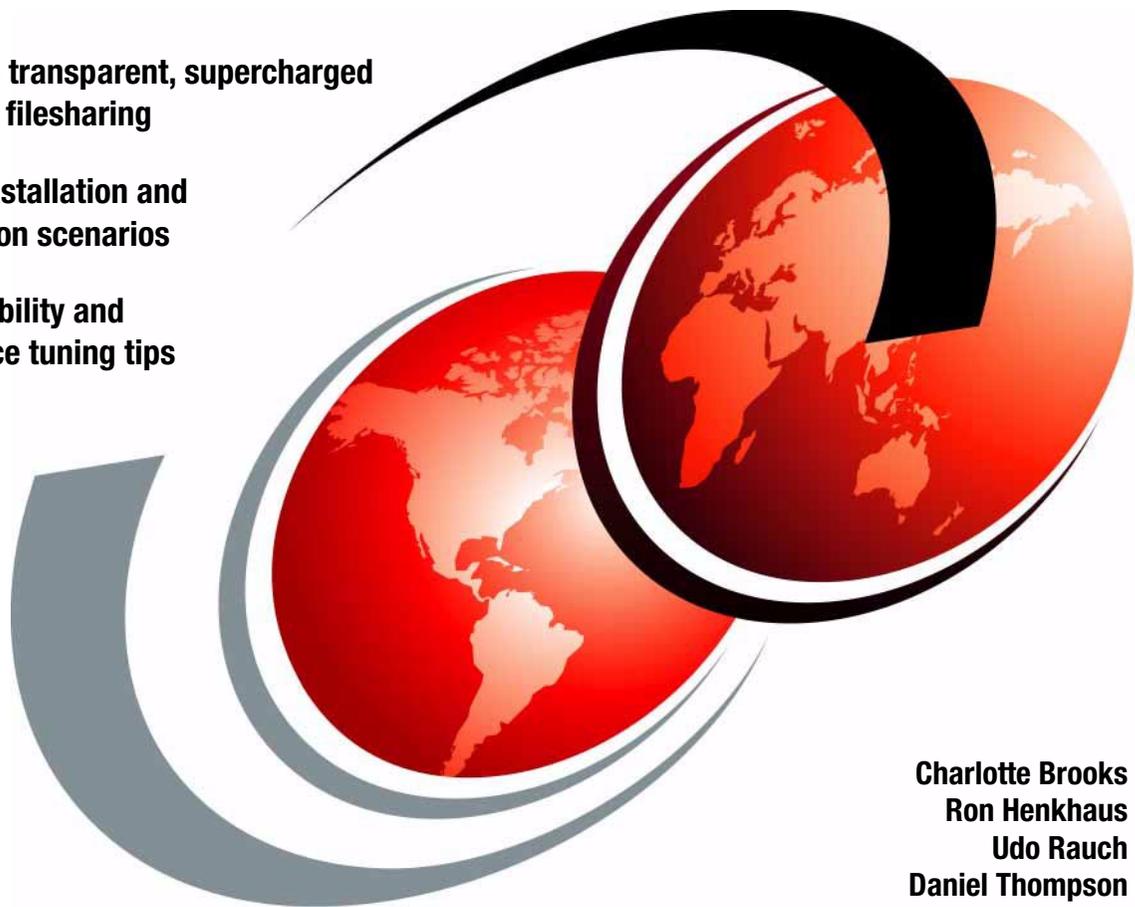


# A Practical Guide to Tivoli SANergy

Application transparent, supercharged  
SAN-based filesharing

Practical installation and  
configuration scenarios

High availability and  
performance tuning tips



Charlotte Brooks  
Ron Henkhaus  
Udo Rauch  
Daniel Thompson

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

**Redbooks**





International Technical Support Organization

**A Practical Guide to Tivoli SANergy**

June 2001

**Take Note!**

Before using this information and the product it supports, be sure to read the general information in Appendix E, "Special notices" on page 291.

**First Edition (June 2001)**

This edition applies to Version 2.2 and later of Tivoli SANergy, 5698-SFS for use with the Windows 2000, Windows NT, Sun Solaris and Linux (plus all SANergy host platforms) Operating Systems.

Comments may be addressed to:

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

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

**© Copyright International Business Machines Corporation 2001. All rights reserved.**

Note to U.S Government Users – Documentation related to restricted rights – Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

# Contents

|  |      |
|--|------|
| <b>Figures</b> .....   | vii  |
| <b>Preface</b> .....   | xiii |
| The team that wrote this redbook .....                                 | xiii |
| Comments welcome .....   | xv   |
| <hr/>  |      |
| <b>Part 1. SANergy basics</b> .....                                    | 1    |
| <b>Chapter 1. Introduction to SANergy</b> .....                        | 3    |
| 1.1 Who should read this book? .....                                   | 3    |
| 1.2 New storage technology .....                                       | 4    |
| 1.3 Current SAN challenges .....                                       | 5    |
| 1.4 Sharing data on a SAN .....  | 7    |
| 1.5 Tivoli SANergy .....   | 9    |
| 1.5.1 What are the benefits of SANergy? .....                          | 10   |
| 1.5.2 SANergy limitations .....  | 14   |
| 1.6 Summary .....  | 15   |
| 1.7 SANergy requirements .....   | 15   |
| <b>Chapter 2. SANergy with a Windows MDC</b> .....                     | 19   |
| 2.1 File sharing setup .....   | 19   |
| 2.1.1 Windows server file sharing .....                                | 21   |
| 2.1.2 Client file sharing .....  | 33   |
| 2.2 Installing and configuring the MDC on Windows .....                | 37   |
| 2.3 Installing and configuring SANergy hosts on UNIX and Windows ..... | 44   |
| 2.3.1 Installing and configuring SANergy hosts on Windows .....        | 45   |
| 2.3.2 Installing and configuring SANergy hosts on UNIX .....           | 52   |
| <b>Chapter 3. SANergy with a UNIX MDC</b> .....                        | 57   |
| 3.1 File sharing setup .....   | 57   |
| 3.1.1 UNIX server file sharing .....                                   | 58   |
| 3.1.2 Client file sharing .....  | 66   |
| 3.2 UNIX MDC installation and configuration .....                      | 70   |
| 3.2.1 SANergy base installation .....                                  | 71   |
| 3.2.2 SANergy patch installation .....                                 | 71   |
| 3.2.3 Linux system considerations .....                                | 71   |
| 3.2.4 Solaris system considerations .....                              | 72   |
| 3.2.5 MDC configuration .....  | 72   |
| 3.3 SANergy host installation and configuration .....                  | 74   |
| 3.3.1 Installing and configuring SANergy hosts on UNIX .....           | 74   |
| 3.3.2 Installing and configuring SANergy hosts on Windows .....        | 80   |

|  |     |
|--|-----|
| <b>Part 2. SANergy advanced topics</b> . . . . .                         | 87  |
| <b>Chapter 4. Performance</b> . . . . .                                  | 89  |
| 4.1 General performance considerations . . . . .                         | 89  |
| 4.1.1 UNIX NFS, LAN tuning, and performance problem diagnosis . . . . .  | 89  |
| 4.1.2 Operating system tools and considerations. . . . .                 | 93  |
| 4.2 SANergy performance parameters . . . . .                             | 96  |
| 4.2.1 Cache settings . . . . .   | 97  |
| 4.2.2 Hyperextension . . . . .   | 101 |
| 4.2.3 Minimum fused file size . . . . .                                  | 102 |
| 4.2.4 Fusion exclusion list . . . . .                                    | 103 |
| 4.2.5 Logging . . . . .  | 103 |
| 4.2.6 Datagram threads (dgram) parameter. . . . .                        | 104 |
| 4.2.7 Error handling. . . . .  | 104 |
| <b>Chapter 5. Advanced SANergy configuration</b> . . . . .               | 105 |
| 5.1 MSCS clustering for the MDC. . . . .                                 | 105 |
| 5.1.1 Base configuration . . . . .                                       | 106 |
| 5.1.2 How to integrate SANergy and MSCS. . . . .                         | 107 |
| 5.1.3 Installing and configuring SANergy with MSCS. . . . .              | 108 |
| 5.2 Sharing Windows Dynamic Disks and Stripe Sets . . . . .              | 132 |
| 5.2.1 SANergy and Windows NT Stripe Sets . . . . .                       | 134 |
| 5.2.2 SANergy and Windows 2000 Dynamic Disks . . . . .                   | 150 |
| 5.2.3 SANergy Performance on Stripe Sets and Dynamic Disks . . . . .     | 160 |
| <b>Chapter 6. Putting it all together</b> . . . . .                      | 169 |
| 6.1 High availability . . . . .  | 171 |
| 6.2 High performance. . . . .  | 173 |
| 6.3 Scalable file server farm. . . . .                                   | 176 |
| 6.4 Web serving . . . . .  | 179 |
| 6.5 Databases on SANergy hosts. . . . .                                  | 180 |
| 6.5.1 Creating a database on a mapped network drive . . . . .            | 183 |
| 6.5.2 Migrating an existing database from local to shared disk . . . . . | 186 |
| 6.6 Conclusion . . . . .   | 187 |
| <b>Part 3. SANergy and other Tivoli applications</b> . . . . .           | 189 |
| <b>Chapter 7. SANergy and Tivoli Storage Manager</b> . . . . .           | 191 |
| 7.1 Application server-free backup and restore . . . . .                 | 191 |
| 7.1.1 Client implementation and testing without SANergy . . . . .        | 193 |
| 7.1.2 Tivoli Storage Manager client with SANergy . . . . .               | 200 |
| 7.2 SANergy managed backup sets . . . . .                                | 205 |
| 7.2.1 Tivoli Storage Manager server setup . . . . .                      | 206 |

|   |  |            |
|---|--|------------|
| 7.2.2   | Defining the backup set device class . . . . .               | 208        |
| 7.2.3   | Backup set generation . . . . .                              | 208        |
| 7.2.4   | Backup sets as seen by the server . . . . .                  | 215        |
| 7.2.5   | Restoring from a backup set . . . . .                        | 217        |
| 7.3   | Commercial database backup from SANergy host . . . . .       | 223        |
| 7.4   | Heterogeneous SANergy backup configurations . . . . .        | 225        |
| <b>Chapter 8. SANergy and OTG DiskXtender 2000 . . . . .</b>                  |  | <b>227</b> |
| 8.1   | Configuring the Tivoli Storage Manager server . . . . .      | 229        |
| 8.2   | Installing the client and API on the SANergy MDC . . . . .   | 231        |
| 8.3   | Installing OTG DiskXtender 2000 on the SANergy MDC . . . . . | 231        |
| 8.4   | Preparing for LAN-free data transfer. . . . .                | 253        |
| 8.4.1   | Installing the Tivoli Storage Agent . . . . .                | 254        |
| 8.4.2   | Configuring the Tivoli Storage Agent . . . . .               | 254        |
| 8.4.3   | Communication between server and Storage Agent . . . . .     | 255        |
| 8.4.4   | Creating the drive mapping. . . . .                          | 257        |
| 8.4.5   | Install Storage Agent as a service . . . . .                 | 258        |
| 8.5   | LAN-free migration . . . . .                                 | 259        |
| <b>Appendix A. Linux with a SAN and SANergy . . . . .</b>                     |  | <b>261</b> |
| A.1   | Linux on a SAN . . . . .                                     | 261        |
| A.1.1   | Helpful Web sites . . . . .                                  | 262        |
| A.1.2   | Linux devices . . . . .                                      | 262        |
| A.1.3   | Linux device drivers for Fibre Channel HBAs . . . . .        | 263        |
| A.1.4   | Accessing the SCSI disks . . . . .                           | 265        |
| A.1.5   | SANergy and the SCSI disks . . . . .                         | 266        |
| A.2   | Using SANergy on Linux. . . . .                              | 267        |
| <b>Appendix B. Windows server drive mapping and SANergy . . . . .</b>         |  | <b>271</b> |
| B.1   | Detailed considerations. . . . .                             | 271        |
| B.1.1   | Access to the drive . . . . .                                | 272        |
| B.1.2   | What type of share? . . . . .                                | 272        |
| B.1.3   | When to map drives . . . . .                                 | 272        |
| B.1.4   | Connection persistence . . . . .                             | 272        |
| B.1.5   | Server access to the mapped drive . . . . .                  | 273        |
| B.2   | A procedure for mapping a drive to a server. . . . .         | 273        |
| B.3   | Sample source listings . . . . .                             | 275        |
| B.3.1   | userexit.c listing . . . . .                                 | 276        |
| B.3.2   | userexit.h listing . . . . .                                 | 280        |
| <b>Appendix C. Using ZOOM to improve small file I/O performance . . . . .</b> |  | <b>283</b> |
| C.1   | Background. . . . .  | 283        |
| C.2   | ZOOM questions and answers . . . . .                         | 283        |
| C.2.1   | What does ZOOM do? . . . . .                                 | 283        |

