the time of a disaster.

## Improved management and alerting

The TPC-R for System z provides the ability to set a “warning” and a “severe” threshold on Recovery Point Objective (RPO) metrics for a Global Mirror session. If the RPO surpasses the warning threshold, an SNMP event will be thrown and a message issued for the Tivoli Storage Productivity Center for Replication for System z session indicating that the RPO has passed the warning point and, therefore, might soon pass the severe threshold. If the RPO surpasses the severe threshold, an SNMP event will be activated, a message issued, and the session should indicate a severe status until the RPO again drops below the threshold.

Additionally, Tivoli Storage Productivity Center for Replication for System z allows the user to export Global Mirror history to a CSV file for RPO and Out of Sync analysis. This feature extends the reporting capability already available and provides a means of storing this history for extended periods of time for greater report capability.

## Global Mirror enhancements

TPC-R V5.1 provides the ability to export Global Mirror history to a CSV file for RPO and Out of Sync Track analysis. This feature extends the reporting capability already available and provides a means of storing this history for extended periods of time for greater report capability.

## Enhanced session flexibility

TPC-R V5.1 for System z provides the ability to start a Global Mirror session in Global Copy mode. This gives the user better control of replication services activity as DS8000 family storage systems begin forming consistency groups.

## IBM WebSphere Application Services

Tivoli Storage Productivity Center for Replication for System z supports IBM WebSphere® Application Services for z/OS V7 and V8 in 31 bit and 64 bit.

## HyperSwap enhancements

The new HyperSwap enhancements require APAR OA37632. Exploits features of the IBM DS8700 and DS8800 that allows HyperSwap to obtain information about the cause of storage replication suspension events, and based on that information, is able to allow systems to continue to run where it was not previously possible.

## 12.9.2 References

For more information about system requirements and operability, see the following Redbooks publications:

- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z V5.1*, SC27-4055
- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z V5.1 Installation and Configuration Guide*, SC27-4053
- ▶ *IBM Tivoli Storage Productivity Center for Replication for System z V5.1 User's Guide*, SC27-4054

## 12.10 Full Disk Encryption (FDE)

Encryption disks must be managed by a key server, usually a couple of dedicated TKLM servers that provide full access to the DS8000.

When encryption is enabled on a DS8000 system, the disk is locked when a rank is created, assigning an access credential. It occurs before data is stored. To access data, the DS8000 system must provide the access credential each time a locked disk is powered on. If a rank is deleted, the disk generates a new encryption key, which replaces the existing encryption key and crypto-erases the disk. The DS8000 generates the access credential for a locked IBM FDE drive using a data key that it obtains from the Tivoli Key Lifecycle Manager.

After this step completes, the DS8000 can write and read data on the disk, with the disk encrypting data on writes and decrypting data on reads. Without access to the Tivoli Key Lifecycle Manager, the DS8000 system does not know the disk access credential and cannot read or write data on the disk. When an encryption group is configured on the DS8000 system, the system obtains a data key from the Tivoli Key Lifecycle Manager.

To prevent key server encryption deadlock, you might have more than one individual with access to any single piece of information that is required to determine an encryption key. When this method is used, you can configure redundant key servers.

Redundant key servers are two or more key servers that have independent communication paths to encrypting storage devices. You must create at least two key servers before you can create an encryption group. For more information about redundancy and deadlock prevention, see the “Encryption concepts” section of the Information Center.

Starting from release 6.2 (LMC level 7.6.20.xx) on z/OS environments, the security key can be also managed internally by z/OS hosts using ISKLM 1.1. It will provide same functionality provided by TKLM servers, as described previously, without needing machines external to the host.

Nothing has changed from the DS8000 point of view. The new version of key server software for zSeries platform is based off the original encryption key manager (EKM) that was used for tapes.

- ▶ ISKLM 1.1 provides a GUI interface similar to TKLM.
- ▶ It provides a migration path for zSeries EKM Customers to manage security key without needing to have TKLM servers.

**Encryption:** The DS8800 now has a full range of encrypted disks, including SSD 400 GB, that permits to have Easy Tier fully operational also on encrypted machines.

For more detailed information, see the Redbooks publication, *IBM System Storage Data Encryption*, SG24-7797, and the Redpaper publication: *IBM System Storage DS8700 Disk Encryption*, REDP-4500.





# IBM SAN Volume Controller considerations

This chapter describes the guidelines of how to attach an IBM System Storage DS8000 system to the IBM SAN Volume Controller (SVC) system.

