

Storage Management in Data Centers

Volker Herminghaus • Albrecht Scriba
Authors

Storage Management in Data Centers

Understanding, Exploiting, Tuning,
and Troubleshooting
Veritas Storage Foundation

 Springer

Volker Herminghaus
Nieder-Olm
Germany
v.herminghaus@anykey-dcs.de

Dr. Albrecht Scriba
Mainz
Germany
albrecht@albrecht-scriba.de

ISBN 978-3-540-85022-9

e-ISBN 978-3-540-85023-6

DOI 10.1007/978-3-540-85023-6

Library of Congress Control Number: 2009921159

© Springer-Verlag Berlin Heidelberg 2009

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, reuse of illustrations, recitation, broadcasting, reproduction on microfilm or in any other way, and storage in data banks. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permissions for use must always be obtained from Springer-Verlag. Violations are liable for prosecution under the German Copyright Law.

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Cover design: KuenkelLopka GmbH Heidelberg,

Printed on acid-free paper

springer.com

For my wife and children, who taught me what really counts.

Volker Herminghaus

Dedicated to my wife and children.

Dr. Albrecht Scriba

PREFACE

This book is designed to meet the needs of UNIX architects and administrators working in data centers. While it will be useful for the computer science student or the newcomer who has been attracted to volume management by Symantec's release of a free version of its Volume Manager software its focus is on the data center. Most data center applications nowadays handle amounts of data that had been unconceivable at the time when the most commonly used storage media - the hard disk - was developed. As a consequence, the design of the hard disk simply cannot match the requirements posed by current applications. Its physical attributes and limits need to be overcome by additional layers of hardware or software. These layers, if properly designed and thoughtfully applied, convert a set of physical disks to a supply of storage space whose properties better match application requirements. Instead of physical disks with their physical limitations, logical entities known as volumes are now commonly used. These volumes can be fault tolerant, accelerated to the limits, replicated to remote locations, and made almost infinitely large. Additionally, volumes can even be reshaped and their features changed while they are in use, enabling the data center administrator to adapt to changing requirements without suffering an application downtime.

The technical term for this software or hardware layer is "volume management".

Veritas Volume Manager® is the most widely used software for volume management. It is used in data centers all over the world and has proven to be stable and deliver high performance under most circumstances. While there are other volume management software products on the market (e.g. AIX LVM, Sun microsystems' SunVM, or several Linux LVMs), most of them suffer from one or more limitations that hamper their widespread deployment. They are either limited to the manufacturer's operating system or they have less to offer than the Veritas product. In most cases, both is true at the same time. This has led to Veritas Volume Manager, or VxVM in short, being the most widely deployed product on the market, which in turn led to most administrators learning at least the basic skills required for its administration.

However, mastering the basic skills is something quite different from fully understanding a product and making full use of the available power. In data center operations it is imperative that the operators know precisely how things are supposed to work, rather than apply the skills of "experimental computer science". Even today's personal computers are too complex for any kind of experimental approach to solving a problem or finding a solution. This is much more true in data centers, where the motto must be: "If you do not know it, then learn it or leave it, but don't fumble it".

In my time as both an independent data center consultant and an independent trainer for the Veritas product suite I have tried to educate people enough so that they would at least realize what is possible if they could harness VxVM to its full extent. Staying in close

contact with my clients, it dawned on me that what they need is a written guide they can rely on when they actually try some of the more advanced features. If you are responsible for a mission critical application then the last thing you want is to incur a downtime. And with only some diffuse background knowledge and elementary skills left over from the last VxVM training, most of you would rather stick to established procedures than try something new.

My first attempt at writing down what I knew was a training companion book called "Veritas Storage Foundation" published by Springer in 2006 (ISBN: 3-540-34610-4) and endorsed by Switzerland's biggest Symantec partner, Infoniqa SQL AG (www.sql.ch) as their official training material. This book had been written together with Albrecht Scriba, one of the most respected Veritas trainers in Europe. It covered Veritas Volume Manager (VxVM) and Veritas Cluster Server (VCS) and was received very well by the administrators. However, its drawback was that it was written in German, our native tongue, so its distribution was severely limited by the language barrier. Having been approached numerous times by international colleagues I decided to take the next step and write a new, English book that concentrates on VxVM and the Veritas File System (VxFS), again with Albrecht acting as the co-author for some of the toughest chapters. It is not a training companion like the first one but uses a more classical approach. There are many walkthroughs to make you understand what you can do, how you can do it and what exactly is going on inside VxVM and VxFS so you can understand it and repeat it step by step on your own systems. It also holds a large section on troubleshooting that points out how problems can be found and fixed.