|  |   |            |
|--|---|------------|
| C.2.2  | Can files still be written on a ZOOM machine? . . . . .               | 283        |
| C.2.3  | Will all files opened for “read” be accelerated by ZOOM? . . . . .    | 283        |
| C.2.4  | How does ZOOM work? . . . . .   | 284        |
| C.2.5  | Which SANergy MDCs support ZOOM? . . . . .                            | 284        |
| C.2.6  | What does the ZOOM server do? . . . . .                               | 284        |
| C.2.7  | Will ZOOM work with HBA failover or dual paths? . . . . .             | 284        |
| C.2.8  | Will ZOOM work with the Veritas Volume manager? . . . . .             | 285        |
| C.2.9  | Will ZOOM work with other file systems? . . . . .                     | 285        |
| C.2.10   | How fast is ZOOM? . . . . .   | 285        |
| C.2.11   | What are the exact steps that occur when a file is changed? . . . . . | 285        |
| C.2.12   | What happens if the same file changes frequently? . . . . .           | 286        |
| C.2.13   | What is saturation? . . . . .   | 286        |
| C.2.14   | Can ZOOM ignore some directories that change too often? . . . . .     | 286        |
| C.2.15   | What is an inappropriate type of system usage for ZOOM? . . . . .     | 286        |
| C.2.16   | What is the best way to start with ZOOM and to test it? . . . . .     | 287        |
| C.2.17   | What configurations are appropriate? . . . . .                        | 288        |
| <b>Appendix D. Using the additional material . . . . .</b> |   | <b>289</b> |
| D.1  | Locating the additional material on the Internet . . . . .            | 289        |
| D.2  | Using the Web material . . . . .                                      | 289        |
| D.2.1  | System requirements for downloading the Web material . . . . .        | 289        |
| D.2.2  | How to use the Web material . . . . .                                 | 289        |
| <b>Appendix E. Special notices . . . . .</b>               |   | <b>291</b> |
| <b>Appendix F. Related publications . . . . .</b>          |   | <b>295</b> |
| F.1  | IBM Redbooks . . . . .  | 295        |
| F.2  | IBM Redbooks collections . . . . .                                    | 296        |
| F.3  | Other resources . . . . .   | 296        |
| F.4  | Referenced Web sites . . . . .  | 297        |
| <b>How to get IBM Redbooks . . . . .</b>                   |   | <b>299</b> |
| IBM Redbooks fax order form . . . . .                      |   | 300        |
| <b>Abbreviations and acronyms . . . . .</b>                |   | <b>301</b> |
| <b>Index . . . . .</b>                                     |   | <b>303</b> |
| <b>IBM Redbooks review . . . . .</b>                       |   | <b>307</b> |

## Figures

|  |     |
|--|-----|
| 1. The authors (left to right): Ron, Charlotte, Udo, and Dan . . . . .       | xiv |
| 2. Traditional storage architecture versus Storage Area Networks . . . . .   | 4   |
| 3. Sharing storage devices with Storage Area Networks . . . . .              | 7   |
| 4. Sharing of volumes for homogeneous operating systems . . . . .            | 8   |
| 5. Sharing volumes from heterogeneous hosts with SANergy . . . . .           | 10  |
| 6. Locally attached storage compared to exploiting a NAS appliance . . . . . | 11  |
| 7. Sharing files with SANergy and Storage Area Networks . . . . .            | 14  |
| 8. Example of SANergy configuration . . . . .                                | 17  |
| 9. File sharing environment with Windows file server . . . . .               | 21  |
| 10. Sharing option in explorer view . . . . .                                | 22  |
| 11. File sharing properties . . . . .  | 23  |
| 12. Share icon for exported partition . . . . .                              | 24  |
| 13. Control panel showing Maestro NFS . . . . .                              | 25  |
| 14. NFS Maestro server general configuration . . . . .                       | 26  |
| 15. NFS Maestro Server Status . . . . .                                      | 27  |
| 16. NFS Maestro server name mapping . . . . .                                | 28  |
| 17. NFS Maestro Exported File Systems . . . . .                              | 29  |
| 18. NFS Current Exports . . . . .  | 29  |
| 19. NFS sharing with SFU . . . . .   | 30  |
| 20. Granting permissions to hosts with SFU NFS . . . . .                     | 31  |
| 21. User mapping administration . . . . .                                    | 32  |
| 22. Map a shared network drive . . . . .                                     | 33  |
| 23. Type UNC path to map shared network drive . . . . .                      | 34  |
| 24. Access to a new share . . . . .  | 35  |
| 25. The linuxconf utility . . . . .  | 36  |
| 26. Disk Management on Windows 2000 MDC . . . . .                            | 37  |
| 27. Initial SANergy installation screen . . . . .                            | 38  |
| 28. Select components to install . . . . .                                   | 39  |
| 29. Select managed buses for Windows MDC . . . . .                           | 40  |
| 30. Device assignment for Windows MDC . . . . .                              | 41  |
| 31. Volume assignment for Windows MDC . . . . .                              | 42  |
| 32. Performance Tester tab on Windows MDC . . . . .                          | 43  |
| 33. SANergy environment with a Windows MDC . . . . .                         | 45  |
| 34. Disk Administrator tool before attaching the host to the SAN . . . . .   | 46  |
| 35. Disk Administrator tool showing a new device . . . . .                   | 47  |
| 36. Disk Administrator after deleting drive letter . . . . .                 | 48  |
| 37. Device assignment to a SANergy host . . . . .                            | 49  |
| 38. Volume Assignment on the SANergy host . . . . .                          | 50  |
| 39. Verify installation with the performance tester . . . . .                | 51  |
| 40. Tivoli SANergy main screen on Linux . . . . .                            | 53  |

|  |     |
|--|-----|
| 41. Mount point before fusing . . . . .  | 54  |
| 42. Mount point which is fused. . . . .  | 54  |
| 43. Performance Testing and Statistics panel . . . . .                           | 54  |
| 44. UNIX server file sharing configuration . . . . .                             | 58  |
| 45. Samba swat home page . . . . .   | 62  |
| 46. Samba global settings . . . . .  | 63  |
| 47. Samba share creation . . . . .   | 64  |
| 48. Samba configuration summary . . . . .  | 65  |
| 49. Samba server status . . . . .  | 66  |
| 50. Map network drive on Windows systems. . . . .                                | 69  |
| 51. Access to a new share from a UNIX system . . . . .                           | 70  |
| 52. UNIX SANergy file sharing configuration. . . . .                             | 71  |
| 53. Create signature on new device . . . . .                                     | 81  |
| 54. UNIX formatted disk in the Disk Administrator view. . . . .                  | 81  |
| 55. Select Managed Buses dialog box. . . . .                                     | 82  |
| 56. Device Assignment on the Windows host . . . . .                              | 83  |
| 57. Volume Assignment on the Windows host. . . . .                               | 84  |
| 58. Verify installation with the performance tester on the Windows host. . . . . | 85  |
| 59. Windows 2000 disk defragmenter analysis report . . . . .                     | 95  |
| 60. Options tab on the SANergy Setup Tool on Windows platforms . . . . .         | 96  |
| 61. Configuration tool on UNIX systems . . . . .                                 | 97  |
| 62. Cache settings with high cache line size . . . . .                           | 99  |
| 63. Cache settings with smaller cache line size. . . . .                         | 100 |
| 64. Retry panel on the option tab. . . . .                                       | 104 |
| 65. Testing cluster configuration . . . . .                                      | 106 |
| 66. Removing previous MSCS installation. . . . .                                 | 109 |
| 67. Validating that machine can access SAN disks. . . . .                        | 110 |
| 68. Assigning drive letter to volume . . . . .                                   | 111 |
| 69. Install MSCS . . . . .   | 112 |
| 70. Identifying managed disks . . . . .  | 113 |
| 71. Define quorum disk . . . . .   | 114 |
| 72. Install Tivoli SANergy file sharing . . . . .                                | 115 |
| 73. Device assignment — defaults are acceptable . . . . .                        | 115 |
| 74. Special names on volume assignment . . . . .                                 | 116 |
| 75. Tivoli SANergy MSCS install menu . . . . .                                   | 117 |
| 76. Installing SANergy MSCS on cluster node other than final node. . . . .       | 117 |
| 77. Installing SANergy MSCS on final node in a cluster . . . . .                 | 118 |
| 78. Last window of SANergy MSCS install on final cluster node. . . . .           | 118 |
| 79. Validate that SANergy Volume resource type is available. . . . .             | 119 |
| 80. Adding a new cluster resource. . . . .                                       | 120 |
| 81. Defining a new resource . . . . .  | 121 |
| 82. Identifying which nodes can support the resource. . . . .                    | 122 |
| 83. Identifying resource dependencies . . . . .                                  | 123 |

|   |     |
|---|-----|
| 84. Setting SANergy Volume resource parameters . . . . .                        | 124 |
| 85. Bringing SANergy Volume resources online . . . . .                          | 125 |
| 86. Defining a File Share resource . . . . .                                    | 126 |
| 87. Making File Share dependent upon SANergy Volume resource . . . . .          | 127 |
| 88. Defining CIFS share parameters . . . . .                                    | 128 |
| 89. Map like any other CIFS share . . . . .                                     | 129 |
| 90. Validating the installation using SANergy Setup . . . . .                   | 130 |
| 91. Move MSCS group to test failover . . . . .                                  | 131 |
| 92. MSCS group going offline on current node . . . . .                          | 131 |
| 93. MSCS group online on the other node. . . . .                                | 131 |
| 94. Windows NT Stripe Set testing configuration. . . . .                        | 134 |
| 95. Create a Stripe Set using Disk Administrator. . . . .                       | 136 |
| 96. Commit changes to disk configuration . . . . .                              | 137 |
| 97. Format Stripe Set. . . . .  | 138 |
| 98. Format parameters . . . . .   | 139 |
| 99. Saving the MDC's disk configuration . . . . .                               | 140 |
| 100. Defined a CIFS share for the Stripe Set . . . . .                          | 141 |
| 101. Set ownership of devices in Stripe Set to the MDC. . . . .                 | 142 |
| 102. Assign the volume to the MDC . . . . .                                     | 143 |
| 103. View and document your disk configuration . . . . .                        | 144 |
| 104. Restore the disk configuration. . . . .                                    | 145 |
| 105. Disk configuration after restore; Disk 0 definition is now wrong . . . . . | 146 |
| 106. Disk configuration after restore and manual update . . . . .               | 147 |
| 107. Mount the CIFS share . . . . .   | 148 |
| 108. Dynamic disk testing configuration . . . . .                               | 150 |
| 109. Disk Management applet before creating Dynamic Disk. . . . .               | 152 |
| 110. Begin conversion of basic disks to Dynamic Disk . . . . .                  | 152 |
| 111. Select disks to upgrade. . . . .   | 153 |
| 112. Initiate the creation of a volume on the Dynamic Disks. . . . .            | 153 |
| 113. Identify which disks to use in creating the new volume. . . . .            | 154 |
| 114. Identify Striped Volume as the format . . . . .                            | 154 |
| 115. Format volume . . . . .  | 155 |
| 116. Validate the parameters and then proceed . . . . .                         | 155 |
| 117. Progress of a format process . . . . .                                     | 156 |
| 118. Importing a foreign Dynamic Disk to the host machines . . . . .            | 156 |
| 119. Select the set of disks to be imported . . . . .                           | 157 |
| 120. Validate that the correct volume will be imported . . . . .                | 157 |
| 121. Changing the drive letter of imported volume . . . . .                     | 158 |
| 122. Removing the drive letter to prevent data corruption . . . . .             | 158 |
| 123. Device ownership settings. . . . .   | 159 |
| 124. Volume assignment settings . . . . .                                       | 159 |
| 125. Performance Test of SANergy MDC . . . . .                                  | 161 |
| 126. SANergy performance test from host . . . . .                               | 162 |