The following topics are covered:

- ▶ IBM System Storage SAN Volume controller
- ▶ SAN Volume controller multipathing
- ▶ Configuration guidelines

## 13.1 IBM System Storage SAN Volume Controller

The IBM System Storage SAN Volume Controller (SVC) is designed to increase the flexibility of your storage infrastructure by introducing an in-band virtualization layer between the servers and the storage systems. The SAN Volume Controller can enable a tiered storage environment to increase flexibility in storage management. The SAN Volume Controller combines the capacity from multiple disk storage systems into a single storage pool. It can be managed from a central point, which is simpler to manage and helps increase disk capacity utilization. It also allows you to apply SVC advanced Copy Services across storage systems from many vendors to help further simplify operations.

For more information about SAN Volume Controller, see the Redbooks publication, *Implementing the IBM System Storage SAN Volume Controller V6.3*, SG24-7933.

## 13.2 SAN Volume Controller multipathing

Each SAN Volume Controller node presents a VDisk to the SAN via multiple paths. A typical configuration is to have four paths per VDisk. In normal operation, two nodes provide redundant paths to the same storage, which means that, depending on zoning and SAN architecture, a single server might see eight paths to each LUN presented by the SAN Volume Controller. Each server host bus adapter (HBA) port needs to be zoned to a single port on each SAN Volume Controller node.

Because most operating systems cannot resolve multiple paths back to a single physical device, IBM provides a multipathing device driver. The multipathing driver supported by the SAN Volume Controller is the IBM Subsystem Device Driver (SDD). SDD groups all available paths to a virtual disk device and presents it to the operating system. SDD performs all the path handling and selects the active I/O paths.

SDD supports the concurrent attachment of various DS8000 models, Storwize® V7000, and SAN Volume Controller storage systems to the same host system. Where one or more alternate storage systems are to be attached, you can identify the required version of SDD at this website:

<http://www.ibm.com/support/docview.wss?uid=ssg1S7001350>

## 13.3 Configuration guidelines for SVC

The following sections give general guidelines for attaching the SVC to DS8000.

### 13.3.1 Determining the number of controller ports for DS8000

To avoid performance issues, it is important to configure a minimum of eight controller ports to the SVC per controller regardless of the number of nodes in the cluster. Configure 16 controller ports for large controller configurations where more than 48 ranks are being presented to the SVC cluster.

Additionally, it is advisable that no more than two ports of each of the DS8000's 4-port adapters are used.

Table 13-1 shows the preferred number of DS8000 ports and adapters based on rank count.

Table 13-1 Preferred number of ports and adapters

Ranks	Ports	Adapter
2 - 48	8	4-8
>48	16	8 - 16

**Configurations:**

- ▶ Configure a minimum of eight ports per DS8000.
- ▶ Configure 16 ports per DS8000 when > 48 ranks are presented to the SVC cluster.
- ▶ Configure a maximum of two ports per four port DS8000 adapter.
- ▶ Configure adapters across redundant SAN networks from different I/O enclosures.

### 13.3.2 LUN masking

For a given storage controller, all SVC nodes must see the same set of LUNs from all target ports that have logged into the SVC nodes. If target ports are visible to the nodes that do not have the same set of LUNs assigned, SVC treats this situation as an error condition and generates error code *1625*.

Validating the LUN masking from the storage controller and then confirming the correct pathcount from within the SVC are critical.

Example 13-1 shows *lshostconnect* output from the DS8000. Here, you can see that all 8 ports of the 2-node cluster are assigned to the same volume group (V1) and, therefore, were assigned to the same set of LUNs.

Example 13-1 The *lshostconnect* command output

```

dscli> lshostconnect
Date/Time: 31. Mai 2012 14:04:33 CEST IBM DSCLI Version: 7.6.20.221 DS: IBM.2107-75TV181
Name          ID   WWPN                HostType Profile                portgrp volgrpID
=====
ITS0_SVC_N1_1 0000 500507680110455A SVC      San Volume Controller  1 V1
ITS0_SVC_N1_2 0001 500507680120455A SVC      San Volume Controller  1 V1
ITS0_SVC_N1_3 0002 500507680130455A SVC      San Volume Controller  1 V1
ITS0_SVC_N1_4 0003 500507680140455A SVC      San Volume Controller  1 V1
ITS0_SVC_N2_1 0004 5005076801104CD1 SVC      San Volume Controller  1 V1
ITS0_SVC_N2_2 0005 5005076801204CD1 SVC      San Volume Controller  1 V1
ITS0_SVC_N2_3 0006 5005076801304CD1 SVC      San Volume Controller  1 V1
ITS0_SVC_N2_4 0007 5005076801404CD1 SVC      San Volume Controller  1 V1
=====

```

It is advisable to have one single volume group for the whole SVC cluster and to assign this volume group to all SVC ports.

**Important:** Data corruption can occur if LUNs are assigned to both SVC nodes and non-SVC nodes, that is, direct-attached hosts.