So here is your guide that helps you understand - in detail - the principles and the problems of mass storage, volume management, and file systems and how to manage them. It also tries to correct some common misconceptions about storage and UNIX, and highlights the most limiting factors in today's data center environments: anachronistic thinking and the sluggish speed of light!

ABOUT THE AUTHORS

Volker Herminghaus

Born in the stone ages (1963) and raised in a family full of physicists, he studied mechanical engineering and computer science in Darmstadt, Germany. Took some really deep looks at the kernel of AT&T UNIX System V Release 3.2 for his thesis and has been claiming to know what he's talking about since then.

He started computing on a Commodore 64 and switched to Atari ST as it became available, then finally to NeXTSTEP. Deeply enamoured with its elegance and power he has since stuck to descendants of this operating system (MacOS X) for his own use. Professionally, he has been working on Solaris and other UNIX variants as a consultant since the early 1990s. He has just co-founded his second data center consulting company: the **anykey-dcs**.

Dr. Albrecht Scriba

Albrecht studied mathematics and religious science in Mainz until 1998 (including thesis and State doctorate). Being familiar with ancient languages like Aramaic as well as computer programming he hacked the Atari ST's Signum! program in assembler to optimize printing of those languages' fonts. Albrecht has long been working as a Consultant and as a Trainer for Veritas/Symantec. He left Symantec in 2008 and is now working for anykey-dcs, Symantec and other companies as a free lancer. Fallen in love with Unix since his very first encounter, his motto is: "Never type a command twice, write a program for it!"

ACKNOWLEDGEMENTS

A book like this is not created out of thin air. It takes a lot of work, energy, resources, and determination to keep going all the way to the publication. Many people have helped us finish this task and have thus contributed to the successful completion of this book. The first round of thanks goes to the contact persons at Springer: Hermann Engesser and Gabriele Fischer, for providing a perfect office into which to throw all the unstructured suggestions, ideas, questions, and draft versions of this book. But most of all, for saying "YES" before we could even finish our sentence asking if they would like to publish our second book. That immediate and unquestioning positive reply provided the motivation required to kick off the project.

The second round goes to our wives and kids, who always suffer most when fathers decide to dedicate the better part of two years to sitting down late at night hacking, experimenting, and writing.

The third round goes to all the people that gave us gems of background information on storage management: Of these, Ron Karr and particularly Oleg Kiselev, two of the inventors of Veritas Volume Manager and extremely smart people, provided the most insight into VxVM's design ideas and implementation as well as a broad overview about modern storage systems in general.

Sun Microsystems' benchmarking center in Langen, Germany, allowed us to access their powerful Sun servers tied to high-end storage in order to run a multitude of tests and simulations. Special thanks go to all involved at Sun for their efficient, competent and friendly support: Kirsten Prahst, Rüdiger Frenk (who also maintains the only complete Sun hardware museum in the world), and Peter Hausdorf.

A final round of thanks goes to the many hundreds of people who have participated in our trainings or been our consulting clients. They never failed to come up with new questions, setups or problems that kept our brains busy.

TABLE OF CONTENTS

Preface	VII
About the authors	IX
Acknowledgements	XI
1 DISK AND STORAGE SYSTEM BASICS	1
1.1 Overview	1
1.1.1 Storage Hardware Situation and Outlook.	1
1.1.2 Physical Limits	3
1.1.3 Trying to Fix the Problems - and Failing!	7
1.1.4 SAN-Attached Hard Disks.	10
1.1.5 Storage Arrays and LUNs	10
1.1.6 Common Problems	15
1.1.7 Physical Disks vs. LUNs.	17
1.2 Disk Addressing and Layout.	19
1.3 Paths and path redundancy.	23
1.4 The Trouble with Networked Disk Access	30
1.4.1 Summary	36
2 EXPLORING VXVM	39
2.1 Getting Started	39
2.1.1 Hello, Volume!.	40
2.1.2 vxdisksetup: Turning Disks into VM Disks.	40
2.1.3 Disk Groups: Putting VM Disks into Virtual Storage Boxes	42
2.2 The Hard Way: a Low-level Walkthrough	45
2.2.1 Subdisks: Extents for Persistent Backing Store	45
2.2.2 Plexes: Mapping Virtual Extents to Physical Extents	46
2.2.3 Volumes: Virtual Partitions for Any Purpose	48
2.2.4 Volume Start: Prepare for Takeoff	52
2.3 The Easy Way: vxassist	53
2.3.1 Summary	53
3 INCORPORATING DISKS INTO VXVM	55
3.1 Solaris Disk Handling	56
3.1.1 Getting a New Disk into Solaris	56
3.1.2 You Don't Format with "format"	57
3.1.3 Finding New Disks in VxVM	57
3.1.4 What if My New Disk is Not Found?	59
3.1.5 Leaving Physics Behind – Welcome to VxVM!	61

