

CISCO SYSTEMS

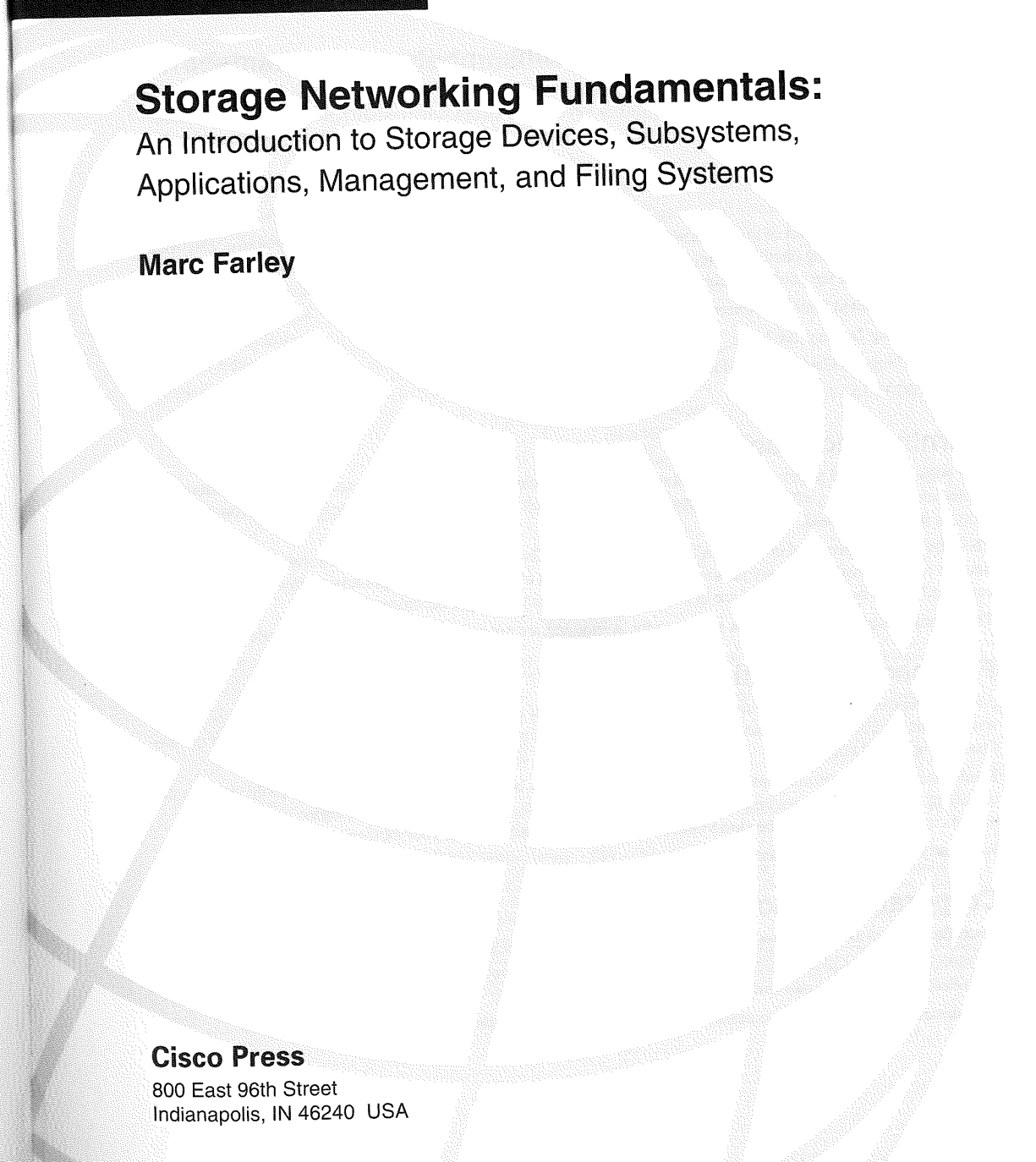


Storage Networking Fundamentals

An Introduction to Storage Devices, Subsystems,
Applications, Management, and File Systems

ciscopress.com

Marc Farley



Storage Networking Fundamentals:

An Introduction to Storage Devices, Subsystems,
Applications, Management, and Filing Systems

Marc Farley

Cisco Press

800 East 96th Street
Indianapolis, IN 46240 USA

Storage Networking Fundamentals

Marc Farley

Copyright© 2005 Cisco Systems, Inc.

Published by:

Cisco Press

800 East 96th Street

Indianapolis, IN 46240 USA

All rights reserved. No part of this book may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage and retrieval system, without written permission from the publisher, except for the inclusion of brief quotations in a review.

Printed in the United States of America 2 3 4 5 6 7 8 9 0

Second Printing January 2007

Library of Congress Cataloging-in-Publication Number: 2003108300

ISBN: 1-58705-162-1

Warning and Disclaimer

This book is designed to provide information about storage technologies used in SAN and NAS network storage implementations. Every effort has been made to make this book as complete and accurate as possible, but no warranty or fitness is implied.

The information is provided on an "as is" basis. The authors, Cisco Press, and Cisco Systems, Inc. shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book or from the use of the discs or programs that might accompany it.

The opinions expressed in this book belong to the author and are not necessarily those of Cisco Systems, Inc.

Feedback Information

At Cisco Press, our goal is to create in-depth technical books of the highest quality and value. Each book is crafted with care and precision, undergoing rigorous development that involves the unique expertise of members of the professional technical community.

Reader feedback is a natural continuation of this process. If you have any comments on how we could improve the quality of this book, or otherwise alter it to better suit your needs, you can contact us through e-mail at feedback@ciscopress.com. Please be sure to include the book title and ISBN in your message.

We greatly appreciate your assistance.

Corporate and Government Sales

Cisco Press offers excellent discounts on this book when ordered in quantity for bulk purchases or special sales.

For more information please contact: U.S. Corporate and Government Sales 1-800-382-3419
corpsales@pearsontechgroup.com

For sales outside the U.S., please contact: International Sales international@pearsoned.com

Trademark Acknowledgments

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Cisco Press or Cisco Systems, Inc. cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

Publisher	John Wait
Editor-in-Chief	John Kane
Executive Editor	Jim Schachterle
Cisco Representative	Anthony Wolfenden
Cisco Press Program Manager	Nannette M. Noble
Production Manager	Patrick Kanouse
Development Editor	Andrew Cupp
Project Editor	Marc Fowler
Copy Editor	Gayle Johnson
Technical Editors	Philip Lowden
	John Nelsen
	Guy Shimabuku
Team Coordinator	Tammi Barnett
Book and Cover Designer	Louisa Adair
Composition	Mark Shiraz
Indexer	Larry Sweazy



Corporate Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-4000
800 553-NETS (6387)
Fax: 408 526-4100

European Headquarters
Cisco Systems International BV
Haarlerbergpark
Haarlerbergweg 13-19
1101 CH Amsterdam
The Netherlands
www.europe.cisco.com
Tel: 31 0 20 357 1000
Fax: 31 0 20 357 1100

Americas Headquarters
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA
www.cisco.com
Tel: 408 526-7660
Fax: 408 527-0883

Asia Pacific Headquarters
Cisco Systems, Inc.
Capital Tower
168 Robinson Road
#22-01 to #29-01
Singapore 068912
www.cisco.com
Tel: +65 6317 7777
Fax: +65 6317 7799

Cisco Systems has more than 200 offices in the following countries and regions. Addresses, phone numbers, and fax numbers are listed on the
Cisco.com Web site at www.cisco.com/go/offices.

Argentina • Australia • Austria • Belgium • Brazil • Bulgaria • Canada • Chile • China PRC • Colombia • Costa Rica • Croatia • Czech Republic
Denmark • Dubai, UAE • Finland • France • Germany • Greece • Hong Kong SAR • Hungary • India • Indonesia • Ireland • Israel • Italy
Japan • Korea • Luxembourg • Malaysia • Mexico • The Netherlands • New Zealand • Norway • Peru • Philippines • Poland • Portugal
Puerto Rico • Romania • Russia • Saudi Arabia • Scotland • Singapore • Slovakia • Slovenia • South Africa • Spain • Sweden
Switzerland • Taiwan • Thailand • Turkey • Ukraine • United Kingdom • United States • Venezuela • Vietnam • Zimbabwe

Copyright © 2003 Cisco Systems, Inc. All rights reserved. CCIP, CCNP, the Cisco Arrow logo, the Cisco Powered Network mark, the Cisco Systems Verified logo, Cisco Unity, Follow Me Browsing, FormShare, iQ Net Headlines Scorecard, Networking Academy, and ScriptShare are trademarks of Cisco Systems, Inc.; and Aironet, ASIST, BPX, Catalyst, CCDA, CCDP, CCIE, CCNA, CCNP, Cisco, the Cisco Certified Internetwork Expert logo, Cisco IOS, the Cisco IOS logo, Cisco Press, Cisco Systems, Cisco Systems Capital, the Cisco Systems logo, Empowering the Internet Generation, Enterprise Solver, EtherChannel, EtherSwitch, Fast Step, GigaStack, Internet Quotient, IOS, IPTV, iQ Expertise, the iQ logo, LightStream, MGX, MICA, the Networkers logo, Network Registrar, Packet, PDX, Post-Routing, Pre-Routing, RateMUX, Registrar, SlideCast, SMARTnet, StrataView Plus, Stratum, SwitchProbe, TeleRouter, TransPath, and VCO are registered trademarks of Cisco Systems, Inc. and/or its affiliates in the U.S. and certain other countries.

All other trademarks mentioned in this document or Web site are the property of their respective owners. The use of the word partner does not imply a partnership relationship between Cisco and any other company. (0303R)

Printed in the USA

About the Author

Marc Farley is well-known throughout the network storage industry for his objective and in-depth analysis of network storage technology. His writings include *Building Storage Networks*, First and Second editions, as well as numerous articles, technology white papers, and opinion pieces that have been published in the industry trade press. He regularly speaks at industry events and participates in online seminars and discussions.