The Managed Disk Link Count (*mdisk link count*) represents the total number of MDisks presented to the SVC cluster by that specific controller.

### 13.3.3 DS8000 extent pool implications

In the DS8000 architecture, extent pools are used to manage one or more ranks. An extent pool is visible to both processor complexes in the DS8000, but it is directly managed by only one of them. You must define a minimum of two extent pools with one extent pool created for each processor complex to fully use the resources.

The current approach, considering the availability of Storage Pool Striping and Easy Tier for the DS8000, is to create a few DS8000 only extent pools, for instance, two. Then, use either DS8000 storage pool striping or automated Easy Tier rebalancing to help prevent from overloading individual ranks.

You need only one MDisk volume size with this approach, because plenty of space is available in each large DS8000 extent pool. Often, clients choose 2 TiB (2048 GiB) MDisks with this approach. Create many 2-TiB volumes in each extent pool until the DS8000 extent pool is full, and provide these MDisks to the SAN Volume Controller to build the storage pools.

You can find a more detailed discussion in the IBM Redbooks publication, *DS8800 Performance Monitoring and Tuning*, SG24-8013.

# Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this book.

## IBM Redbooks publications

The following IBM Redbooks publications provide additional information about the topic in this document. Some publications referenced in this list might be available in softcopy only.

- ▶ *DS8000 Performance Monitoring and Tuning*, SG24-7146
- ▶ *DS8000 Thin Provisioning*, REDP-4554
- ▶ *IBM PowerVM Virtualization Managing and Monitoring*, SG24-7590
- ▶ *IBM System z Connectivity Handbook*, SG24-5444
- ▶ *IBM System Storage DS8000: LDAP Authentication*, REDP-4505
- ▶ *IBM System Storage DS8700 Easy Tier*, REDP-4667
- ▶ *Migrating to IBM System Storage DS8000*, SG24-7432
- ▶ *PowerVM and SAN Copy Services*, REDP-4610
- ▶ *PowerVM Virtualization on IBM System p: Introduction and Configuration Fourth Edition*, SG24-7940
- ▶ *Multiple Subchannel Sets: An Implementation View*, REDP-4387

You can search for, view, or download Redbooks publications, Redpaper publications, Technotes, draft publications, and Additional materials, as well as order hardcopy Redbooks publications, at this website:

[ibm.com/redbooks](http://ibm.com/redbooks)

## Other publications

These publications are also relevant as further information sources:

- ▶ *Device Support Facilities: User's Guide and Reference*, GC35-0033
- ▶ *IBM System Storage DS8000: Command-Line Interface User's Guide*, SC26-7916
- ▶ *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917
- ▶ *IBM System Storage DS8000 Introduction and Planning Guide*, GC35-0515
- ▶ *IBM System Storage DS Open Application Programming Interface Reference*, GC35-0516
- ▶ *IBM System Storage Multipath Subsystem Device Driver User's Guide*, GC52-1309

## Online resources

These Web sites are also relevant as further information sources:

- ▶ Documentation for the DS8000:  
<http://publib.boulder.ibm.com/infocenter/dsichelp/ds8000ic/index.jsp>
- ▶ IBM data storage feature activation (DSFA) website:  
<http://www.ibm.com/storage/dsfa>
- ▶ System Storage Interoperation Center (SSIC):  
<http://www-03.ibm.com/systems/support/storage/ssic/interoperability.wss>

## Help from IBM

IBM Support and downloads:

[ibm.com/support](http://ibm.com/support)

IBM Global Services:

[ibm.com/services](http://ibm.com/services)



Redbooks

## IBM System Storage DS8000: Host Attachment and Interoperability

(0.5" spine)

0.475" x 0.873"

250 <-> 459 pages









# IBM System Storage DS8000 Host Attachment and Interoperability



**Redbooks®**

**Learn how to attach DS8000 to open systems, IBM System z, and IBM i**

**See how to gain maximum availability with multipathing**

**Discover best practices and considerations for SAN boot**

This IBM Redbooks publication addresses host attachment and interoperability considerations for the IBM System Storage DS8000 series. Within this book, you can find information about the most popular host operating systems platforms, including Windows, IBM AIX, VIOS, Linux, Solaris, HP-UX, VMware, Apple, and IBM z/OS

The topics covered in this book target administrators or other technical personnel with a working knowledge of storage systems and a general understanding of open systems. You can use this book as guidance when installing, attaching, and configuring System Storage DS8000.

The practical, usage-oriented guidance provided in this book complements the *IBM System Storage DS8000 Host Systems Attachment Guide*, SC26-7917.

## **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)

SG24-8887-01

ISBN 073843759X