Table Of Contents

3.2	VxVM disk handling	62
3.2.1	VxVM Disk Formats	62
3.2.2	cdsdisk and sliced	63
3.2.3	How to Mix CDS and Sliced Disks in a Disk Group?	66
3.2.4	Other Disk Formats	66
3.2.5	Encapsulation Overview – Integrating Legacy Data.	67
3.2.6	Summary	69
4	DISK GROUPS	71
4.1	Overview	71
4.1.1	What is a Disk Group?	71
4.2	Simple Disk Group Operations	74
4.3	Advanced Disk Group Operations	80
4.3.1	Options for Importing or Exporting a DG	81
4.3.2	Disk Group Operations for Off-Host Processing	83
4.3.3	Miscellaneous Disk Group Operations	85
4.3.4	Summary	87
4.4	Disk Group Implementation Details	89
4.4.1	Major and Minor Numbers for Volumes and Partitions	97
5	VOLUMES	99
5.1	Overview	99
5.1.1	What is a Volume?	99
5.2	Simple Volume Operations	101
5.2.1	Creating, Using and Displaying a Volume.	101
5.2.2	Useful vxprint Flags Explained	103
5.2.3	Starting and Stopping Volumes	105
5.3	Volume Layouts and RAID Levels	106
5.3.1	Volume Features Supported by VxVM	106
5.4	Volume Maintenance	114
5.5	Tuning vxassist Behavior.	120
5.5.1	Storage Attributes – Specifying Allocation Strategies	120
5.5.2	Skipping Initial Mirror Synchronisation	126
5.5.3	Changing the Layout of a Volume	127
5.6	Methods of Synchronisation	130
5.6.1	Atomic Copy	131
5.6.2	Read-Writeback, Schrödinger's Cat, and Quantum Physics	132
5.7	Volume Features in Detail	137
5.7.1	concat	137
5.7.2	stripe	137
5.7.3	mirror	139
5.7.4	RAID-4 and RAID-5.	142
5.7.5	mirror-concat	146

5.7.6	mirror-stripe	146
5.7.7	Mixed Layouts	146
5.8	Relayout in Detail	147
6	LAYERED VOLUMES	153
6.1	Overview	153
6.1.1	Why Use Layered Volumes?	153
6.2	Introducing Layered Volumes	158
6.2.1	concat-mirror	160
6.2.2	stripe-mirror	161
6.2.3	Understanding vxprint Output for Layered Volumes	162
6.3	Understanding Layered Volumes	165
6.3.1	Manually Creating a Layered Volume	165
6.3.2	Mirroring RAID-5 Volumes	169
7	LOGS	173
7.1	Overview	173
7.1.1	What is a Log?	174
7.1.2	Simple Log Operations	175
7.2	Log Maintenance	177
7.3	Details About Logs	180
7.3.1	DRL (Dirty Region Log)	180
7.3.2	DCL/DCO (Data Change Log / Data Change Object)	184
7.3.3	raid5log	188
8	DUAL DATA CENTERS	191
8.1	Volume Management in Dual Data Centers	191
8.1.1	Growing a Mirrored Volume Across Sites	192
8.1.2	Growing Existing Volumes Across Sites	196
8.1.3	Mirroring Site-Aware Volumes Across Sites	204
8.1.4	Summary	212
8.2	Replication Across Data Centers	213
8.2.1	Replication vs. Mirroring	213
8.2.2	The Speed of Light and Latency	214
8.2.3	Replication Using Storage Array Logic	217
8.2.4	Replication Using Kernel Mode Logic	220
8.3	Estimating Replication Speed	223
9	POINT IN TIME COPIES (SNAPSHOTS)	233
9.1	Overview	233
9.1.1	Types of Snapshots	233
9.1.2	Consistency Problems for Snapshots	235
9.2	Physical Raw Device Snapshots	237