Farley started working in network storage in 1991 as a systems engineer at Palindrome Corporation, a company that developed network backup systems for Novell NetWare environments. Since then, he has worked in a variety of marketing strategy roles on a wide variety of storage technologies, including storage routers, disk subsystems, file-level virtualization, and storage switches. Farley has a bachelor of science in physics from the University of Washington.

Ab

Philip
Syste
UND
and s
degre
scien
is a S

John
Syste
indus
2000
on th
focus
to bu
degre
1988

Guy
func
mana
opm
field
work

About the Technical Editors

Philip Lowden currently holds the position of customer support engineer for Storage Area Networking at Cisco Systems, Inc. Before this role, he worked four years at Cisco and six years at Electronic Data Systems as a senior UNIX systems administrator, performing production systems architecture and support duties on a variety of host and storage platforms. He was also an officer in the U.S. Air Force for six years. He holds a master of science degree in computer engineering from North Carolina State University, a bachelor of science degree in computer science from the University of Nebraska, and a bachelor of arts degree in English from Saint Meinrad College. He is a SNIA-certified FC-SAN Specialist. He is married and has two children.

John Nelsen is a member of the Cisco Infrastructure Architecture Team and a project manager within the Cisco Systems IT Enterprise Storage Solutions Group. Before joining Cisco, he worked for 10 years in the pharmaceutical industry. He has 20 years of experience in various IT architecture and operational roles. When he joined Cisco in 2000, his primary responsibilities included defining and selling a strategy to reduce storage TCO that was founded on the concept of a consolidated storage utility. Since gaining acceptance for the strategy in 2001, Nelsen has focused on defining the hardware and software technologies along with the business processes and projects required to build and implement a storage architecture that can support the storage utility. He received a bachelor of science degree from Allegheny College in 1984 and a master of science degree from North Carolina State University in 1988.

Guy Shimabuku is a senior engineer with the Architecture and Enterprise Services Group for Verizon. His primary function is evaluating storage and networking elements for Verizon's next-generation data centers. In addition, he manages a research laboratory that encompasses 125 TB of storage and 500 Fibre Channel ports serving 600 development servers. He is a graduate of California Polytechnic University School of Engineering and has been in the field of networking and data storage for 15 years. He has been published and interviewed in trade magazines for his work on testing and implementing broadband networks.

Dedications

To you, Dad. You are imprinted on everything I think and do. Even so, I'm doing fine. I still miss you terribly after all these years.

Ac

I'd like
when I
He also
uted he

Toward
Bruce t
relation
works,

Mike V
undisp
and ho
talk ab
teacher

A heap
This is
clusters
and org
world.

Big tha
it was j
project
John N
that ne

Thanks
Potter f

Acknowledgments

I'd like to thank several people who made major contributions to this book. Matt O'Keefe helped me immensely when I was just getting under way by pointing out weaknesses in my previous analysis of system storage processes. He also got me started on the more realistic and Byzantine analysis that runs throughout this book. Matt has contributed heavily to the development of open-source storage software for Linux and possesses major storage wisdom.

Toward the end of the project, I circled up with Bruce Thompson for a sanity check and some finishing touches. Bruce truly has an advanced view of storage, coming from many years of working through the strange and frustrating relationships between file systems, operating systems, and device drivers. When Bruce tells you how something works, you can take it to the bank.

Mike Workman is no longer involved in the day-to-day grindings of the disk drive industry; nonetheless, he is an undisputed authority on the subject. Sincere best wishes for you, Mike, and your team at Pillar. May your star rise and hover long. Big thanks also to my good friend John Groves, the SCSI I/O master in Austin. John and I like to talk about everything from music to politics to storage. But when it comes to storage I/O processes, John is the teacher and I am the student.

A heaping helping of gratitude to Steve Legg, a soft-spoken, hugely talented, and patient man if ever there was one. This is a person I want to take human being lessons from, but, in the meantime, I'll settle for insights on virtualization, clusters, and multipathing. Joel Harrison is one of those rare people who comprehends technology, industry, people, and organizations equally well with amazing deep-blue clarity and deft humor. He is a true master of the storage world.

Big thanks to James Long for his generous assistance with determining the contents and scope of the book when it was just a bunch of raw ideas. Also, an enormous debt of gratitude goes to Jim Schachterle for marshalling this project through the long journey with an author who seldom met deadlines. The technical editors, Phil Lowden, John Nelsen, and Guy Shimabuku, also deserve credit for their work reviewing the contents and pointing out things that needed to be set right.

Thanks also to John Howarth, Ron Riffe, Jeff Browning, Jeff Barnett, Joaquin Ruiz, Gordon Arnold, and Charles Potter for their help.

Contents at a Glance

Introduction xxv

Part I The Big Picture of Storage Networking 3

Chapter 1 Data Access in the Internet Era 5

Chapter 2 Establishing a Context for Understanding Storage Networks 23

Chapter 3 Getting Down with Storage I/O 41

Part II Working with Devices and Subsystems in Storage Networks 67

Chapter 4 Storage Devices 69

Chapter 5 Storage Subsystems 89

Chapter 6 SCSI Storage Fundamentals and SAN Adapters 117

Chapter 7 Device Interconnect Technologies for Storage Networks 139

Part III Applications for Data Redundancy 171

Chapter 8 An Introduction to Data Redundancy and Mirroring 173

Chapter 9 Bigger, Faster, More Reliable Storage with RAID 191

Chapter 10 Redundancy Over Distance with Remote Copy 209

Chapter 11 Connection Redundancy in Storage Networks and Dynamic Multipathing 231

Part IV The Foundations of Storage and Data Management 247

Chapter 12 Storage Virtualization: The Power in Volume Management Software and SAN Virtualization Systems 249

Chapter 13 Network Backup: The Foundation of Storage Management 275

Part II Filing Systems and Data Management in Networks 305

Chapter 14 File System Fundamentals 307

Chapter 15 Network File Systems and Network Attached Storage 325

Chapter 16 New Directions in Network Filing: Clustered File Systems, Distributed File Systems, and Network Storage for Databases 345

Chapter 17 Data Management 371

Part VI Appendixes 391

Appendix A Q & A Answers 393

Appendix B INCITS Storage Standards 409

Glossary 415

Index 431

Contents

Introduction xxv

Part I The Big Picture of Storage Networking 3

Chapter 1 Data Access in the Internet Era 5

Availability Requirements for Network Storage 6

Traditional Client/Server Computing with Direct Attached Storage 7

Introducing DAS 8

Connectivity Shortcomings of DAS 8

 Data Availability Depends on Server System Health 9

 Static Configuration of DAS Storage 9

 Distance Limitations of DAS 10

Business Issues with DAS 11

 High Cost of Managing DAS Storage 11

 Captive Resources and Storage Utilization 12

 Limited Scalability with DAS 13

Network Storage Architectures That Overcome DAS Limitations 14

 One-to-Many Relationships with DAS 14

 Many-to-Many Relationships with Storage Networking 15

 Connection Flexibility of Storage Networks 16

Architectures for Data-Centric Computing 18

 Enterprise Storage 20

 Redundancy and Data-Centric Topologies 20

Summary 21

Q & A 21

Chapter 2 Establishing a Context for Understanding Storage Networks 23

The Three Primary Functions of Storage Networking 23

Connecting 24

 A Quick Look at Buses Used for Connecting DAS Storage 24

 A Quick Look at Networks Used for Connecting SAN Storage 26

 The Logical Side of Storage Connections 26

Storing 27

 Role of the Storage Controller 27

 Initiators, Targets, and Command/Response Protocols 28

Block I/O 28

The Storage Address Space 29

 Storage Address Space Manipulation 29

 Volumes, Exported Drives, and Virtual Storage 30

Filing, File Systems, and Operating Systems	31
Space Allocation	31
Filing's Role in Access Control and Security	32
File I/O	32
SAN and NAS as Storing and Filing Applications	34
Summary	38
Q & A	39

Chapter 3	Getting Down with Storage I/O	41
	Requirements for Storage I/O	41
	Performance Requirements	41
	Bandwidth	41
	Latency	42
	Reliability Requirements	44
	Integrity Requirements: Write Ordering and Data Consistency	44
	The I/O Path	46
	Host System I/O Path Elements	47
	Core System Hardware: Processors, Memory, and Memory Bus	47
	System I/O Hardware: The I/O Bus and Host Bus Adapters	48
	Host System Storage Software: File Systems, Volume Management, Multipathing Drivers, and HBA Device Drivers	49
	The Complete Host System I/O Path	54
	Network Systems and Components in the I/O Path	55
	Network Hardware	55
	Network Software	56
	The Complete Path Through Network Systems and Components	58
	Storage Subsystem Hardware	58
	Subsystem Network Ports	59
	Storage Controller	59
	Cache Memory	59
	Internal Interconnect	59
	Storage Device	59
	Storage Media	59
	The Complete View of Storage Subsystem I/O Path Elements	60
	The Operating System's Role in the I/O Path	60
	File System Integration	61
	The I/O Operation	62
	Installable File Systems	62
	Summary	63
	Q & A	64

Part II Working with Devices and Subsystems in Storage Networks 67**Chapter 4 Storage Devices 69**