|   |     |
|---|-----|
| 127.Simultaneous performance test — view of MDC . . . . .                     | 163 |
| 128.Simultaneous performance test — view of SANergy host. . . . .             | 164 |
| 129.Perfmon report of single 40MB/sec data stream . . . . .                   | 165 |
| 130.Perfmon report of multiple 20MB/sec data streams — MDC view . . . . .     | 165 |
| 131.Perfmon report of multiple 20MB/sec data streams — SANergy host . . . . . | 166 |
| 132.Super data server . . . . .   | 170 |
| 133.RAID5 configuration without SANergy . . . . .                             | 172 |
| 134.RAID5 with SANergy . . . . .  | 173 |
| 135.Storage Hardware (striped volumes). . . . .                               | 174 |
| 136.Stripe configuration with Dynamic Disk . . . . .                          | 175 |
| 137.Performance of a large filesystem. . . . .                                | 176 |
| 138.Re-sharing an imported NFS mount via Samba . . . . .                      | 178 |
| 139.Web site served by the Linux SANergy host rainier . . . . .               | 180 |
| 140.Database servers running in a conventional manner . . . . .               | 181 |
| 141.Database servers in a SANergy environment . . . . .                       | 182 |
| 142.SQL Server Enterprise Manager . . . . .                                   | 183 |
| 143.SQL Server Properties . . . . .   | 184 |
| 144.Startup parameters . . . . .  | 184 |
| 145.Properties of created database . . . . .                                  | 186 |
| 146.Application server-free backup configuration . . . . .                    | 192 |
| 147.Tivoli Storage Manager client GUI for backup. . . . .                     | 194 |
| 148.Backup data flow without SANergy . . . . .                                | 195 |
| 149.Backup results without SANergy . . . . .                                  | 196 |
| 150.TSMtest directory properties . . . . .                                    | 197 |
| 151.TSMtest file properties . . . . .   | 198 |
| 152.Client GUI for restore . . . . .  | 199 |
| 153.Restore results without SANergy . . . . .                                 | 200 |
| 154.SANergy setup on the client . . . . .                                     | 201 |
| 155.Backup data flow with SANergy . . . . .                                   | 202 |
| 156.Backup results with SANergy . . . . .                                     | 203 |
| 157.Restore results with SANergy . . . . .                                    | 204 |
| 158.Backup set configuration. . . . .   | 206 |
| 159.Tivoli Storage Manager server mapped drive . . . . .                      | 207 |
| 160.Windows GUI for backup set generation. . . . .                            | 209 |
| 161.Windows file system backup results . . . . .                              | 210 |
| 162.SANergy statistics before generate backup set . . . . .                   | 212 |
| 163.SANergy statistics after generate backup set . . . . .                    | 213 |
| 164.Backup set restore data flow . . . . .                                    | 218 |
| 165.Windows backup set file selection. . . . .                                | 219 |
| 166.Windows backup set restore . . . . .                                      | 220 |
| 167.Windows next backup set volume . . . . .                                  | 220 |
| 168.Windows backup set restore statistics . . . . .                           | 221 |
| 169.Backing up databases with SANergy . . . . .                               | 224 |

|  |     |
|--|-----|
| 170.Space management in a SANergy environment . . . . .        | 228 |
| 171.Installation program for DiskXtender 2000 . . . . .        | 232 |
| 172.DiskXtender Installation Options . . . . .                 | 232 |
| 173.DiskXtender license type. . . . .                          | 233 |
| 174.Choose target computers . . . . .                          | 234 |
| 175.Setup complete . . . . .                                   | 234 |
| 176.No media services currently configured . . . . .           | 235 |
| 177.Configure Media Services Type . . . . .                    | 235 |
| 178.TSM Information . . . . .                                  | 236 |
| 179.TSM Media Service for DiskXtender . . . . .                | 237 |
| 180.Media Service Properties . . . . .                         | 237 |
| 181.Create TSM Media . . . . .                                 | 238 |
| 182.Media list on Media Service Properties. . . . .            | 239 |
| 183.No extended drives . . . . .                               | 239 |
| 184.New Extended Drive screen . . . . .                        | 240 |
| 185.Select Drive for extension . . . . .                       | 241 |
| 186.Assign Media To Extended Drive . . . . .                   | 242 |
| 187.Settings for DiskXtender operations . . . . .              | 243 |
| 188.DiskXtender Scheduler . . . . .                            | 244 |
| 189.Drive Scan scheduling. . . . .                             | 244 |
| 190.Options for extended drive . . . . .                       | 245 |
| 191.No media folder. . . . .                                   | 246 |
| 192.Create Media Folder . . . . .                              | 246 |
| 193.DiskXtender Administrator main window. . . . .             | 247 |
| 194.Select Media to Add . . . . .                              | 248 |
| 195.New Move Group . . . . .                                   | 249 |
| 196.Select Move Group Media. . . . .                           | 250 |
| 197.Move Rules definition for folder. . . . .                  | 251 |
| 198.Migrated files in the explorer view. . . . .               | 252 |
| 199.Tivoli Storage Manager service log on properties . . . . . | 274 |



## **Preface**

Providing shared access to files is an important aspect of today's computing environment, as it allows easier control and consolidation of data and other assets and enhances information flow through an organization. Traditionally filesharing has been done over the traditional TCP/IP LAN. Storage Area Networks (or SANs) provide the ability for storage to be accessed and moved across a separate dedicated high-speed network. Tivoli SANergy transparently brings these two concepts together, providing all the benefits of LAN-based filesharing at the speed of the SAN.

This IBM Redbook provides an introduction to Tivoli SANergy in various environments, including various flavors of UNIX, Microsoft Windows NT, and Windows 2000. It covers installation and setup of the product, with advice and guidance on tuning and performance. It also describes integrating SANergy with other products, including Tivoli Storage Manager and Microsoft Cluster Services.

This book is written for IBM, Tivoli, customer, vendor, and consulting personnel who wish to gain an understanding of the Tivoli SANergy product and how best to use it in their environments. We assume a basic understanding of filesharing concepts in various operating system environments, as well as SAN concepts and implementation.

---

### **The team that wrote this redbook**

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

**Charlotte Brooks** is a Project Leader for Tivoli Storage Manager. She has 11 years of experience with IBM in the fields of RISC System/6000 and Storage Management. She has written five redbooks and teaches IBM classes worldwide in all areas of Storage Management. Before joining the ITSO a year ago, Charlotte was the Technical Support Manager for Tivoli Storage Management in the Asia Pacific region.

**Ron Henkhaus** is an IBM Certified IT Specialist at IBM Storage Systems Advanced Technical Support in Kansas City. His areas of expertise include Tivoli SANergy, Tivoli Storage Manager, and other Tivoli storage products. Ron has 9 years of experience in field technical support; 4 years in consulting and implementation services with IBM Global Services; 4 years as an IBM Systems Engineer; and 9 years as a systems and application programmer. He holds a

degree in education from Concordia University Nebraska and is a member of the IEEE Computer Society.

**Udo Rauch** is a Certified Tivoli Consultant in Düsseldorf, Germany. His focus is on technical presales in Tivoli/IBM for Storage Solutions, Tivoli Storage Manager, Space Management, Tivoli SANergy, and Tivoli Storage Network Manager. Before joining Tivoli in 2000 he worked in an IT Service company implementing Tivoli Backup and SAN solutions at various customer sites. He has a Diploma in Physics from the Technical University of Munich. His current consulting coverage scope is the EMEA Central Region.

**Daniel Thompson** is a systems engineer for Tivoli based in Texas. Dan's computing experience includes 14 years of mainframe computing (from operations to systems programming), 8 years of PC/LAN server computing, and 5 years of mid-range platforms. His specialties are storage and data management, systems recovery, and systems automation/scheduling.



*Figure 1. The authors (left to right): Ron, Charlotte, Udo, and Dan*

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

Chris Stakutis  
CTO and founder, Tivoli SANergy Development, Westford

Earl Gooch  
Tivoli Systems, Westford

William Haselton  
Tivoli SANergy Development, Westford

Jeff Shagory  
Tivoli SANergy Development, Westford

Chris Walley  
Tivoli SANergy Development, Westford

Randy Larson  
IBM Advanced Technical Support, San Jose

Chris Burgman  
Tivoli Information Development, San Jose

Richard Gillies  
OTG

Tricia Jiang  
Tivoli Marketing, San Jose

Lance Evans  
LSCSi

Emma Jacobs  
ITSO San Jose

Yvonne Lyon  
ITSO San Jose

---

## Comments welcome

### Your comments are important to us!

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

- Fax the evaluation form found in “IBM Redbooks review” on page 307 to the fax number shown on the form.
- Use the online evaluation form found at [ibm.com/redbooks](http://ibm.com/redbooks)
- Send your comments in an Internet note to [redbook@us.ibm.com](mailto:redbook@us.ibm.com)



## **Part 1. SANergy basics**

---

## **2** A Practical Guide to Tivoli SANergy

---

## Chapter 1. Introduction to SANergy

The purpose of this IBM Redbook is to provide information on the practical use of Tivoli SANergy and some typical SANergy implementations. SANergy, like many new storage technology tools, can bring significant value to an organization. However, the complex and rapidly changing nature of storage management in today's environment can make it difficult to differentiate between the new types of components or applications, as well as when each should best be used to make an environment more robust, powerful, or cost effective.

In an attempt to help clarify these topics, we will first offer a brief introduction to new storage technology concepts as well as the new capabilities that this technology can provide. Equally important, we will also discuss the challenges you will face in harnessing these capabilities. This chapter also includes a description of what SANergy is and when you will use it in your environment.

After this introductory chapter, we will focus on topics centered around helping you to implement and exploit SANergy in the most effective manner possible. These topics include:

- SANergy with a Windows Meta-Data Controller (MDC)
- SANergy with a UNIX MDC
- SANergy performance considerations
- SANergy with Microsoft Cluster Services
- SANergy with disk striping
- SANergy super file serving capabilities
- SANergy with Tivoli Storage Manager
- SANergy with OTG DiskXtender 2000

---

### 1.1 Who should read this book?

This book will mainly focus on the implementation and use of SANergy in real world situations. The intended audience is those who will actually be planning and performing these implementations.

However, this introductory chapter is also intended for anybody who simply wishes to understand what SANergy is and where it fits into today's technological landscape.

## 1.2 New storage technology

The last few years have seen tremendous advances in storage technology. These advances have been driven by the explosion in both the amount of data being stored by an enterprise, as well as continuing diversity in the applications which leverage this data. Without a doubt, the largest advances are those of Storage Area Networks (SANs) and Fibre Channel technology, intelligent storage subsystems, and Network Attached Storage (NAS) devices.

Unlike traditional storage layouts where each host directly attaches its storage, SANs allow the storage devices to be attached to a dedicated, high-speed network that typically contains multiple hosts (see Figure 2). They offer many significant advantages over traditional, locally attached, storage devices. SAN-based SCSI protocols are capable of sustaining faster transfer rates than previous standards, most commonly a maximum throughput of 100 megabytes per second. Enhancements to these standards are done on an ongoing basis, and components capable of 200 megabytes per second are available today.