Table Of Contents

9.2.1	Overview	237
9.2.2	A Look at What Goes on Inside	238
9.2.3	A Logical File System Snapshot	245
9.3	Features of and Improvements on the Raw Device Snapshot	249
9.3.1	Snapshot Region Logging by the Data Change Log	249
9.3.2	Reverting the Resynchronization Direction	253
9.3.3	The Snap Objects	254
9.3.4	Clearing the Snapshot Relation	256
9.3.5	Deleting the Snapshot	257
9.3.6	Offhost Processing	258
9.3.7	Full Sized Volume Based Instant Snapshots	262
9.3.8	Snapshot Refresh	267
9.3.9	Space Optimized Volume Based Instant Snapshots	268
9.3.10	Autogrow Related Attributes	274
9.3.11	Cascading Snapshots	278
9.3.12	A Final Example for Volume Snapshots	279
9.4	Veritas File System Based Snapshots	282
9.4.1	Cache Overflow on a Traditional Snapshot	282
9.4.2	VxFS Storage Checkpoints	286
9.5	Creating a Full Sized Volume Snapshot Using Low-Level Commands	300
9.6	Legacy Snapshot Commands	303
9.6.1	Full Sized Snapshot without FMR	303
9.6.2	Full Sized Snapshot with Kernel Based FMR	306
9.6.3	Full Sized Snapshot with DCL Volume Based FMR Version 0	307
9.7	DCO Version 0 and Version 20	308
9.8	VxFS Storage Checkpoint Behavior	313
10	ENCAPSULATION AND ROOT MIRRORING	319
10.1	Introduction and Overview	319
10.2	The Secrets of Encapsulation	321
10.3	Root Disk Encapsulation	323
10.4	Root Disk Mirroring	324
10.5	Remarks to vxencap and OS Mirroring	327
10.6	The Ghost Subdisk	330
10.7	Manual Encapsulation Walkthrough	338
10.7.1	Assumptions and Prerequisites	338
10.7.2	Basic Considerations	339
10.7.3	Storing the Disk Layout	340
10.7.4	Defining Private and Public Region	340
10.7.5	Creating Subdisks, Plexes, and Volumes	341
10.7.6	Mirroring and Preparing for CDS	345
10.7.7	Converting to CDS	348

11	TROUBLESHOOTING	349
11.1	Introduction	349
11.2	Disk Outage	352
11.2.1	Disk Permanently damaged	355
11.2.2	Disk Temporarily Unavailable	358
11.2.3	Replacing an OS Disk	359
11.3	Disk Outage in Detail	362
11.3.1	A Complete Disk Array Temporarily Unavailable	362
11.3.2	A Disk Group Temporarily Inaccessible	363
11.3.3	A Partially Failed Disk ("Failing")	365
11.3.4	Hot Relocation	367
11.3.5	Hot Spare	374
11.4	Synchronization Tasks	380
11.4.1	Optimizing Resynchronization	380
11.4.2	Controlling Synchronization Behavior	383
11.5	Restore of Lost VxVM Objects	391
11.5.1	vxprint and vxmake Capabilities	391
11.5.2	Restore of All Volumes in a Disk Group	392
11.5.3	Restore of Some Volumes in a Disk Group	393
11.5.4	Restore of the Entire Disk Group Configuration	394
11.5.5	Restore of a Destroyed Disk Group	398
11.5.6	Serial Split Brain of a Disk Group	401
11.6	Booting without VxVM	406
11.7	More than Two OS Mirrors: Emergency Disk	412
11.8	Hot Relocation Troubles	420
11.8.1	Plex Synchronization Skipped	420
11.8.2	Unrelocation of Split Subdisks	424
11.9	Plex States Overview	426
12	FILE SYSTEMS	429
12.1	Block Based File Systems	429
12.1.1	Just for Fun: Commodore 64's Rudimentary File Access	430
12.1.2	FAT – Not a Big Improvement	430
12.1.3	UFS – Finally Something Decent	432
12.2	Extent Based File Systems	434
12.2.1	VxFS	434
12.3	Advanced File System Operations	441
12.3.1	Summary	445
13	TUNING STORAGE FOUNDATION	447
13.1	Basics About Tuning Storage Foundation	447
13.1.1	Tuning VxVM by Using Reasonable Parameters	449
13.1.2	Understanding and Modifying VxVM Defaults	451

Table Of Contents

13.1.3	Tuning VxFS	454
13.2	Tools for Performance Tuning VxVM on SAN Storage	461
13.3	Performance Tuning	468
13.3.1	Overview and Disclaimer	468
13.3.2	Identifying Performance and Performance Requirements	468
13.3.3	Comparative Benchmarks of Various Volume Layouts	473
13.3.4	Summary	477
14	MISCELLANEOUS	479
14.1	Disk Flags	479
14.1.1	Summary	485
15	STORAGE FOUNDATION SOFTWARE STACK	487
15.1	Software Overview	487
15.1.1	Structure of Storage Foundation Components	488
15.2	Kernel Space Drivers	491
15.3	User Space Processes	494
15.4	Reducing VxVM's Footprint	495
15.4.1	Essential VxVM Processes	496
15.4.2	Unessential VxVM Processes	496
15.4.3	Potentially Undesirable VxVM Processes	497
16	INDEX.	501