Disk Drives 69

Major Parts of a Disk Drive 69

Disk Platters 70

Read and Write Heads 71

Read/Write Channel 72

Arms and Actuators 72

Drive Spindle Motor and Servo Control Electronics 73

Buffer Memory 73

Disk Controller 74

Data Structures on Disk Drives 75

Tracks, Sectors, and Cylinders 75

Disk Partitions 75

Logical Block Addressing 76

Geometry of Disk Drives and Zoned-Bit Recording 77

Disk Drive Specifications 78

Mean Time Between Failures 78

Rotational Speed and Latency 78

Average Seek Time 79

Media Transfer Rate 79

Sustained Transfer Rate 80

Optimizing Disk Drive Performance in Storage Networks 80

Limiting Drive Contention 80

Short-Stroking the Drive 81

Matching Rotation Speeds 81

Aligning Zones 81

Tape Drives 82

A Look at Tape Media 82

Caring for Tape 83

Caring for Tape Heads 83

Tape Drive Performance 83

Streaming and Start-Stop Operations 84

Compression 84

The Tale of Two Technologies 84

Linear Tape Technology 84

Helical Scan Tape Technology 85

Comparing Tape Technologies 85

Summary 86

Q & A 86

Chapter 5 Storage Subsystems 89

- Subsystem Architecture 89
 - Subsystem Controllers 90
 - Subsystem Internal Connectivity 91
 - Aggregating Devices in Subsystems 92
 - High Data Availability with Redundant Subsystem Designs 93
 - Redundant Storage Devices 93
 - Dual-Ported Devices and Dual Internal Networks or Buses 94
 - Redundancy in Storage Controllers and Processors 95
 - Redundant Cache Memory 96
 - Redundant Network Ports 97
 - Redundant Power and Cooling 97
 - Capacity Scalability 98
 - Capacity Scaling with Striping 98
 - Capacity Scaling with Concatenation 98
 - Performance Advantages of Subsystems 99
 - Parallelism and Overlapped I/Os 99
 - Caching and I/O Performance 100
 - Write-Through and Write-Back Caches 101
 - Modular Enclosure Designs 102
 - Hot Swapping 102
 - Hot Spares 103
 - Storage Enclosure Services 103
 - Modular Cabinet Components 104
 - Common Functions of Storage Subsystems 105
 - Storage Address Space Management 105
 - Differentiating Arrays and Subsystems 106
 - Accessing Exported Storage with LUN Addressing 107
 - Multiplexing I/Os, Link Oversubscription, and Fan-In 108
 - LUN Masking 110
 - The SMI Standard and Container Management 111
 - Redundant Access to Subsystem Storage 112
 - JBOD 112
 - Tape Subsystems and Disk Subsystems for Backup 113
 - Autoloaders 113
 - Libraries 113
 - Disk Subsystems 114
 - Summary 114
 - Q&A 114

Chapt

Chapt

Chapter 6	SCSI Storage Fundamentals and SAN Adapters	117
	The Architecture and Logic of SCSI	118
	History of SCSI	118
	The T10 SCSI Standards Committees	118
	SCSI-3 Connection Independence	118
	SCSI Architecture Model	119
	Initiators and Targets	119
	Initiator and Target Ports	120
	SCSI Remote Procedure Call Structure	121
	Overlapped I/O	122
	Asymmetrical Communications in SCSI	122
	Dual-Mode Controllers	123
	No Guarantee for Ordered Delivery	123
	SCSI Ports, IDs, and Names	123
	SCSI Logical Units	124
	Tasks, Task Sets, and Tagged Tasks	126
	SCSI Nexus and Connection Relationships	127
	Tagged Command Queuing	127
	SCSI Command Basics	128
	Command Descriptor Blocks	129
	Types of SCSI Commands	129
	Basic SCSI Commands	130
	SCSI Reservation	131
	Extended Copy and Third-Party Copy	131
	HBAs and Network Storage Interfaces	132
	HBAs as SCSI Initiators	133
	Qualification of HBAs by System and Subsystem Vendors	134
	Multiple SAN, Multiple HBA Configurations	134
	Device and Subsystem Network Controllers/Adapters	135
	Summary	135
	Q & A	136
Chapter 7	Device Interconnect Technologies for Storage Networks	139
	Fundamental Concepts in Storage Interconnects	140
	Legacy DAS Interconnects	141
	The Parallel SCSI Bus: the Legacy Server Storage Interconnect	141
	Parallel SCSI Bus Connections	142
	Parallel SCSI Addresses, Arbitration, and Priority	143
	A Comparison of Single-Ended, Differential, and LVD SCSI Buses	145
	Parallel SCSI Applications	146
	Parallel SCSI Storage Network Configurations	146
	Dual-Controller Parallel SCSI Connections	151

ATA: The Evolution of Desktop Storage to Lightweight Server Storage	152
A Short History of ATA	152
ATA Connections	154
ATA Interconnect Performance	154
ATA Network Storage Applications	155
ATA Storage Network Configurations	155
Serial Storage Interconnects	158
Fibre Channel Loops as an Interconnect for Network Storage	158
Fibre Channel Topologies	158
A Brief History of Fibre Channel	158
Fibre Channel Interconnect Connections	159
Fibre Channel Performance and Applications	160
Fibre Channel Interconnect Configurations	160
Fibre Channel Disk Drives in JBOD and RAID Subsystems	161
Retooling ATA Storage with Serial ATA	162
Point-to-Point Connections	162
The Short, Confusing History of SATA	162
Performance Expectations for SATA	164
SATA Port Multiplier	164
SATA Port Selector	165
SATA Storage Network Applications and Configurations	165
SATA Disk Drives Within Systems	165
SATA in SAN Disk Subsystems	166
Summary	167
Q & A	168
Part III Applications for Data Redundancy	171
Chapter 8 An Introduction to Data Redundancy and Mirroring	173
The Necessity of Redundancy in Storage Networks	173
Threats to Data	174
Redundancy Metrics: MTBF, MTDA, MTDL, and MTTR	174
MTBF	175
MTDA	175
MTDL	175
MTTR	175
Forms of Data Redundancy	176
Duplication	176
Parity	177
Delta or Difference Redundancy	177

Mirroring	178
Mirroring Operators, Targets, and Mirrored Pairs	178
Disk Mirroring	179
Location of Mirroring Operators	179
Mirroring in Host System Software	180
Mirroring in Host Storage Controllers	181
Mirroring in a Network Device or Switch	181
Mirroring in a Storage Subsystem	182
Multiple Mirrors and Multilevel Data Redundancy	183
The Mirroring Function	183
I/O Termination and Reinitiation	183
Identically Sized Storage Address Spaces	185
Overlapped Read Operations for Performance	186
Assumptions About Distance and Mirroring	186
Summary	187
Q & A	188
Chapter 9 Bigger, Faster, More Reliable Storage with RAID	191
RAID History	191
RAID Fundamentals	192
RAID Arrays and Members	192
RAID Controllers	193
Benefits of RAID	193
RAID Redundancy with Parity	193
Capacity Scaling Through Striping	194
Consolidated Management	195
Performance Through Striping	195
RAID Levels	196
RAID Level 1	196
RAID Level 5	196
RAID Level 0	197
RAID 10 and Multilevel RAID	197
RAID 6	197
Parity Redundancy in RAID	197
Reliability of Parity RAID Arrays	198
XOR Parity	199
Properties of XOR	199
Degraded Mode Operations	201
Parity Rebuild Operations	201
Strips and Stripes Stripe Depth	202

RAID 5	203
RAID 5 Write Penalty	203
Avoiding Parity Limitations of RAID with RAID 10	204
RAID Product Options	205
Host Volume Managers	205
RAID on Motherboard (ROM)	206
Host Bus Adapters	206
Disk Subsystems	206
Network Systems	206
Summary	207
Q & A	207
Chapter 10 Redundancy Over Distance with Remote Copy	209
Fundamentals of Remote Copy	209
Basic Architecture of Remote Copy Applications	210
Remote Copy Sites and Storage Hierarchies	210
Primary Storage and Sites	210
Secondary Storage and Sites	211
Tertiary Storage and Sites	211
Bunker Sites	212
Objectives of Remote Copy Applications	212
Immediate Availability to Support Ongoing Operations	212
Data Integrity, Consistency, and Atomicity	213
Resuming Normal Primary Site Operations	215
Remote Copy Application Processing	216
The Remote Copy System	216
Forwarding Controller as SCSI Target	216
Forwarding Controller as SCSI Initiator	216
Receiving Controller in Dual Modes: SCSI Target and Initiator	217
Specialized Networking Equipment	218
Performance Implications of Remote Copy	218
Latency of Light Propagation	219
The Four-Stroke SCSI Write Process	219
Synchronous, Asynchronous, and Semisynchronous Operating Modes	220
Synchronous Mode	220
Asynchronous Mode	221
Semisynchronous Mode	222
Link Failure Mode	222

Remote Copy Implementation Options	223
Remote Copy in Disk Subsystems	223
Remote Copy in the SAN with SAN Virtualization Systems, SAN Appliances, and Switch Application Blades	224
Remote Copy in Host Software	225
Examples of Remote Copy Architectures	226
"Ping Pong" Remote Copy Between Two Sites	226
Bunker Secondary Storage and Remote Tertiary Storage	227
Summary	227
Q & A	228

Chapter 11 Connection Redundancy in Storage Networks and Dynamic Multipathing 231

Redundant I/O Path Elements	231
Redundant Server Systems	232
Redundant HBAs, HBA Ports, and Cables	232
Network Switching Equipment	233
Subsystem Ports	233
Subsystem Interconnects	233
Dynamic Multipathing	234
The Big Picture of Dynamic Multipathing	234
Differentiating Between Mirroring, Routing, and Multipathing	235
Subsystem Multipathing Structures	235
A Review of SCSI Logical Units and LUNs	235
World Wide Node Names and World Wide Port Names	237
WWPNs, LUNs, and LUs	237
Host System Multipathing Functions	239
Host Storage Initiators in Multipathing	239
Implementing Multipathing Software	239
Determining Paths	240
Active/Passive Configurations and Static Load Balancing	241
Active/Active Configurations with Dynamic Load Balancing	243
Failover Processing	243
Recognizing Path Failures	243
Preparing to Fail Over	244
Initiating Operations on the New Active Path	244
Failing Back	244
Multipathing and Network Route Convergence	244
Summary	245
Q & A	245