More detailed information on SANs is available in these redbooks:

- *Introduction to Storage Area Network, SAN, SG24-5758*
- *Designing an IBM Storage Area Network, SG24-5470*
- *Planning and Implementing an IBM SAN, SG24-6116*

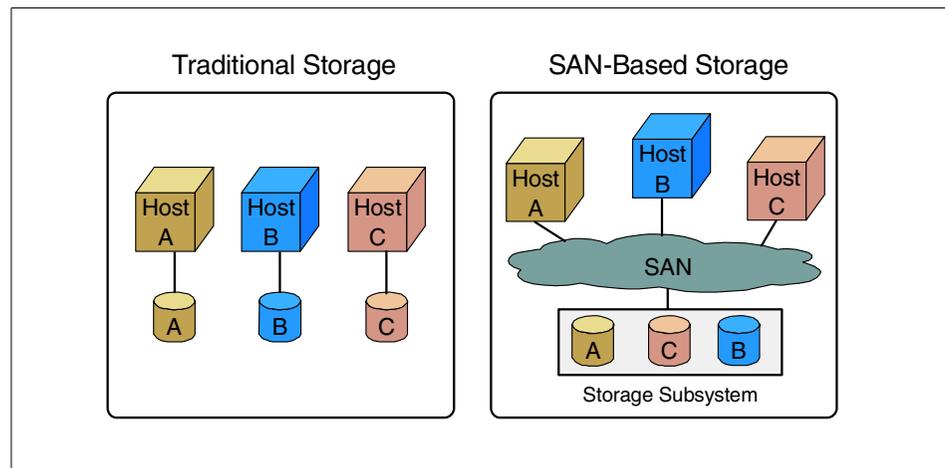


Figure 2. Traditional storage architecture versus Storage Area Networks

In addition to higher levels of performance, SANs also enable many advanced capabilities for data availability, sharing, and protection. Sophisticated storage devices can be attached to the SAN and used by multiple hosts of various operating systems. An example of such a device is an Intelligent Disk Subsystem (IDS) such as the Enterprise Storage Server (ESS) from IBM. The ESS offers powerful capabilities, such as making near instantaneous copies of storage volumes or keeping synchronized copies of the volumes in multiple locations.

There are many cost benefits of implementing a SAN, as well. Although the cost of an IDS will be more than a basic storage subsystem, the cost can be spread across many more hosts. The effort and man-hours required to monitor, administer, and maintain many smaller and isolated storage devices can be greatly reduced by the implementation of fewer, larger, and more manageable devices.

To even further increase the value of a SAN, an organization may choose to implement a SAN-based backup infrastructure. The higher levels of performance offered by SANs potentially decrease the elapsed time of both backups and restores, compared to their former performance over a congested LAN. In addition, larger tape libraries can be used to serve a large number of hosts to further reduce the total cost of ownership.

SCSI over IP, or iSCSI, is a network-based alternative to Fibre Channel based SANs. In iSCSI standards, traditional networks are used to carry the SCSI packets. Each host uses the network based storage as if it were locally attached. An implementation of iSCSI will be able to leverage existing networks and network components but will not achieve the same throughput as a dedicated SAN.

Network Attached Storage (NAS) has also been very important in the growing storage landscape. NAS appliances provide the user with the ability to install and manage LAN-based storage with a significant reduction in the administrative overhead that is necessary with standard file servers. For more information on NAS and iSCSI technologies, please see the redbook, *IP Storage Networking: IBM NAS and iSCSI Solutions*, SG24-6240.

---

### 1.3 Current SAN challenges

Storage Area Networks suffer many of the challenges typical of emerging technology. The four most significant challenges are:

- Interoperability of the SAN components is an issue.
- SAN management standards are new.

- SAN management tools are just now becoming available.
- Existing operating systems, file systems, and applications do not exploit SANs to their fullest potential.

There are many components that make up a SAN. There must be interface cards on (known as host bus adapters or HBAs), storage devices and networking components (Fibre Channel switches, hubs, and routers) to connect them together. These are produced by a wide variety of vendors. Many of the components do not yet work together harmoniously. Even those that do interoperate safely may still be complex to implement. Some companies, such as IBM, operate extensive interoperability labs that perform validation testing on all SAN equipment that they and other leading vendors manufacture.

One of the primary reasons for the interoperability problems is that there did not exist sufficient SAN management standards at the time these devices were designed and manufactured. The standards governing the functionality of the devices, transfer of data, and other core topics had been implemented. However, there did not exist widely accepted standards on how the individual devices would be configured, administrated, and managed.

There are, inevitably, impediments to the development of those standards, such as companies competing to having their own standards selected as the foundation for future development, as well as the expense involved in implementing them. However, much effort has been invested in overcoming these roadblocks. Industry leaders such as Compaq and IBM are investing significant resources to supporting the development of truly open standards in storage networking. Most importantly, these standards are to be governed by recognized standards bodies, such as ANSI (American National Standards Institute). Tivoli is one of a handful of companies who are now offering a SAN management solution. This product, Tivoli Storage Network Manager, fully adheres to the major new SAN management standards, such as the T11 standard governed by ANSI.

The final challenge to fully exploiting SANs is that many of the software components do not yet take advantage of the SAN's full potential. This is not uncommon, as the evolution of new computing technology is often accomplished in phases, with the software following acceptance of new hardware features. It is very important for anyone doing planning to have realistic expectations of what existing software can and cannot do with SAN hardware.

## 1.4 Sharing data on a SAN

Traditionally, storage devices are locally attached and fully owned by an individual host. At most, a couple of hosts may be attached and can control any device at a given time.

Despite the fact that SAN hardware may enable multiple hosts to connect to the same storage, the hosts' operating systems and file systems cannot yet utilize this capability without additional software. In other words, multiple hosts cannot share ownership of a volume. Attempting to share storage directly using a traditional file system will, at best, simply not work. At worst it may yield an unstable configuration that will result in inconsistent data access. Therefore, most implementations of SAN storage allow hosts to share the devices, but not the logical volumes themselves (see Figure 3). In other words, each host will own a slice of the entire storage subsystem. This means that many, small slices will have to be allocated, monitored and maintained.

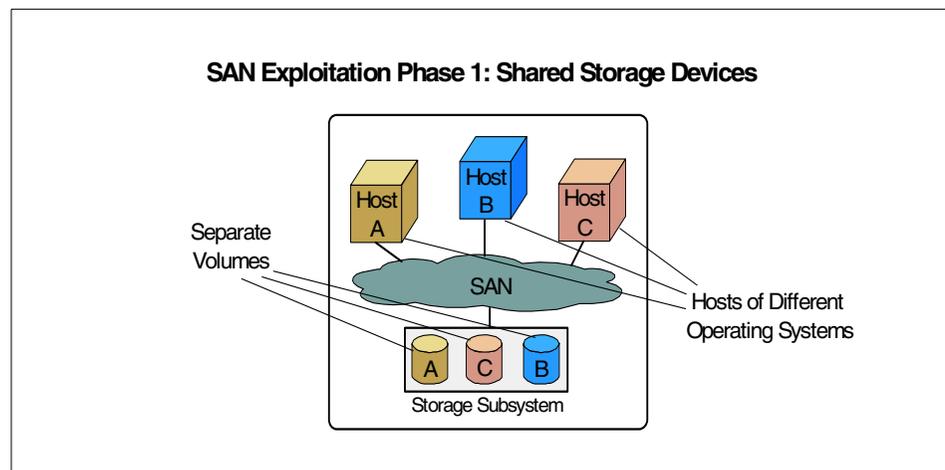


Figure 3. Sharing storage devices with Storage Area Networks

Today there are several examples of file systems that are designed to safely share volumes on SAN-attached storage devices, but they operate only with hosts running the same operating system. IBM's AIX, Compaq's Tru64, and SGI's Irix operating systems are some that have their own proprietary clustered file systems. For example, multiple AIX hosts can share the same disk storage volumes utilizing the General Parallel File System (GPFS). However, no other operating system can share that storage directly. This can be called homogeneous sharing, since it works only for a single operating system (see Figure 4).

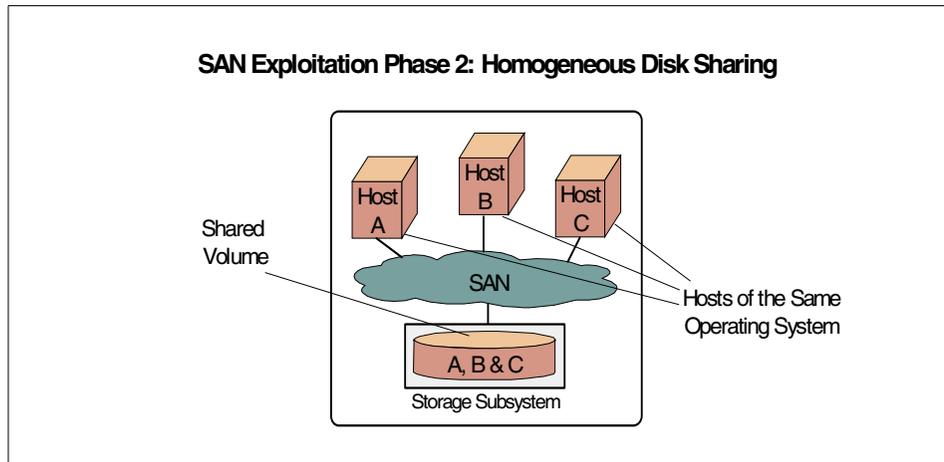


Figure 4. Sharing of volumes for homogeneous operating systems

Although there are some operating systems with a native file system capable of directly sharing volumes, there are many key operating systems that lack this capability. Most notable are the Microsoft Windows NT and Windows 2000 operating systems. There are some third-party vendors that do offer such file systems as add-on components, but they do not offer the ability to share the file system with multiple operating systems and are normally prohibitively expensive to implement. When most users need to share storage owned by a Microsoft Windows server, they normally fall back on LAN-based sharing using native capabilities such as CIFS. Pure LAN-based file sharing has significant disadvantages over a file system that allows multiple hosts direct access to the volumes. We will outline those differences later in this chapter.

Since there is a lack of support of SAN-based volume sharing in current operating systems and file systems, it is obvious that application software also has the same limitation. Once the executive software (operating system and file system) become SAN-aware and offer new capabilities to the application software they host, the applications themselves may need to be modified to fully exploit the new capabilities.

There is good news on the horizon. Several vendors are developing software that enables volume sharing to SAN-attached hosts. This sort of software creates a virtual view of the storage on the SAN with a common naming standard that each of the hosts can resolve using client software. IBM and Tivoli are taking storage virtualization even further by developing software to bring Systems Managed Storage (SMS), a concept familiar from the

mainframe environment, to the open operating system arena. This software will enable policy-based data placement and management, server free data management as well as sharing. Data management will include backup/recovery, hierarchical storage management (HSM) and expiration and replication. All of the data management will take place over the SAN, transparent to the attached hosts.

For more information on SMS and storage virtualization, see the redbook, *Storage Networking Virtualization: What's it all about?*, SG24-6210.

---

## 1.5 Tivoli SANergy

Although the promise of the software components currently under development is very exciting, there are many organizations who need to begin exploiting SANs today. Also, many organizations will not need a full storage virtualization infrastructure, but do require SAN volume or file sharing.

SANergy from Tivoli Systems is a unique application that allows you to realize the cost savings, performance benefits, and new capabilities provided by SANs, today. SANergy allows you to share storage volumes to multiple hosts running various operating systems. You will not need to implement a new operating system or new file system to accomplish this, and existing applications will run as they are currently designed. The skills of existing personnel for administrating LAN-based shared storage will be similar to those needed to manage SANergy shared storage. SANergy is a very mature product (relative to this area of technology) and currently runs on over 5000 hosts and 1000 customer environments.

SANergy accomplishes the sharing of SAN storage by integrating with traditional LAN-based file sharing applications, NFS and CIFS. Through this integration, SANergy hosts will access the shared data over the SAN instead of the LAN. This redirection of the data I/O is transparent to the hosts' operating system and applications. The applications see the disk volumes as if they were accessing them using a traditional LAN-based configuration.

NFS (Network File System) is a file-level protocol for sharing file systems across the network. It originated on UNIX platforms and was developed by Sun Microsystems. CIFS (Common Internet File System) is also a file-level protocol for sharing disk storage. It originated in the Microsoft operating system world. It was formally known as SMB (System Message Block). CIFS, like NFS, is available on both UNIX and Microsoft platforms.

By utilizing this unique and deceptively simple technique, SANergy allows you to fully exploit the SAN without re-engineering your entire infrastructure (other than implementing the SAN itself). SANergy enables an organization to exploit their SAN in a wide variety of ways. It is also important to note that SANergy is truly “set and forget” software — that is, after setting up the initial configuration, there is basically no ongoing maintenance required to keep the configuration performing effectively (apart from adding new file systems or hosts to be managed by SANergy). SANergy will simply do its job quietly in the background.

### 1.5.1 What are the benefits of SANergy?

Here we discuss the main benefits you can expect to receive from implementing SANergy, including sharing disk volumes, backup and recovery, and file sharing.

#### 1.5.1.1 Sharing disk volumes

One of the most beneficial features of SANergy is the sharing of disk volumes between hosts of multiple operating systems — heterogeneous disk sharing (see Figure 5). This not only reduces the total number of logical volumes to be administered within an enterprise, it also solves several technical and business problems as well.

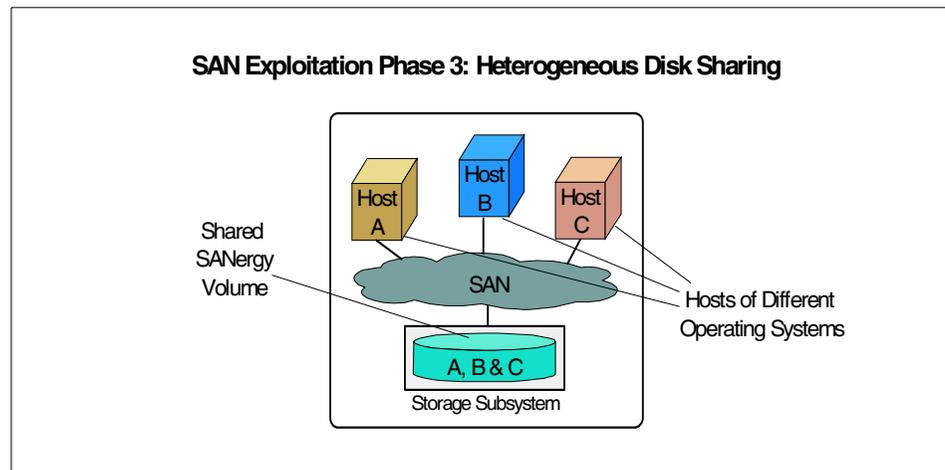


Figure 5. Sharing volumes from heterogeneous hosts with SANergy

When considering whether a configuration will benefit from disk volume sharing using SANergy, we will first look at what benefits may come from using Network Attached Storage (NAS) devices or normal LAN-based file sharing. The attraction of NAS is that an organization can invest much less effort in administering its storage. For example, it is not uncommon for an NT server to have 2-3 disk volumes used to store its data. Each of these must be installed, administered, and monitored. If an organization utilizes a NAS device for 10 of these NT servers, the number of disk volumes could be reduced from 20-30 to 1 or 2 (see Figure 6).

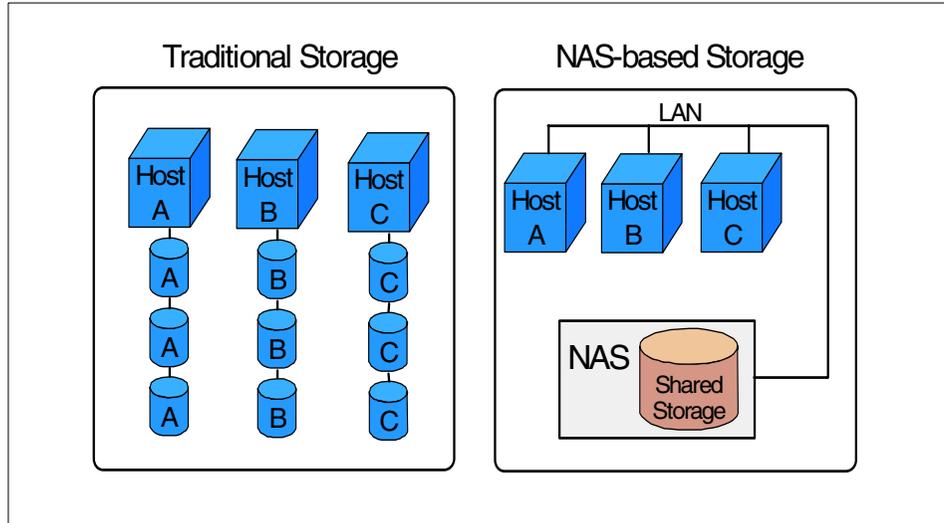


Figure 6. Locally attached storage compared to exploiting a NAS appliance

SANergy will provide the manageability of NAS or file servers for a configuration and bring additional benefits. For example, NAS vendors are trying to influence customers to begin storing their databases on NAS storage. This brings the advantage of ease of administration as well as storage consolidation. However, a purely LAN-based configuration may well suffer poor throughput as well as place high demands upon the CPU resources of the host machines. The CPU workload of large data transfers using TCP/IP can quickly overwhelm a host, but SAN I/O has negligible impact.

When using SANergy for volume sharing, instead of NAS, you can gain the advantages of easier administration and storage consolidation, but the performance will be at SAN speeds and the CPU resources of the hosts will not be overtaxed. Other advantages include a lower utilization of the existing LAN as well as more consistent data writes. NAS and LAN-based file serving solutions may signal that a write I/O is complete before it is written to disk. This problem, referred to as *unsettled writes*, has previously prevented critical applications, such as databases, from being implemented in a LAN-based file sharing environment.

Volume sharing has other advantages apart from ease of administration and storage consolidation, including functional advantages. Consider the sharing of volumes needed by Web servers or file servers. Although they both benefit from sharing their disk volumes, they do so for slightly different reasons.

When an organization designs a Web serving infrastructure, they often require more horsepower than is available in a single machine. Without SANergy, there are a couple of options, each with significant drawbacks. The first option is to increase the horsepower of the existing machines, which may be expensive and not provide sufficient redundancy. The other option is to add more Web servers. While this may allow you to use less expensive servers and does provide redundancy, there are some drawbacks.

When using multiple servers, there will either be duplication of the data being served or you will have to administer a Web site that is stored in many locations. Also, there needs to be a process to ensure that all copies of the data are kept synchronized and organized. By using SANergy, you can implement a configuration that takes advantage of multiple, cheaper Web servers without the disadvantages of each server having its own storage. Each of the Web servers will simply share a copy of the entire Web site which is stored on shared volumes.

File servers also benefit from the sharing of disk volumes. Many organizations require multiple file servers to adequately support their user community. Each of these file servers will typically have its own storage that will have to be monitored and administered. By consolidating this to fewer, larger volumes, the administrative overhead is reduced, along with the frequency of maintenance required to increase storage or relocate data. A new file server is often needed because there is no available disk space on existing servers. By consolidating storage, the number of file servers required is dictated not by the amount of maximum storage per server, but rather by the actual number of machines needed to adequately serve the workload. It is more desirable to have each tier in a multiple tier architecture scaled by the function it performs, rather than by another tier's requirements.

There are other advantages to using SANergy in the two scenarios described above. For example, a SANergy implementation could also include space management software to perform automatic migration and expiration based upon the age and size of files. This means that all of the hosts sharing the volumes have access to an almost unlimited amount of virtual storage. In Chapter 8, “SANergy and OTG DiskXtender 2000” on page 227 we describe how to setup DiskXtender 2000 from OTG and Tivoli (which provides HSM function for Microsoft hosts) with SANergy.

Another significant advantage to utilizing SANergy in these configurations is that less expensive hosts can be used for different functions in the configuration. For example, a Windows NT server can be used in a file serving infrastructure to provide authentication as well as space management. The remaining hosts can then be Linux machines built upon less expensive hardware which perform the actual serving of the files via Samba.

#### **1.5.1.2 Backup and recovery**

When most organizations choose to implement a SAN, they will also want to enhance their data protection infrastructure so that backups and restores will take place over the SAN with maximum performance and minimal impact to the hosts. SANergy allows you to perform SAN-based, server-less backups of your shared storage. Backups and restores in such a configuration take place with no impact to the hosts sharing the storage and exploit the enhanced throughput of the SAN. An example of such a configuration using SANergy and Tivoli Storage Manager will be documented later in this redbook (see Chapter 7, “SANergy and Tivoli Storage Manager” on page 191).

#### **1.5.1.3 File sharing**

One of the most common uses of SANergy is to allow the sharing of files by multiple users or applications (see Figure 7). For example, modern filmmakers exploit computer-generated special effects to replace what once required the use of expensive, dangerous stunts and model-based special effects. The data files that contain these movies are very large and a post-processing company may need to have multiple personnel working on the same file, at different locations within the file. SANergy, in conjunction with traditional LAN-based file sharing, allows these huge files to be edited by multiple SAN-attached hosts. The data reads and writes are redirected to utilize the SAN for much greater performance. The application must include a mechanism to safely handle concurrent file access. Thus, SANergy's role in this implementation is to enable SAN-based data I/O. It does not alter the capabilities or design of the application in any other way.

Another key application of this type is the processing of large data files by high-performance computing systems. For example, raw geophysical or medical imaging data may need to be accessed simultaneously by several processes, which may be on multiple nodes in a high-performance cluster. Whenever a configuration calls for the simultaneous access of large files at high speeds, SANergy is an ideal solution.

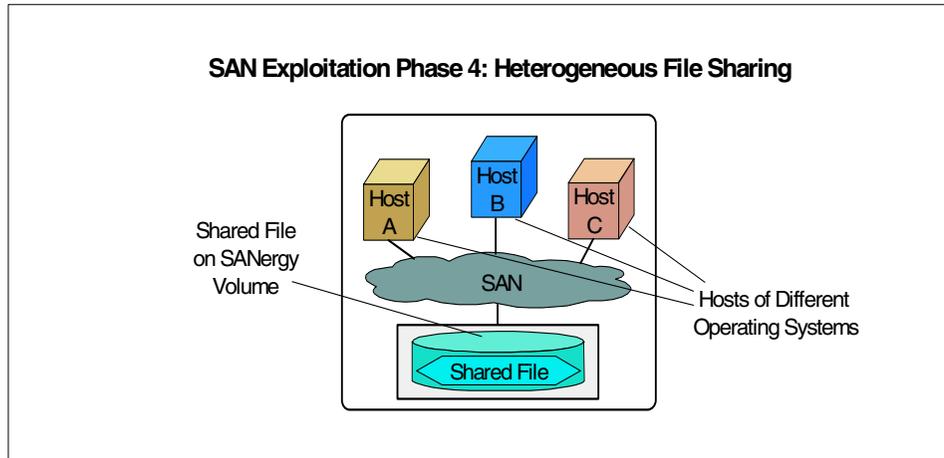


Figure 7. Sharing files with SANergy and Storage Area Networks

### 1.5.2 SANergy limitations

SANergy enables many powerful features, but there are some scenarios where you may choose not to deploy it. SANergy, like most distributed file-sharing solutions, has a small amount of overhead associated with the accessing of each file on a shared volume. If a file is very small, it may actually be more efficient to just send it over the LAN rather than redirecting to the SAN.

This cutover point is normally reached with files 100 KB in size or smaller. Therefore, if an entire workload is made up of files this small, SANergy may not add any performance benefit. Since typical workloads and environments are made up of a variety of file sizes, SANergy can be implemented so that it will bypass tiny files while still being available for files it should handle. However, even in those situations in which SANergy may offer minimal speed increases, you may still find significant advantages by using SANergy's volume sharing capabilities.

While it is not a limitation of SANergy, care must be taken when utilizing SANergy with other products that may intercept and/or redirect I/O. While SANergy has been shown to coexist with other applications that intercept I/O, it is important that these combinations be validated. An example of such an application may be space management or anti-virus software. If there is any question, you should ask your IBM/Tivoli representative if testing has been done with the products working together. If not, consider doing your own testing before implementation in a production environment.