Part IV	The Foundations of Storage and Data Management	247
Chapter 12	Storage Virtualization: The Power in Volume Management Software and SAN Virtualization Systems	249
	The Concept of Storage Virtualization	249
	A Review of Technologies Used in Storage Virtualization	250
	Virtualization Products: Volume Managers and SAN Virtualization Systems	251
	Volume Management	251
	SAN Virtualization Systems	252
	In-Band Virtualization	254
	Out-of-Band Virtualization	254
	Performance of SAN Virtualization	255
	Latency	255
	Caching	255
	Scaling Storage with Virtualization	255
	The Virtualization "Lens"	256
	Fan-Out of a Virtualization Lens	256
	Recursivity of Virtualization	257
	Comparing the "Views" of Volume Managers, SAN Virtualization Systems, and Storage Subsystems	258
	Address Space Manipulation Techniques	259
	The Storage Flat Address Space	260
	Aggregation with Striping and Concatenation	260
	Subdivision	261
	Substitution	261
	Mirroring	261
	Private Data Areas and Metadata	261
	Storage Pooling	262
	Storage Utilization with Pooled Storage	262
	Scalability of Pooled Storage	264
	Target Mix and Breadth in Pooled Storage	265
	Tiered Storage Pools	266
	Performance Classifications for Tiered Storage	266
	Device Performance	266
	Parallelism in Storage	266
	Capacity Utilization Levels	266
	Block Size	266
	Application Types for Storage Pooling	267
	Tiered Storage Pools for Data and Storage Management	267

Extending Management Functions in Volume Management and SAN Virtualization Systems	267
Hot Sparing with Volume Managers and SAN Virtualization Systems	268
Automated Data Migration Using Volume Managers and SAN Virtualization Systems	268
Reliability and Risk Considerations	269
The SAN Network Environment	269
Multipathing with Volume Managers and SAN Virtualization Systems	269
SAN Virtualization System Clusters	270
Summary	271
Q & A	272

Chapter 13 Network Backup: The Foundation of Storage Management 275

Fundamentals of Backup and Recovery	275
Backup Versus Recovery	275
Backup Applications	276
Business Continuity Protection	276
Historical Archiving	276
System and Application Migration	277
Data Sharing	277
Backup as a Filing Application	277
Offline, Online, and Near-Line Storage	277
Offline Storage	278
Online Storage	278
Near-Line Storage	278
The Backup System	279
Backup Engine	279
Backup Agents	279
Operations Scheduler	280
Backup Transfer Network	280
Removable Backup Media and Media Management	281
Tape Devices and Subsystems	282
Device Interconnect or SAN Connection	282
Backup Metadata	282
The Big Picture of Backup	283
Backup Operations	284
The Backup Window	284
Types of Automated Backup Operations	284
Full Backup	285
Incremental Backup	285
Differential Backup	285
Special-Purpose Backup Operations	286

Hot Backup and Copy-on-Write	286
Cold Backup Operations	286
Hot Backup Operations	287
Copy-on-Write	288
Continuous Backup	289
Limitations of Hot Backup	289
Backup Applications in Storage Networks	289
Problems with Legacy Network Backup	289
LAN Limitations	290
Distributed Backup Systems and Management	290
Optimal Performance and Management with SANs	291
Separating the Control and Data Paths	291
LAN-Free Backup	292
Serverless Backup Using Extended Copy	294
Serverless Backup Data Movers	294
Serverless Backup Engines	295
Serverless Backup Agent Software	295
Virtualization and Serverless Backup	296
Disk-Based Backup	296
Advantages and Disadvantages of Disk Technology for Backup	296
Problems with Tape	297
Advantages of Disk Subsystems for Backup	298
Reasons for Tape's Continued Use	298
Disk Backup Architecture	299
Disk to Disk	300
Disk to Disk to Tape	300
Integrated Disk and Tape	300
Summary	301
Q & A	302
Part V	Filing Systems and Data Management in Networks
Chapter 14	File System Fundamentals
	307
	The Strange and Wonderful Relationship Between File Systems and Operating Systems
	307
	The Separation of Time and Space in Systems
	308
	File Systems as High-Priority Applications
	309
	Public and Private Interfaces
	310
	File System Structures
	310
	Logical Directory Structures
	310
	Files
	311
	Directories
	311
	Full and Relative Path Names
	312

Layout of Data Within a Storage Address Space	313
Block Size Definition	314
Allocating Storage from File System Free Space	314
Fragmentation	315
The Layout Reference System	315
Traversing the File System	317
File System Constructs and Functions	317
Superblock	317
The Name Space	318
Metadata and Attributes	318
Access Permissions	319
Interface Semantics	319
Locking	319
File System Buffers and Cache	319
File System Checking and Journaling	320
Scaling File Systems	321
Copying Files into a New File System on New Storage	321
File System Expansion	322
Summary	322
Q & A	323

Chapter 15 Network File Systems and Network Attached Storage 325

Fundamentals of Network File Systems	325
A Review of Block I/O and File I/O	326
Network File System Clients	327
File I/O Redirection	327
Remote-Mounting the Network File System	327
File Server Systems	327
File Server Software and the Network Filing Service	328
The Server File System	329
The Complete I/O Process for File I/O in a Network	329
NFS and CIFS Protocols for Network File Systems	331
NFS	331
NFSV4	331
CIFS	332
Locking Differences Between NFS and CIFS	333
Applications for Network File Storage	334
Centralized Storage Management	334
Data Sharing	334

Extended Features of Network File Systems	335
Disk Quotas	335
Name Space Virtualization	335
Large Block Size Definitions	335
Volume Management	335
Storage Resource Management	336
Domain Services	336
SAMBA	336
Network Attached Storage (NAS)	337
NAS Designs	337
Physical Dimensions	337
Storage Capacity	338
Interconnect and Disk Technology Selection	338
Operating System	338
File System	338
Journaling (or Not)	339
Data Management	339
Performance and Throughput	339
Availability and Reliability	340
Dual-Function Storage: NAS and SAN	340
Network Domain Services	341
Backup and Recovery of NAS Systems with NDMP	341
NAS Scalability	342
Dynamic File System Expansion	342
NAS Head	342
Adding More NAS Systems: Less Than Wonderful	342
Summary	343
Q & A	343
Chapter 16 New Directions in Network Filing: Clustered File Systems, Distributed File Systems, and Network Storage for Databases	345
Cluster File Systems	345
Basic Cluster Designs	346
Two-Node Active/Passive Clusters	347
Multinode Active/Passive Clusters	347
Active/Active Clusters	348
Two File System Approaches for Clusters: Share Everything or Share Nothing	349
Shared Everything Cluster	349
Shared Nothing Cluster	350
Structural Considerations for Cluster File Systems	351
Layout Reference System	351
Buffer Memory	351

Locking	352
Journaling	352
Global Name Space	352
Implementing Storage for Clustered File Systems	353
Multipathing and Clustering	353
Importance of SANs in Clusters	353
Uniformity of Storage Address Spaces in Clusters	353
Data Caching in Clusters	354
Disk Drive Considerations for Clusters	354
RAID Levels for High Availability and Performance in Clusters	355
NAS Clusters	355
Super-Scaling Network File Systems with SAN-Based Distributed File Systems	356
DFS Designs	357
Farm-Style Redundancy	357
Global Name Space	357
Application Integration or Separation	358
Layout Reference System Location	360
Integrated Volume Manager	362
Distributed or Centralized Lock Manager	363
Expanded Metadata (Optional)	363
Advantages of Distributed File Systems	363
Modular Scalability and Performance Tuning	363
Storage for Tiered Storage	364
Dynamic Address Space Expansion	364
Storage Migration	365
Data Snapshots	365
Network Storage for Databases	365
Database Filing: Raw Partitions	365
Storing Databases on File Systems	366
Oracle 10g Automated Storage Manager	366
Database Data Types	366
Log Data	367
Tablespace Data	367
Database Backup and Business Continuity	367
Databases and Network Storage	368
Databases and SANs	368
Databases and NAS	368
Databases and Distributed File Systems	368
Summary	368
Q & A	369

Chapter 17	Data Management	371
	Managing Data in Time	372
	Recreating a System State Through Point-in-Time Snapshot Technology	372
	Whole Volume Snapshots	373
	Volume Delta Snapshots	375
	File System Snapshots	376
	Practice Good Data Hygiene and Flush Data from Host Buffers	377
	Continuous Volume Snapshots	377
	File Replication	378
	Historical Versions of Files	378
	Creating Historical File Versions	379
	Data Archiving	379
	Creating Historical Versions with Backup	380
	Creating Historical Versions with Snapshot Technology	381
	Accessing Historical File Versions	381
	Restoring Files from Backup Tapes	382
	Accessing Snapshot Versions of Files	382
	Storing Data in Compliance with Government Regulations	383
	Risks of Noncompliance	383
	Immutable Data	384
	Write-Once Storage	384
	Content Addressable Storage	384
	E-Mail Archiving and Indexing	384
	Capacity Management	385
	Storage Resource Management	386
	Hierarchical Storage Management	386
	Tiered Storage	387
	Information Life Cycle Management	388
	Summary	389
	Q & A	389
Part VI	Appendixes	391
Appendix A	Q & A Answers	393
Appendix B	INCITS Storage Standards	409
Glossary		415
Index		431

Introduction

When I was a
some projects
weak compon

For a couple y
present them
technologies
ested. Fortun
and ignore m
technology re

One of the m
often have a l
virtual disk, c
to distinctly c
end of confus
some differer

There is a lot
outdated assu
tion between
driver softwa
ways—as it c
respective ro

The concept
Understanding
system or da
assure you th
interpreting
address spac

Throughout
storage devi
file system.
what goes on
ming device
Once things
rying old as

Another key
mentals and
readers wor
and works f
tural unders

The concep
has not exis
reason to be
other, newe

Introduction

When I was a boy, my dad used to tell me, "A job is only worth doing if you do it right." I've since learned that for some projects "getting it right" involves many years, repeated attempts, and the ability to identify and throw away weak components that do not further the goals of the project.

For a couple years, I'd been feeling that I wanted to write another book that would take on more difficult topics and present them in a way that allows an experienced networking professional to gain a deeper understanding of storage technologies and processes. So when Cisco Press approached me about working on a new storage book, I was interested. Fortunately, their concept aligned with mine, which was to focus almost entirely on the storage technologies and ignore most of the networking content. I've always preferred the murky Neanderthal swamp of storage to the technology reservoirs of network standards anyway.

One of the most difficult things in discussing storage is terminology. The problem is that the terms used in storage often have a historical context that no longer fits with new architectures and environments. For example, the terms virtual disk, disk volume, and LUN are often used to refer to the same type of storage entity, but they can also refer to distinctly different things within the same expanded I/O system. These types of overlapping definitions cause no end of confusion in the industry. I knew from the outset that explaining storage better this time was going to require some different techniques, terms, and analysis.

There is a lot of literature in computer science where storage is simply referred to as "a disk drive" because of old, outdated assumptions embedded in operating systems. Historically, there had never been a reason to make a distinction between the file system's view of a collection of SCSI block addresses and the thing that an HBA device and driver software actually communicate with. However, when the downstream storage is virtualized in any number of ways—as it often is in a SAN—it becomes very useful to clarify the pieces of storage that come into play and their respective roles.

The concept of a "storage address space" is introduced early in the book (in Chapter 2, "Establishing a Context for Understanding Storage Networks") to refer to the flat, contiguous string of addresses where data is stored by a file system or database. Now the words "storage address space" might not be a phrase that leaps off the tongue, but I can assure you that if you start using them to refer to an abstract, addressable "chunk" of storage, many difficulties interpreting storage become much, much easier. No caveats or asterisks are needed to augment the term storage address space, like there are with more familiar terms like virtual disk, disk, and volume.

Throughout the book, a perspective on the relationships between host storage software, storage subsystems, and storage devices is included. This necessarily includes the relationship between the operating system kernel and the file system. A surprising thing about operating system/file system interactions is that very few people understand what goes on at any detailed level. This is not an environment based on object orientation or structured programming devices, but instead is an environment optimized by performance with cunning algorithms and shortcuts. Once things work well in an operating system, every attempt is made not to change it. Hence, the problem with carrying old assumptions forward, well past their appropriate age.

Another key to understanding storage is knowing SCSI protocols and processes. Chapter 6, "SCSI Storage Fundamentals and SAN Adapters," will be a pivotal chapter for many readers, and I expect it will be referred to often by readers working on concepts in later chapters. I hope my explanation of SCSI architecture is at the appropriate level and works for you. Like so many things, it is not necessary to know the details of a technology if a good architectural understanding exists and provides predictable operating principles.

The concept of a "layout reference system" is used in the discussions of file systems. Again, a good general term has not existed for the logical entity that describes the location of data in a storage address space. There is simply no reason to believe that I nodes or V nodes would necessarily have to be the technology used to track stored data in other, newer file systems. My apologies to file system developers. I am not trying to change what you do; I am only

trying to explain it in an accessible, generic fashion, the same as I would do if I were trying to write a requirements document for a file system.

Trying to clarify old, existing constructs by inventing new generic terms carries some amount of personal risk, because it is not at all clear that people will accept them. But for this attempt it was clear that some new language and new analysis were needed to do the job "right." I didn't see any other way to make storage easier to think about and understand.

This book is the result of a great deal of work and a lot of help from many people who graciously pitched in. From my perspective, I can honestly say that no shortcuts were taken, no assumptions went unquestioned, and no conceptual coasting was done in writing this book. I look forward to hearing from readers in the years to come.

Who Should Read This Book?

This book was written for systems, networking, and storage professionals who want to gain an in-depth understanding of the processes and architectures used in storage. It is intended to be read by systems engineers and architects as well as network, systems, and storage administrators who are responsible for planning, implementing, and managing network storage solutions. It is not a cookbook, but an explanation of storage architectures, applications, and processes used in real-world environments.

How This Book Is Organized

Storage implementations combine many different technologies, and any in-depth discussion of a real-world storage implementation could use concepts from every chapter of this book. One could certainly read the book in chapter sequence, but it is much more likely that readers will first read the chapters that are most immediately applicable to specific tasks and responsibilities in their environments.

In general, the book is structured as follows:

- Chapter 1 discusses the rationale for network storage.
- Chapters 2 and 3 present a broad picture of how storage I/O processes work.
- Chapters 4 and 5 illustrate the functions and architectures of storage devices and subsystems.
- Chapter 6 is primarily about the architecture and protocols of SCSI. A brief discussion of HBAs and NICs appears at the end of this chapter.
- Chapter 7 examines the various storage interconnects used for device attachment.
- Chapters 8 and 9 discuss redundancy, mirroring and RAID.
- Chapters 10 and 11 present the processes used for remote copy and multipathing redundancy protection.
- Chapter 12 is about virtualization and volume management.
- Chapter 13 covers backup and recovery.
- Chapters 14 to 16 look at file system designs for network storage.
- Chapter 17 discusses data management.
- Appendix A contains answers and explanations for the Q & A review questions at the end of each chapter.
- Appendix B is a brief listing of INCITS standards pertaining to storage networking.
- The glossary defines key storage terms used throughout the book.

CHAPTER 2

Establishing a Context for Understanding Storage Networks

People who come to storage networking with experience in Ethernet and TCP/IP networking have a tendency to start analyzing storage network architectures using the familiar protocol stack method that has been so effective in data networking. While protocol stacks obviously exist in storage networks, the stack approach does not necessarily reveal anything very useful or relevant for storage processes because so much of the interesting stuff in storage happens at the network application layer.

Much of this chapter explores a “three-legged stool” analysis for storage networking technology. This analysis breaks the subject into three distinct functional areas that divide the work of storage I/O processes and form a complete, functioning system:

- Connecting
- Storing
- Filing

The end of this chapter covers the major product groups for storage networking, storage area networking (SAN), and network attached storage (NAS). It also applies the analysis model to show how the basic functions are typically implemented in storage network solutions.

The Three Primary Functions of Storage Networking

Storage processes can be broken into the three different functional areas shown in Table 2-1. The fastest way to gain in-depth understanding of network storage is to know what each of these three functional areas does and what products typically provide them.

Table 2-1

The Three Primary Functional Areas Involved with Storage Networking

Functional Area	Description
Connecting	Provides the transmission of data between systems and storage.
Storing	A lower-level application with specialized commands and control protocols for system/device interactions.
Filing	Directs the placement of data objects in storage and is responsible for presenting data to applications and users.

NOTE

Readers who are familiar with my writing on this subject might notice a change with the use of the word “connecting” to replace the word “wiring.” Connecting more accurately conveys the breadth of hardware and software that were intended to be included together. The word “wiring” has strong connotations for many people that limit its interpretation to physical cables. It does not communicate the larger sphere of technologies containing such things as switches, routers, gateways, routing algorithms, addressing schemes, and network management.

Connecting

There has always been some way to connect systems to storage products. Essentially, this means providing some sort of data transmission system between storage devices or subsystems and the system processor.

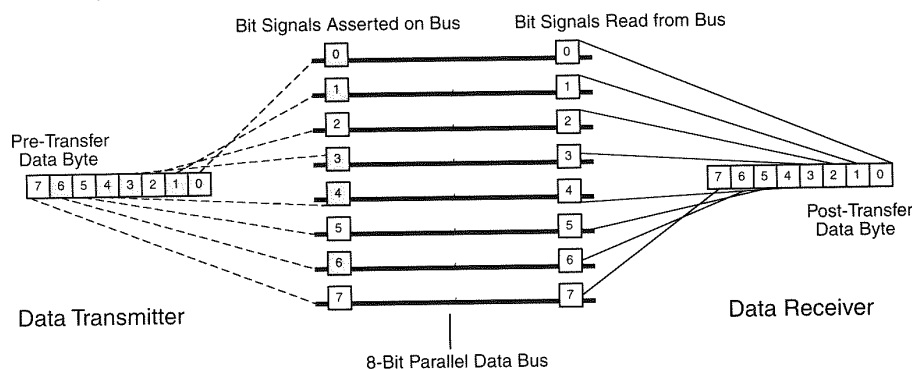
For the most part, storage has used either bus connecting technologies for DAS storage or network connecting technologies for SAN storage. The sections that follow briefly examine these two connecting technologies.

A Quick Look at Buses Used for Connecting DAS Storage

Bus technologies used with DAS storage are constructed from a collection of copper wires working in parallel, each transmitting a single bit of information at a time. Data transmitted over a bus is first disassembled from bytes into bits by the sender, transferred over the bus as binary signals (either a 0 or a 1), and then reassembled into bytes again by the receiver. All information is transferred in parallel, including application data, addresses, and communication protocols.

The process of disassembling data, transferring it over a parallel bus, and reassembling it at the receiving end is shown in Figure 2-1.

Figure 2-1 A Byte of Data Transmitted Across a Parallel Bus



ge with the use of
ately conveys the
The word
o physical cables.
ings as switches,
agement.

essentially, this
es or subsystems

AS storage or
riefly examine

copper wires
transmitted over
the bus as binary
. All information
cation protocols.
sembling it at the



ost-Transfer
Data Byte

ceiver

NOTE

When we discuss storage data transfers, it seems logical to picture storage data “moving” from one place to another, but of course nothing physical actually travels anywhere. Serial network transmissions work by signal propagation, similar to an analog telephone call. You hear somebody talking, but it is a representation of what he said, not the actual acoustic phenomenon.

Parallel bus transmissions, like those in Figure 2-1, are different. A DAS storing controller creates signals by changing the voltage on the eight data lines. The receiving node measures the voltage across these lines in parallel as an actual byte of data.

The physical characteristics of a bus, such as length, are heavily constrained by the requirement for precise time synchronization in sending and receiving each bit accurately. As a result, buses have inverse relationships between transmission rates and length—the faster a bus is, the shorter its maximum length can be.

Buses tend to be finite in scope with few degrees of freedom—and therefore they have limited failure modes. Data on a DAS storage bus is guaranteed to be received in the same order it was sent. This is called *guaranteed in-order delivery*. The primary advantage of bus technology is the reliability of the connection. The primary disadvantage of buses is that they are rigid and inflexible. In addition to their physical limitations, they have limited addressing schemes that do not accommodate key concepts like bridging and routing. Bus environments also lack basic management concepts such as domains, naming, and authentication.