In general, when understanding the capabilities or limitations of SANergy, it is necessary to understand the limitations of the LAN-based file sharing infrastructure itself. For example, neither NFS or CIFS can share a raw partition. Therefore, neither can SANergy. However, it is critical to understand that many of the limitations of LAN-based file sharing, such as poor performance, the heavy load on a host's CPU and unsettled writes, are eliminated when using SANergy.

---

## 1.6 Summary

SANergy is a product that allows you to share SAN-attached disk volumes (such as performance, sharing, cost-cutting, new capabilities) without redesigning all aspects of your infrastructure. SANergy does not attempt to re-implement core features such as security authentication or the advertisement of shares, but rather integrates with standard LAN-based file-sharing systems (NFS and CIFS). A SANergy configuration provides many advantages over simple NAS or LAN-based file sharing, including better performance and data consistency. SANergy allows a customer to truly exploit their SAN's potential

---

## 1.7 SANergy requirements

SANergy has the following requirements:

- At least one machine on the SAN needs to be identified as a Meta Data Controller (MDC). The MDC "owns" the volume being shared and formats it using its file system. There can be as many MDCs as there are volumes being shared on the SAN, or a single MDC may share a large number of volumes. The number of MDCs will depend upon a given configuration and workload (see Figure 8 on page 17).
- All of the machines sharing the storage need to share a common LAN protocol and be able to communicate with one another. across it.

- All of the machines sharing the storage need to share a common SAN with at least one storage device. The hosts' HBA cards must support multi-initiator environments and be compatible with other SAN components.
- All of the machines need to have an NFS or CIFS client capable of sharing files from the MDC. If a UNIX MDC is sharing files to Windows hosts, it will need to have a CIFS server, such as Samba, configured. If a Windows MDC is sharing files to UNIX hosts, it must be running an NFS server. Macintosh hosts will use an SMB client that ships with SANergy and have the same requirements as Windows hosts.
- With Tivoli SANergy 2.2, MDCs running either Windows NT 4.0, Windows 2000, Red Hat Linux, or Solaris are supported. The disk partitions owned by those MDCs must be formatted with either NTFS, UFS, EXT2FS or any other Tivoli SANergy API-compliant file system such as LSCi's "QFS" and "SAMFS".
- The following operating systems are supported SANergy hosts:
  - Microsoft Windows NT
  - Microsoft Windows 2000
  - Sun Solaris
  - IBM AIX
  - Red Hat Linux
  - Apple Macintosh (Mac OS 8.6 or 9)
  - Compaq TRU64 UNIX
  - SGI IRIX
  - Data General DG/UX
- You need a SANergy license for all of the hosts participating in file sharing, either as an MDC or a SANergy host (client).

The new IBM TotalStorage™ Network Attached Storage 300G series also includes SANergy as pre-installed, ready to license component.

For other, more detailed, system requirements, please see the Tivoli SANergy Web site at:

[http://www.tivoli.com/support/sanergy/sanergy\\_req.html](http://www.tivoli.com/support/sanergy/sanergy_req.html)

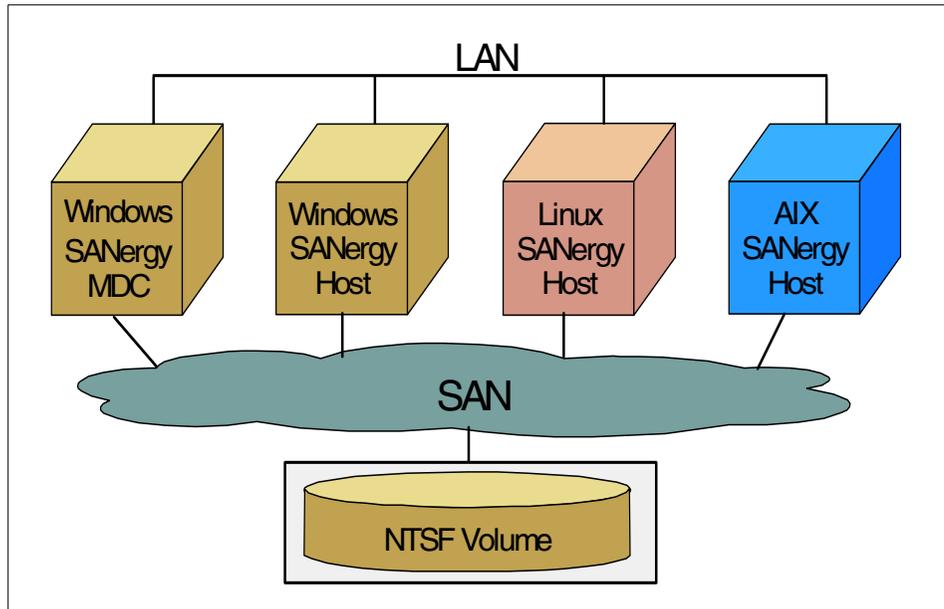


Figure 8. Example of SANergy configuration



---

## Chapter 2. SANergy with a Windows MDC

This chapter covers the procedures required for setting up a SANergy environment with a Windows NT or Windows 2000 computer as Meta Data Controller (MDC). Using the Windows platform as MDC lets you share an NTFS partition on a disk system attached to the SAN to UNIX and Windows hosts. The working of SANergy depends on two basic prerequisites: a well-configured SAN environment and functional LAN-based file sharing. We highly recommend following through the installation procedures carefully, validating at each step to minimize unnecessary and time-consuming troubleshooting in a complex environment afterwards. We assume from here on that your SAN is operational — that is, correct hardware and device drivers are installed so that basic host to storage device access is enabled.

In this and future chapters we will use the term *SANergy host* to refer to a host system which has client access to SAN-shared files, in contrast to the MDC which is the file server. We also use the term *fusing* or *fused I/O* to describe file sharing I/O which is directed through the SAN using SANergy. Unfused I/O, by contrast, refers to standard file sharing I/O which is directed over the TCP/IP LAN.

---

### 2.1 File sharing setup

Before setting up SANergy to exploit the Fibre Channel technology by accelerating access to shared data resources in a networked storage environment, you have to first get file sharing running in a LAN environment. Assuming that your machine designated to be the MDC has access to a storage device in the SAN, these sections will demonstrate how to allow other computers to share this data.

The other computers do not need to be physically connected with access to that storage device over the SAN at this point. In fact, we recommend that at this stage you allow only the machine which will be the MDC to have access to the shared disks. If the SANergy host or client machines have access to the shared disk, there is the risk of them writing their own signatures or otherwise trying to take ownership of the drives.

There are two ways to prevent the SANergy hosts from accessing the disks: You can either disconnect their HBAs from the SAN, or you can use zoning capabilities of your fabric switch to prevent access to the switch. Zoning means grouping together ports on your switch so that the devices connected to those ports form a virtual private storage network. Ports that are members of a group or zone can communicate with each other but are isolated from ports in other zones.

The method that you choose will depend on your particular SAN topology, and what, if any, other access to SAN devices is required by the SANergy hosts. Disconnecting the HBA is a “shotgun” approach — simple but effective. Zoning the shared disks is more elegant, but may be more complex to implement, and may not even be possible if you are in a non-switched environment, or your switch does not support zoning.

In our setup, we used zoning for the IBM SAN Fibre Channel switch. Instructions on how to configure this are available in the redbook *Planning and Implementing an IBM SAN*, SG24-6116. We configured the MDC in a zone with access to the shared disks and put the SANergy hosts into another zone.

Once you have isolated the SANergy hosts from the disks, check that the LAN connection is properly configured, so that there is TCP/IP connectivity between SANergy MDC and hosts. Our first objective is to have the following environment (Figure 9) up and running. We have a Windows 2000 MDC, diomede, and two SANergy hosts — pagopago running Linux and cerium running Windows NT.

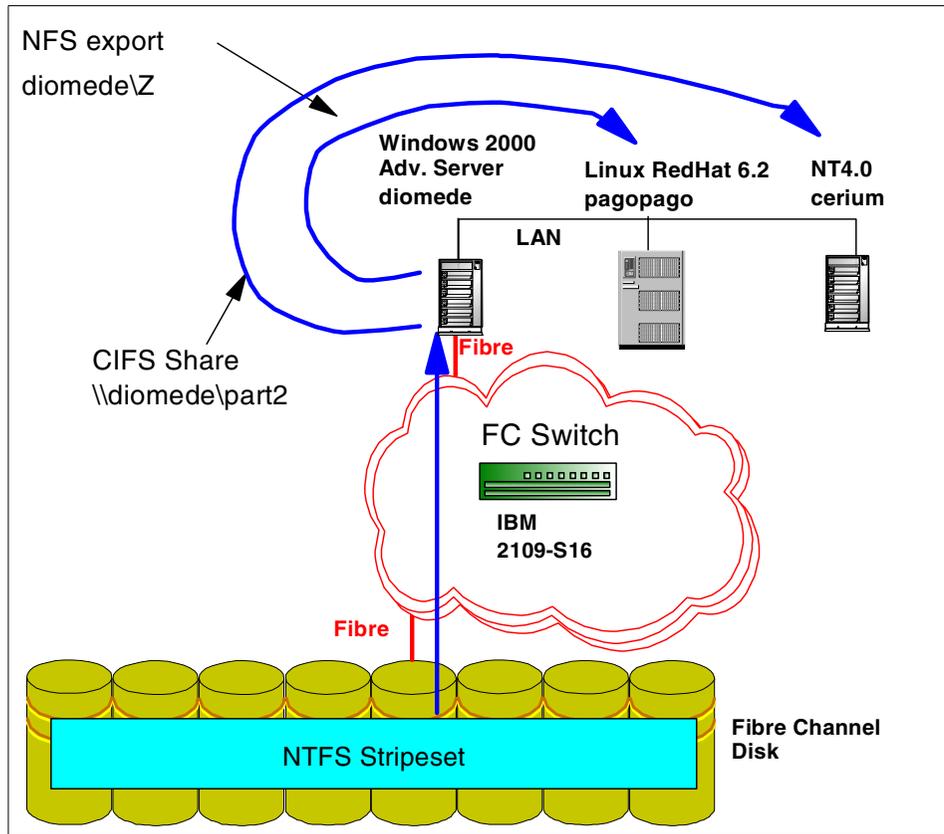


Figure 9. File sharing environment with Windows file server

## 2.1.1 Windows server file sharing

A Windows NT/Windows 2000 server can export files to other Windows or UNIX host computers. Different file sharing procedures are needed for each of those clients — that is, NFS for serving to UNIX hosts and CIFS for serving to Windows hosts. We will show the setup of each of these two options on the MDC in turn.

### 2.1.1.1 CIFS server setup

CIFS stands for Common Internet File System. CIFS is an enhanced version of Microsoft's open, cross-platform Server Message Block (SMB) protocol, the native file-sharing protocol in Microsoft Windows 95, Windows NT, Windows 2000 and OS/2 operating systems and the standard way that PC users share files across corporate intranets. CIFS is also widely available on UNIX, VMS, Macintosh, and other platforms.

For sharing files between Microsoft Windows systems, it is not necessary to install additional CIFS software. The built-in SMB protocol capability of Windows operating systems is sufficient for normal file sharing and SANergy purposes. Once you have access to a disk device on the SAN and formatted the NTFS file system, you can give access to this data to other machines on the local network using native file sharing functions of the Microsoft operating systems. Figure 10 shows the screen which displays when you right-click on the partition you want to share.