The Small Computer Systems Interface (SCSI) bus has been used for two decades to connect storage and other peripherals such as scanners and printers to computer systems. Another common bus has been the Advanced Technology Attachment (ATA) bus, which has been used over the last decade as a way to connect desktop and laptop storage to PC systems.

SCSI bus cables come in two varieties: internal and external. Internal SCSI cables are assembled as ribbon cables, and external SCSI cables are fairly thick and heavy multistrand cables within a common shield. ATA cables are available only as internal ribbon cables. Both SCSI and ATA cables have fairly large connectors containing many pins or connecting sockets.

NOTE

ATA is also often referred to as Integrated Device Electronics (IDE). The terms have been used interchangeably by many, but ATA is more accurate as it refers to the name used in standards documents.

Readers wondering about Serial ATA (SATA) as an exception to the discussion here will find it discussed in Chapter 7, “Device Interconnect Technologies for Storage Networks.”

A Quick Look at Networks Used for Connecting SAN Storage

Unlike bus technologies, which have many parallel wires for transmitting binary signals, network technologies typically have four or fewer wires that serially transmit data as frequency modulated signals. Network cables can be made with fiber-optic technology, twisted-pair wiring, and coaxial cable as well as using wireless transmission. In practice, most storage network implementations use fiber-optic transmission technology due to its superior distance and bandwidth capabilities.

Networks also have much richer connection possibilities than buses. For starters, most modern networks have multifaceted topologies that enable highly scalable implementations. In addition, many types of technologies are used to transmit data between different physical networks, such as bridges, gateways, and routers, which increase a network's scalability. Perhaps most importantly, networking technologies usually have important management characteristics such as large address spaces, naming abstractions, and sophisticated measurement and alerting mechanisms.

For all their advantages, the fact that networks have so much more functional breadth than buses also means that they have many more variables and, therefore, potential problems. This is why products with advanced management functions are so important in storage networks.

As mentioned at the beginning of this chapter, one of the difficulties newcomers have in understanding storage networking is related to the assumptions they make based on their experience with TCP/IP and Ethernet data networks. In general, the design of data networking products assumes there might be delivery problems in the network and that data could be transmitted over different network paths. The bottom line is that in-order delivery of data is not guaranteed, as is the case with bus connectivity. The requirement for in-order delivery turns out to be extremely important for storage applications and is discussed more in Chapter 3, "Getting Down with Storage I/O," and Chapter 10, "Redundancy Over Distance with Remote Copy."

Fibre Channel networking technology has been the most common connecting technology used for SANs. The most recent connecting technology to be introduced is iSCSI, which is implemented on top of TCP/IP, meaning it can run on any physical network that supports TCP/IP. TCP/IP Ethernet networks are by far the most common for connecting systems to NAS storage appliances.

The Logical Side of Storage Connections

The connecting component of a storage network has logical aspects as well as physical. These logical processes incorporate such things as access algorithms, including fairness and congestion control, addressing and naming, network configuration and domain management, network operations management and other topics such as link aggregation and virtual networking. In essence, any technology used in a network for storage networking is part of the connecting component in a storage network.

This is a very rich topic with many interesting nuances. For the most part, this book focuses on storing and filing technologies in storage networks and does not address connectivity topics. Readers interested in exploring the connecting components of storage networks should seek other books that focus on such topics as storage protocols (FCP, iSCSI, and FC/IP), Fibre Channel, Ethernet, TCP/IP, and network switching technologies.

Storing

The second primary functional area in storage networking is *storing*—an area that is explored extensively in this book. Storing encompasses many types of storage technologies, including host-based volume managers and disk managers, storage subsystems, host bus adapters, storage devices, and storage virtualization.

Storing has its technology roots in the interactions between DAS storage devices and host computer systems over a bus. The storing methods used to configure and use storage devices have become standard functions in nearly all major operating systems. Although storing technology has changed a great deal over the years, especially since the advent of network storage, these old and simplistic operating system methods still determine how storage works.

Role of the Storage Controller

The physical electronic device or circuitry that actually performs storing functions is called a *storage controller*. Storage controllers are present in all storage products and are responsible for the detailed operations of reading, writing, and transmitting storage data.

Storage controllers are present in the following products:

- Host bus adapters (HBAs)
- Storage devices such as disk, tape, CD, and DVD drives
- Storage subsystems
- Storage appliances
- Storage routers, gateways, and channel extenders
- Switches with integrated storage functions

Storage controllers are often implemented in products that span both the connecting and storage functions. For instance, an HBA is used to formulate storing commands for storing devices and subsystems, but it also manages the transmission of those commands and storage data over the bus or network that connects them. Sometimes this is evident by the presence of multiple device drivers—there might be one device driver for the storing function plus another driver for the connecting function.

Initiators, Targets, and Command/Response Protocols

Storing processes are characterized by the interactions between controllers. Unlike peer-to-peer and client/server networks, where most entities have the same general capabilities, the functions available in a storage controller typically depend on the role it is designed for.

In a client/server or peer-to-peer network, almost all systems have the capability to act as either a server or a client. However, in storage networks, controllers are designed to function specifically as an *initiator* or a *target*.

Initiators are usually implemented as HBAs. HBA initiators receive requests for storage services through their device driver/kernel interface and then issue storing commands to satisfy those requests. A target is a controller in a storage device or subsystem that performs the actions requested by the initiator.

NOTE

It might be easier to understand the relationship between initiators and targets by using the terms *master* and *slave*. It's not necessarily technically accurate, but it's one of the fastest ways to appreciate the fundamental difference between storage networking and data networking architectures. Readers should know that there is a great deal of subtlety involved in the relationships of initiators and targets. Chapter 6, "SCSI Storage Fundamentals and SAN Adapters," discusses this topic in much more detail.

The protocols used for storage are referred to as command/response protocols and reflect the roles of initiators and targets. It is important to understand the nature of these protocols, as they influence the operation of the underlying network. For instance, a network technology might support full-duplex, bidirectional communications, but if the application protocol (in this case, the storing protocol) does not, the full capabilities of the network might not be able to be realized.

Block I/O

Storing processes store data in blocks, which are the basic storage capacity elements in storage networks. Block I/O is the process of writing and reading data in these granularly-sized amounts. Block I/O commands specify the type of operations to perform, the *block address* to work with, and the storage or control data being transferred. The size of a data block is determined by the file system or database system that manages the complete block storage address space.

The Storage Address Space

We are accustomed to talking about data storage in terms of disk drives. While it is true that nearly all online data is stored on disk drives, the storage element where data is directed and stored in storage networks is usually much more complex than a single disk drive. In a storage network environment, storage is created by merging and segmenting storage from multiple disk drives or storage subsystems. In these cases we often say that the storage is a “virtual disk.”

Unfortunately, the familiar terms disk, disk drive, virtual disk, virtual storage, exported storage, and logical unit number (LUN) are all used to refer to storage elements that provide the exact same fundamental storage function. Newcomers to storage are often confused by the lack of a single term that describes the architectural role of a data storage receptacle. This makes things unnecessarily awkward when discussing storage processes.

To keep things simple, this book uses the term *storage address space* to refer generically to all the various storage elements that are used for online data storage. A storage address space is defined herein as a sequence of contiguous, regularly sized storage blocks between a starting address and an ending address. From a host system perspective, a storage address space is the storage resource that a filing system manages.

Compare the following two sentences:

Data is written to a storage address space.

Data is written to either a physical disk or a virtual disk.

These two sentences say the same thing, but the first is much more straightforward and easier to grasp.

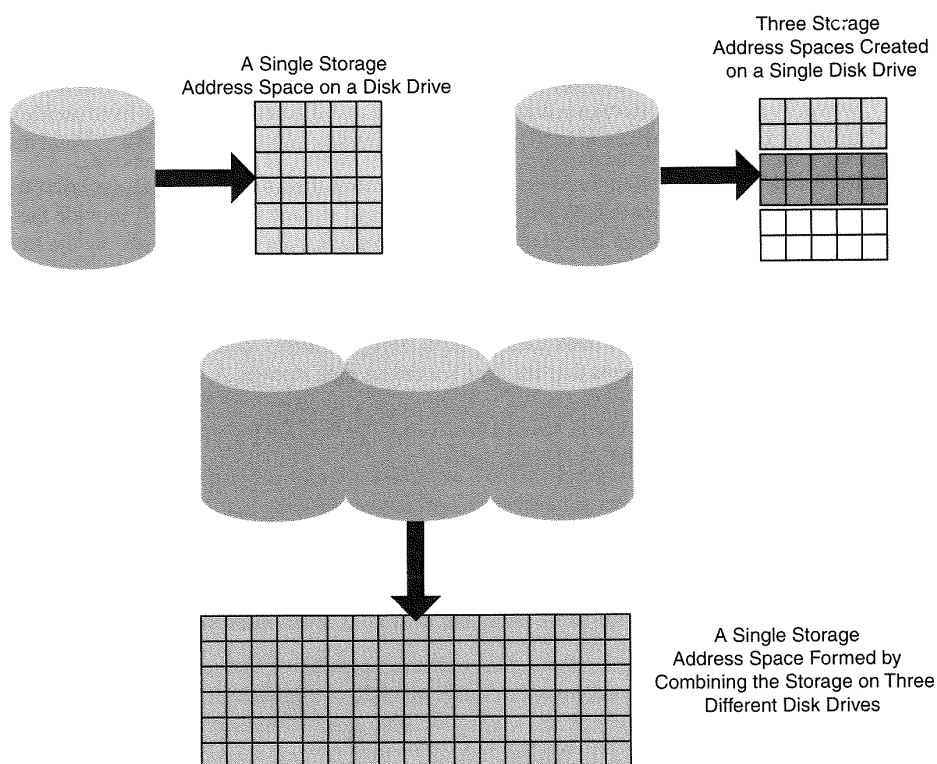
Figure 2-2 represents an address space made up of 20 contiguous blocks.

Figure 2-2 A (Very) Simple Storage Address Space of 20 Contiguous Blocks

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
---	---	---	---	---	---	---	---	---	----	----	----	----	----	----	----	----	----	----	----

Storage Address Space Manipulation

Storage address spaces are commonly manipulated in a variety of ways by storing products. On the simplest level, a disk partitioning process can create a single address space on a disk drive. Slightly more complex, a disk can be partitioned to create multiple logical drives, each with its own independent address space. Conversely, multiple disk drives can be combined to create a single larger address space. These three types of address space manipulation are shown in Figure 2-3.

Figure 2-3 *Basic Storage Address Space Manipulations*

Several address space manipulation techniques are commonly used in storage management applications, such as mirroring and RAID. These forms of address space manipulation are covered in Chapter 8, “An Introduction to Data Redundancy and Mirroring,” and Chapter 9, “Bigger, Faster, More Reliable Storage with RAID,” respectively.

Volumes, Exported Drives, and Virtual Storage

Storage address spaces can be created by several different types of storage products. Some of the most common terms used for storage address spaces are covered in Table 2-2.

Table 2-2 Common Terms Used for Storage Address Spaces

Storing Product	Term Used for Storage Address Space
Host system volume management software	Volume
Disk subsystem	Exported drive or virtual disk
SAN virtualization system	Virtual disk

Filing, File Systems, and Operating Systems

The last of the three fundamental storage network functions is *filing*. In general, filing's role is to organize how data is structured in a storage address space as well as to represent it to applications and users. In practice, two types of filing products are in the market: file systems and databases.

File systems are closely associated with operating systems and have very close interactions with them. Considering how much storage activity there can be within a system, it is imperative that the file system and operating system work very well together without errors.

In most cases, they appear to be part of the same product, but that does not have to be the case. It is possible to use file systems that are not part of the operating system as long as the operating system supports third-party file systems. Most modern operating systems do, including most UNIX systems, Windows server systems, and Linux.

Space Allocation

One way to understand the differences between filing systems (including databases) and operating systems is to separate them by the system functions they manage. Operating systems manage the sequence of processing events in the system. In other words, they schedule all work in the system and establish the proper environment for each task. Filing systems manage the placement of data in storage address spaces.

In general,

- Operating systems manage time
- Filing systems manage space

One of the primary responsibilities of the filing function is *space allocation*. This involves figuring out what storage blocks to use when new data is being written or when data is changed or updated, as well as what to do with blocks that might be freed when objects are changed or deleted. While storing functions perform the actual operations of reading and writing data, filing functions determine what those operations should be.

One of the challenges in space allocation is creating a system that provides consistent performance over an extended period of time as data objects are created, updated, and deleted.

Storage
Products Created
Disk Drive



File
Systems Created
on Three
Drives

Management
Allocation are
discussed in Chapter 9,

Products. Some of
2.

In order to achieve consistent performance, the filing system spreads data over the storage address space, providing a level of fairness to all data and its associated applications.

Filing's Role in Access Control and Security

Another key role of filing is access control. Filing systems often have attributes or access control lists (ACLs) that determine who has the authority to access data objects. To date, storing-level processes do not have the ability to provide access control, as storing level processes have no contextual information about data objects.

NOTE

DAS storing products never had to have access control capabilities, because all security functions were provided by the filing system and other system security functions. However, with SANs, it has become necessary for storage subsystems to provide some amount of access control to protect data resources. One storage technology used for access control is called LUN masking, and it is described in Chapter 5, "Storage Subsystems." Access controls for storage is an area that undoubtedly will see a great deal of development in the years to come.

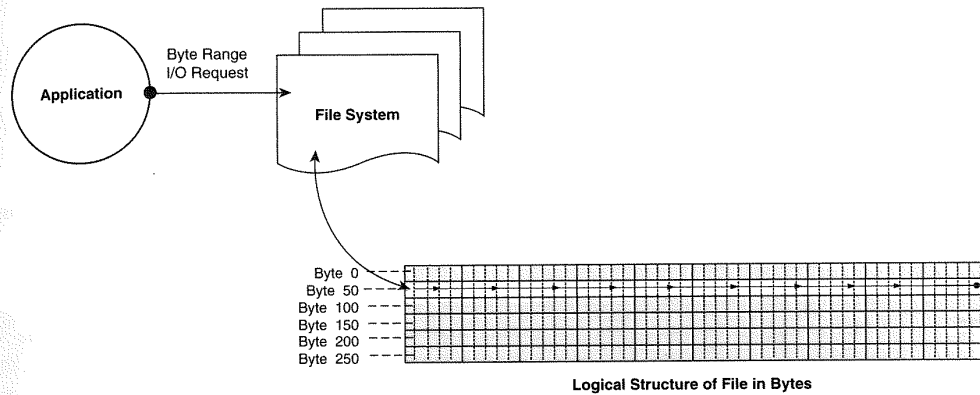
File I/O

As described previously, storing functions work with storage address spaces made up of contiguous, fixed-length block locations. To maintain performance expectations, filing systems logically fragment this neat and orderly structure by scattering data objects throughout the address space. Fortunately, the details of this data scattering are masked from users and applications by the file system.

File systems provide friendlier and more intuitive structures for accessing data, such as a directory tree and filenames. Applications are developed to access data through these structures using published interfaces that are easily invoked in the application's code. This type of data access is called *file I/O*.

While applications are abstracted from storing-level details, they know everything about the byte structure within their data files. For example, the header information in a file might be 100 bytes and the data might be 2000 bytes. Applications today have many different types of data within their files, and it is essential that they are able to access the various discrete sections of files efficiently and accurately.

Therefore, file systems allow applications to locate data within the file by its *byte location* or *byte range* within the file. Byte ranges can be expressed by the number of bytes from the beginning or end of a file as well as by offsets of bytes from these locations. For example, an application might issue a request to read 50 bytes of a file starting at the 50th byte, as shown in Figure 2-4.

Figure 2-4 *An Application Accesses Bytes 50 to 99 in a File*

The file system bears the responsibility of converting these byte-range requests to detailed storing data access operations.

File Systems and Storing Applications

File systems contain the mappings of data objects into storing address spaces. This mapping function might also include the ability to translate file byte range requests into storing requests. Notice that while the byte range request might be for contiguous bytes, the lower-level block storing addresses are probably not contiguous.

So, a question arises: Why aren't file systems considered storing applications? The reason is because I said so, and it's my analysis! There has to be a separating point somewhere, and this is where I drew the line. Mapping a data object to a bunch of storage addresses does not involve storing processes by my way of thinking.

File systems make use of lower-level storing interfaces and functions that are provided by the operating system. The operating system kernel passes storage I/O requests to the proper storage device drivers that formulate the I/O requests into storing commands that are transferred to storage over a bus or network. The file system knows absolutely nothing about storing commands or devices; it only knows about a flat storing address space—and what files are stuffed into which blocks within that storage address space.

This file system perspective of storage is the rationale behind using the term storage address space as a generalization for the wide variety of disks, disk partitions, volumes, and virtual storage.

SAN and NAS as Storing and Filing Applications

The two major product classes in storage networking are SAN and NAS. Analyzing these technologies within the system of connecting, storing, and filing, they can be easily differentiated by stating the following:

- SAN is the application of storing over a network.
- NAS is the application of filing over a network.

SANs are used to convey storing commands and data over a network, while NAS conveys filing commands.

NOTE

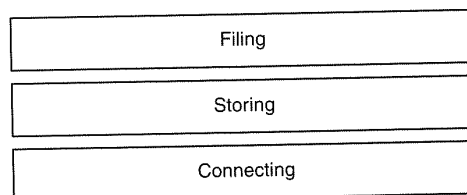
When storage networking was first getting started, there was enormous confusion over the merits of SAN and NAS: Was one inherently superior to the other?, Would one win out in the market in the long run?, and all sorts of other meandering speculations. The whole thing wasted an obnoxious amount of time.

It really bugged me that so many people were spinning their wheels on discussions that would not be resolved due to the lack of a fundamental functional analysis. That's what led me to develop the three-legged stool analysis of connecting, storing, and filing. For the most part, when people understand this analysis, their confusion over SAN and NAS positioning melts away.

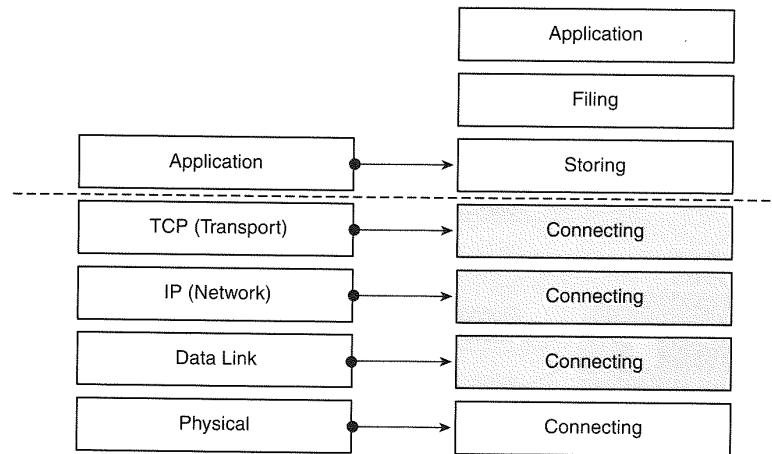
In the remainder of the chapter, we'll look at how connecting, storing, and filing work together in storage networking processes.

If there were a "stack" for storage processes, it would look like Figure 2-5, where connecting is the bottom-most layer, storing is in the middle, and filing is at the top-most layer.

Figure 2-5 *The Storage Process "Stack"*



For people familiar with networking stacks, Figure 2-6 shows the traditional TCP/IP network stack alongside the storage process stack. Notice that most of the interesting parts of the TCP/IP stack are part of the connecting component of the storage process stack.

Figure 2-6 A Comparison of the Storage Process Stack and the TCP/IP Protocol Stack

Now we'll start connecting various products and show their relative function within the storage process stack. Figure 2-7 shows a host bus adapter on the left connected to a disk subsystem on the right. The connecting function is provided by the SCSI bus and the communication function in the HBA and subsystem controller. The storing function is at a higher level in both controllers. In Figure 2-7 both the HBA and the disk subsystem controller have dual roles.