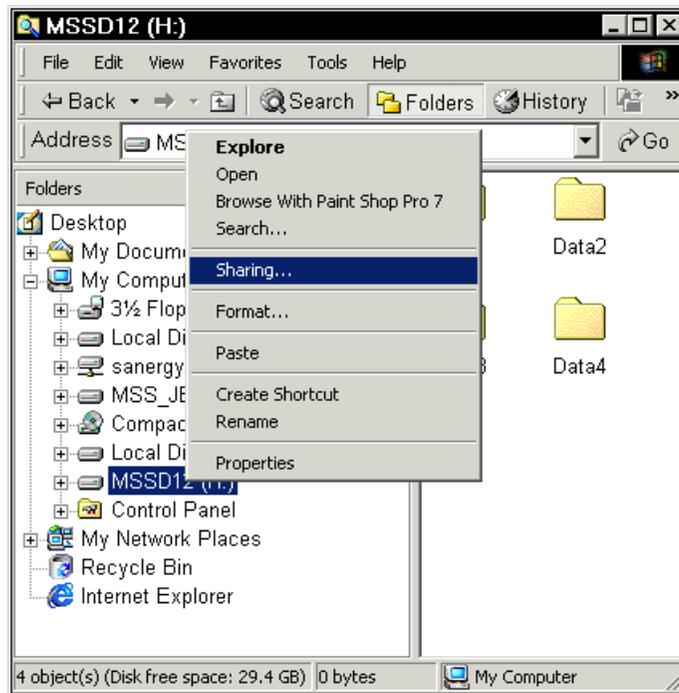


Figure 10. Sharing option in explorer view

Select the sharing option to get to the next window, shown in Figure 11.

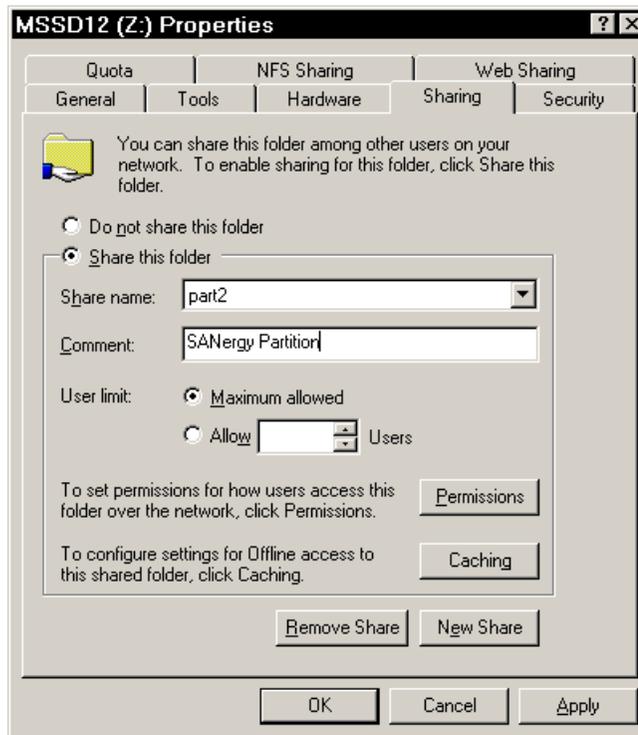


Figure 11. File sharing properties

Select **Share this folder** to export the partition. You have to define a name for this share which will be part of the fully qualified name matching the Uniform Naming Convention (UNC). Other computers can now access the partition under the address: \\computername\sharename (in our case it would be \\cerium\part2).

By default the permissions are set for full access to everyone. You can grant specific permissions (for example, read-only access) by choosing the button **Permissions** in this window. Click **OK**, after which the Explorer view (see Figure 12) indicates that the partition is now shared.

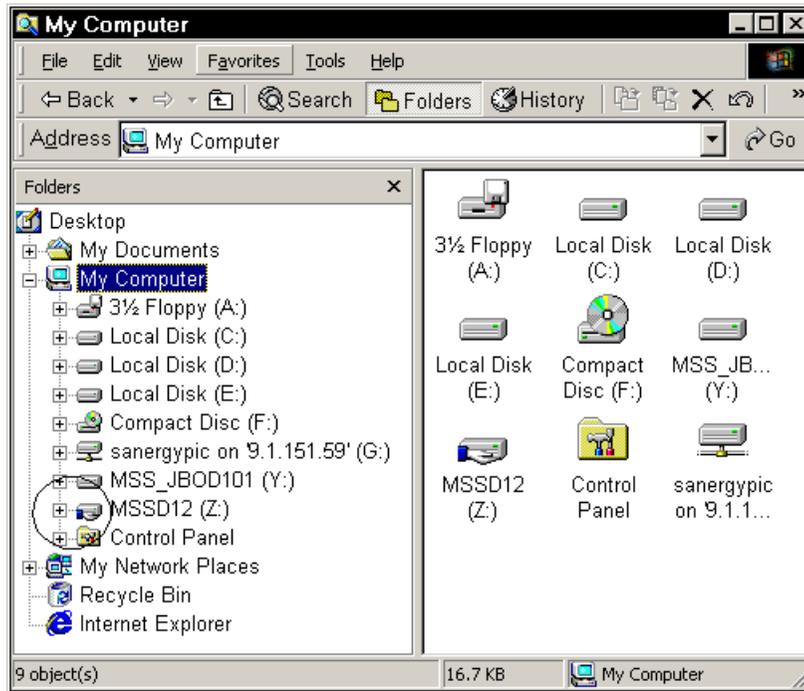


Figure 12. Share icon for exported partition

**Note**

To ensure access for Windows users other than the administrators to a Microsoft share, we always recommend that you define an explicit share as shown for file sharing intentions. Although on Windows NT and Windows 2000 there is already a hidden administrative share present by default, we recommend creating a separate one. We found also that using the default administrative share names caused problems in getting the SANergy volumes to fuse.

### 2.1.1.2 Sharing the storage via NFS

There are different ways for Windows machines to share partitions so that they can be mounted on UNIX machines. In all cases you have to install a supplementary product. We want to cover two scenarios in this chapter: the use of Hummingbird NFS Maestro server and of Microsoft Services For UNIX (SFU) NFS server.

#### ***NFS Maestro server***

The NFS server product used on our Windows system was NFS Maestro from Hummingbird. We installed only the NFS server component of Maestro NFS, using the standard install shield. The installation process creates an icon on the Windows Control Panel called Hummingbird NFS Server. See Figure 13. Double-clicking on this icon will start the NFS Maestro server configuration.

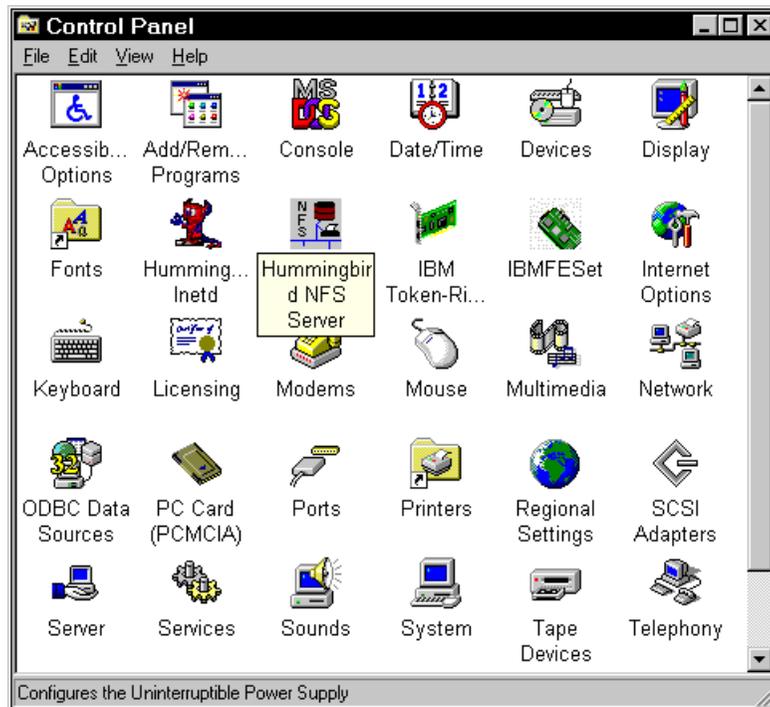


Figure 13. Control panel showing Maestro NFS

Configuration of the NFS server can be done entirely through the GUI. Figure 14 shows the **General** option on the configuration screen which is used to set basic parameters. For use with SANergy, it is recommended to select the **Users** and **Groups Map** options along with **NTFS Style Permissions**.

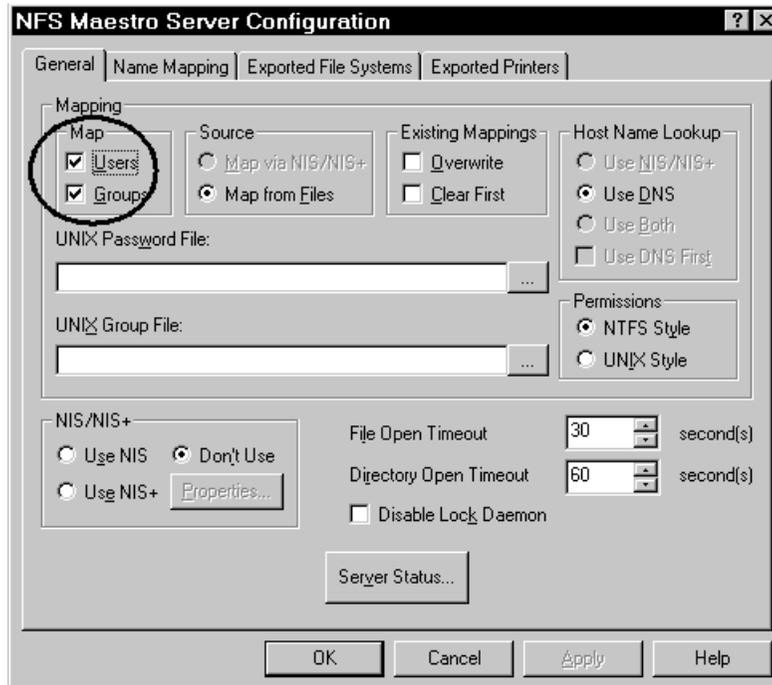


Figure 14. NFS Maestro server general configuration

This general configuration screen is also used to display the NFS Maestro Server Status screen, which shows NFS usage statistics. Figure 15 is an example. A further use for the status screen will be discussed later.

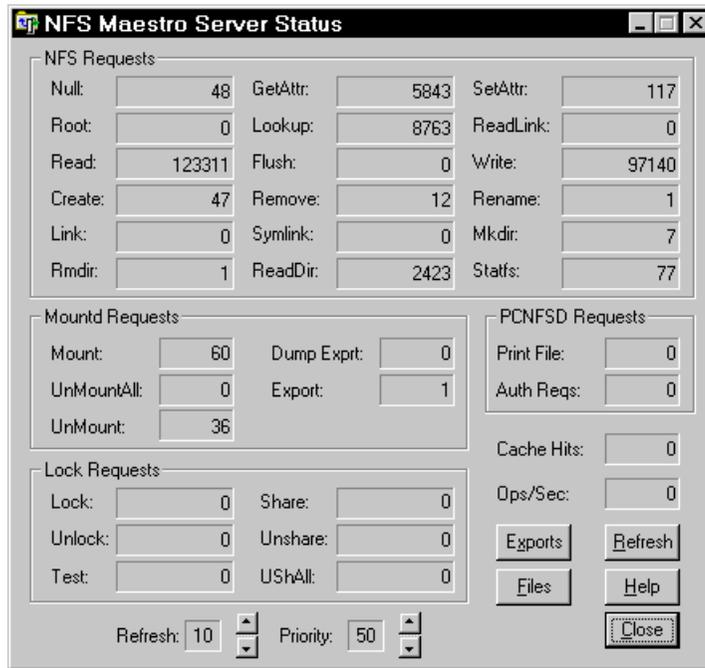


Figure 15. NFS Maestro Server Status

The next step in the NFS Maestro configuration process is name mapping. Mapping UNIX userid's to Windows user names is required to allow UNIX users to access the exported volumes. Figure 16 shows the panel used for name mapping.

There are two types of mapping:

- User mapping
- Group mapping

### User mapping

For SANergy use, the minimum recommended *user mapping* is as follows:

- Windows user Administrator <-> UNIX user id 0 (root userid)
- User Name for Non-Mapped Access: Guest

### Group mapping

For SANergy use, the minimum recommended *group mapping* is as follows:

- Windows group Administrators <-> UNIX group id 0 (system group)
- Group Name for Non-Mapped Access: Guests

For our testing, we took a simple approach to Windows volume access from our UNIX systems. We used the root user for mounting and SANergy use. Depending on your security requirements, it may be desirable to limit Windows volume access to a select group of users on the UNIX system.