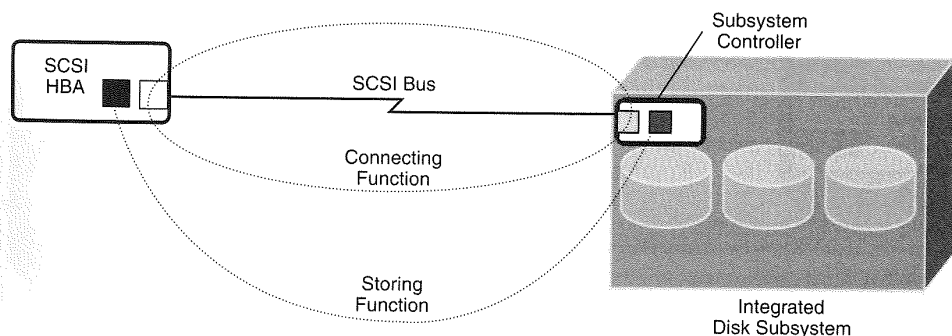
Figure 2-7 Storage and Connecting Processes Between an Interacting HBA and a Disk Subsystem

Figure 2-8 is the exact same figure but with a SAN switch between the HBA and the disk subsystem. Notice the changes in this figure from the last: the SAN switch and SAN cables replace the SCSI connection between the HBA and the subsystem. In Figure 2-8, the switch has only a connecting role.

Figure 2-8 *A Storage Switch and Cables Replace the SCSI Bus in Figure 2-7, Making a SAN*

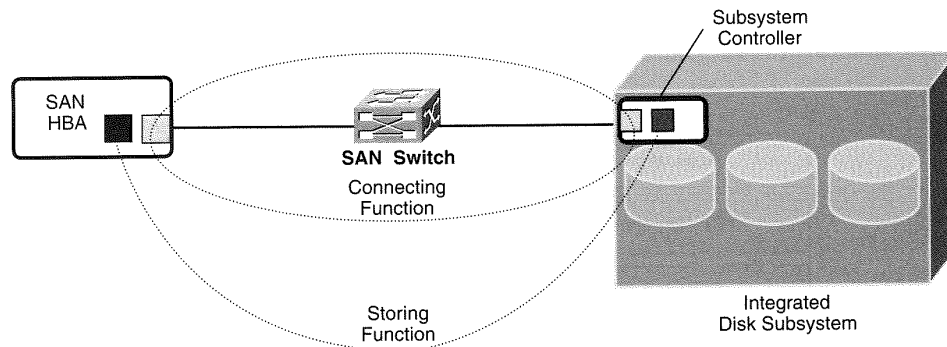


Figure 2-9 is the same figure again, but this time it shows the HBA residing in a host system along with application and file system software. Notice that there is now a filing function on the left, but not on the right.

Figure 2-9 *An Expansion of Figure 2-8, Adding the Host System and a File System*

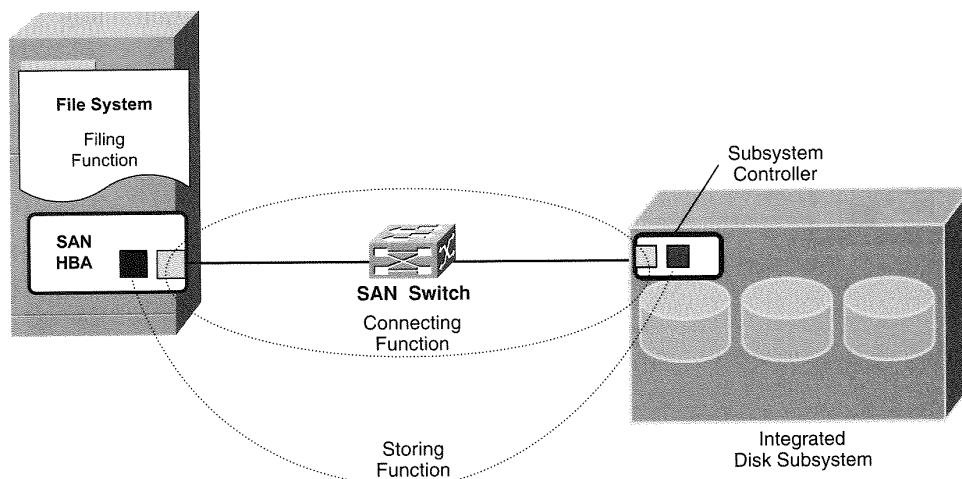
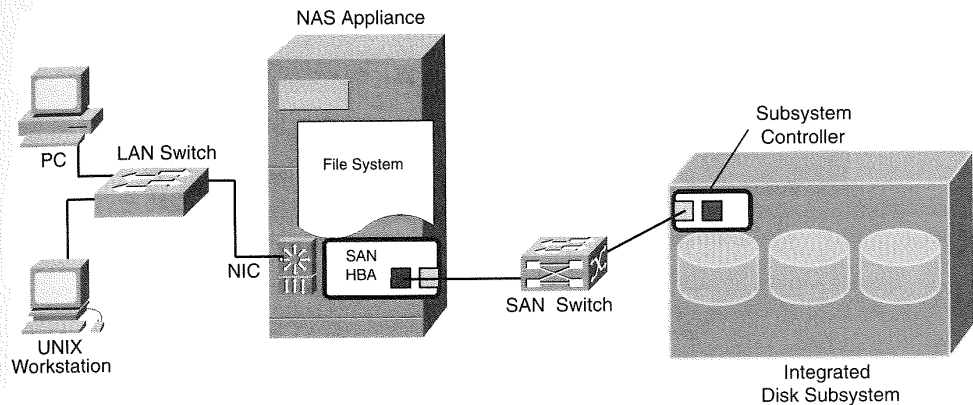


Figure 2-10 converts the system we have been building with this sequence of figures by turning the system from Figure 2-9 into a NAS appliance system with an integrated storage subsystem. This system also has a network interface card (NIC) for communicating with client systems over a LAN.

Figure 2-10 *Enlarging the Figure to Include NAS*

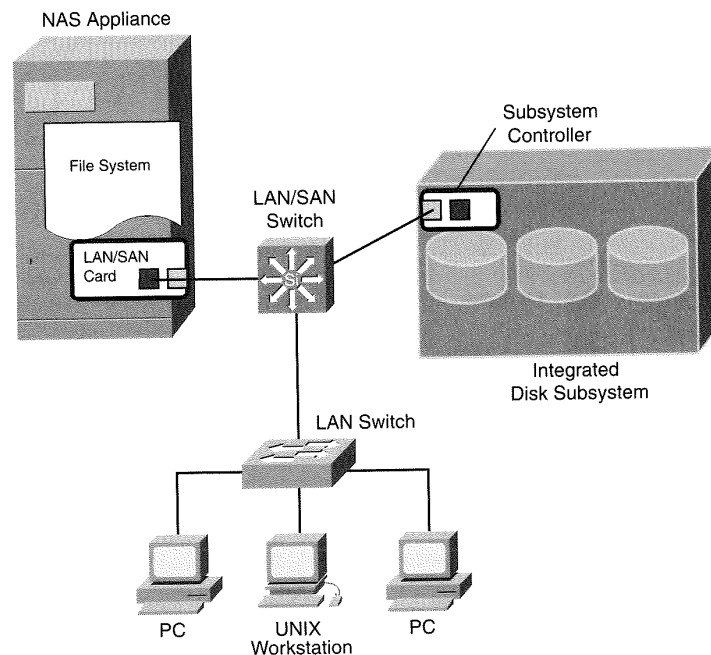


The connecting functions in Figure 2-10 are suddenly much more interesting than in the previous figures. Historically, the connecting technology on the LAN (NAS) side of the drawing has been a different technology from the connecting technology on the SAN side of the drawing.

However, as both filing (NAS) and storing (SAN) functions are at the application layer of the underlying connecting network, there is no reason why the two connecting functions (LAN and SAN) could not use the same technology. In fact, they could even be the same physical network, as shown in Figure 2-11.

NOTE

This example is not intended to suggest that putting SANs and LANs on the same physical network is the preferred way of doing things. But it does help drive home the point that connecting, storing, and filing are, in fact, three independent and orthogonal functions.

Figure 2-11 *A LAN with NAS Connected to a SAN, All Using the Same Connecting Technology*

Summary

This chapter presented readers with the fundamental analysis tools that run throughout the book. As with any complex topic, storage networking has many possible interpretations. It's important to find a stable analysis method that can be used for a wide breadth of products and network designs.

The three functional areas of connecting, storing, and filing can be ascribed to any element of a storage network. It is a powerful analysis, because there is very little overlap between them, which limits confusion to a great extent. It can even be used to clarify the long-standing confusion regarding NAS and SAN.

Another major tool introduced in this chapter was the term *storage address space* as a generalization of all sorts of storage receptacles, including disk drives, disk partitions, disk volumes, virtual disk, exported storage, and LUNs. Using this term simplifies discussions of storage processes immensely without obscuring the nature of how storage works. A storage address space is the element that a filing system uses and manages.

There is a great deal of flexibility in how storage networks can be constructed. The architectural diagrams at the end of the chapter are graphical depictions of the fundamental approaches that can be used. It is hoped that these diagrams will speak volumes to readers who think visually.

Q & A

1. What are the three primary components of storage networking?
2. True or false: Buses guarantee the order of delivery.
3. Which of the following is not part of the storing component: volume management, virtualization, zoning, host bus adapters, or block I/O?
4. True or false: Disk drives do not need a controller.
5. What is the difference between initiator and target controllers?
6. What kinds of storage devices and subsystems create storage address spaces?
7. What are the primary two things filing is responsible for?
8. Explain the different roles of file systems and operating systems.
9. What is the application difference between SAN and NAS?
10. True or false: Storing and filing have to use different networks.

ughout the
etations. It's
products and

ny element of
etween them,
tanding

e as a
tions, disk
scussions of
. A storage