One method to accomplish this would be to define a UNIX group and add the userids requiring SANergy volume access to it. Our example uses a UNIX group id of 206. Consult your own UNIX operating system documentation for details of how to create groups. A corresponding Windows group would also need to be defined. In this example we called it “SANergy Group”. The mapping of group id 206 to Windows group “SANergy Group” is shown in Figure 16. Appropriate Windows permissions would also need to be set up to restrict volume access to just the “SANergy Group”.

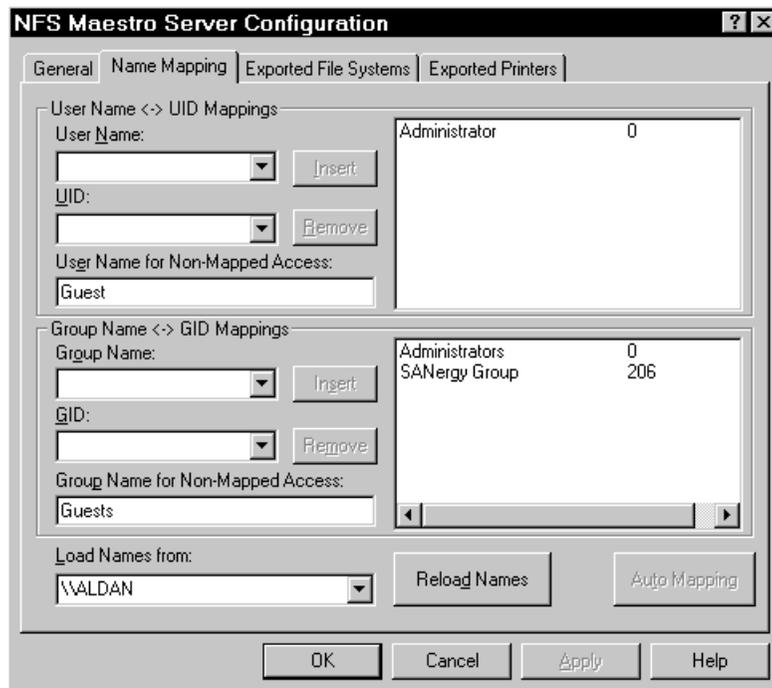


Figure 16. NFS Maestro server name mapping

The last NFS server configuration step is to export the desired Windows drives or volumes. NFS Maestro refers to them as file systems. Figure 17 shows our export configuration. We exported two drives, F: and G: and we used G: for our SANergy testing. Access to the exported drives can be restricted to specific UNIX systems on this panel. For SANergy usage on the selected hosts, normally you will choose the option “Read/Write Access”.

If the root user will be using SANergy, the option “Allow Access for Root User” should also be chosen.

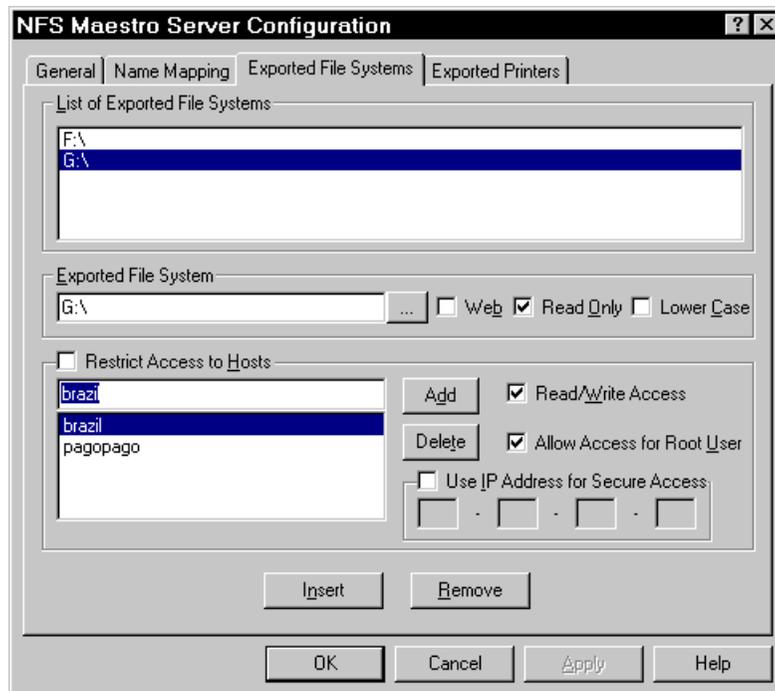


Figure 17. NFS Maestro Exported File Systems

When changes are made on the Exported File Systems panel, the exports must be reloaded. This can be performed dynamically using the **Reload** button on the Current Exports panel, shown in Figure 18. The Current Exports panel is displayed by using the Exports button on the Server Status screen, as shown on Figure 15 on page 27.



Figure 18. NFS Current Exports

The Maestro NFS exports are now ready to mount on the UNIX systems.

### **Microsoft SFU NFS server**

Services for UNIX (SFU) is a separately-priced product from Microsoft that lets you share files from a Windows platform to UNIX hosts. There are different components within the whole SFU bundle. The components we need for SANergy are the SFU NFS server, user name mapping, and server for NFS authentication. The installation is straightforward, using Install Shield. After installation of these products you will be able to directly share a partition or directory from the explorer by right-clicking on the drive that you want to share as for normal CIFS sharing. The window shown previously in Figure 11 on page 23 will display now a new tab named **NFS Sharing** (Figure 19).

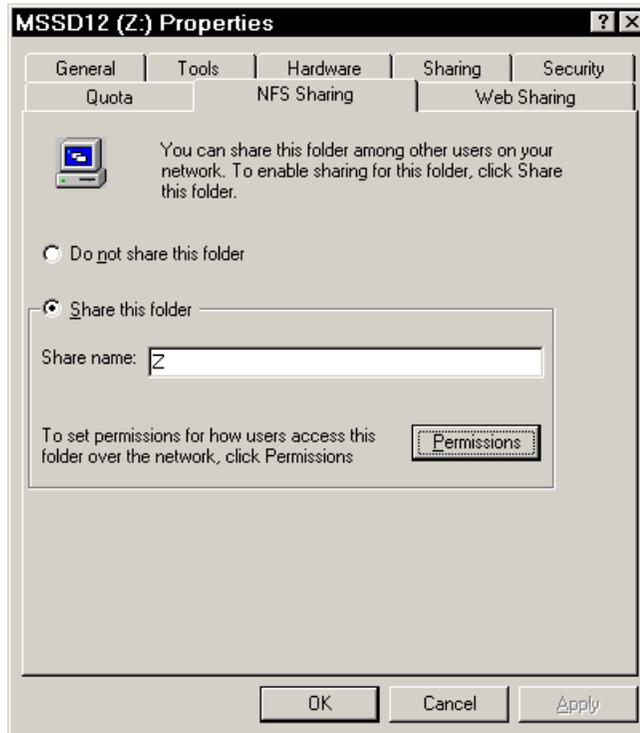


Figure 19. NFS sharing with SFU

In the blank field you can type in the share name after activating the option **Share this folder**. You can also grant access to specific computers for this share. Click on the **Permissions** button to access the dialog box in Figure 20.

**Important**

For SANergy to correctly share the volume, you should use the actual drive letter assigned by the operating system for the share name as shown in the figure. You should use a defined name here.

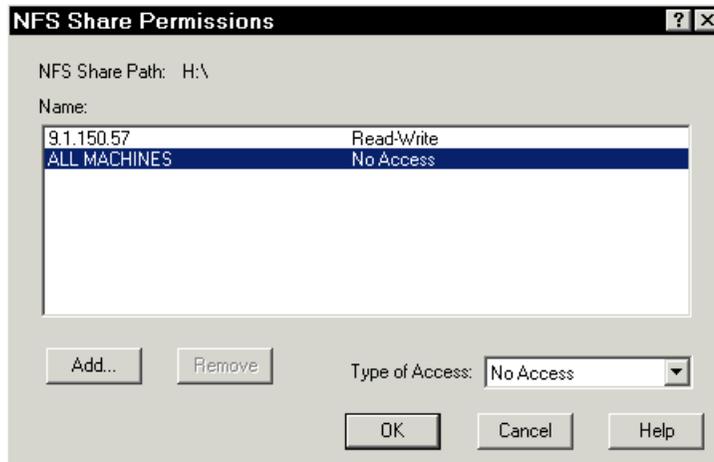


Figure 20. Granting permissions to hosts with SFU NFS.

By default, all machines have read/write permissions on the share; however, it is possible to grant authority for individual machines. Use the **Add** and **Remove** buttons and select the appropriate **Type of Access**. Setting permissions on the user level requires more administrative overhead. To give UNIX users the appropriate permissions on the imported NTFS partition, the Windows system must identify the user and group names from the UNIX machines. Users in a Windows domain should be mapped with users on UNIX machines. To set this up, from the Windows task bar, select **Start > Programs > Windows Services for UNIX > Services for UNIX Administration**.

The easiest case is when your UNIX users are identical with the Windows users. Then you can proceed with the simple mapping option (Figure 21).

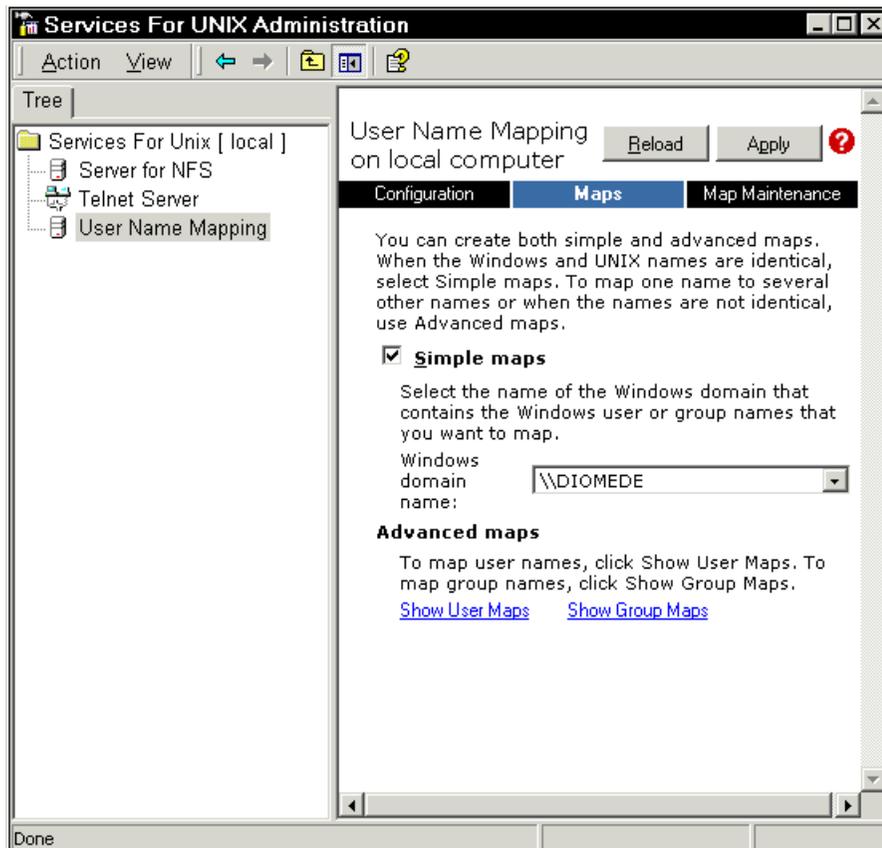


Figure 21. User mapping administration

Select the **User Name Mapping** option in the left-hand panel, check the **Simple maps** box and specify in the **Windows domain name** section your Windows domain name, or the MDC server name if not in a domain.

If the users on both systems are not identical, you will need to use **Advanced maps** to map each user on the Windows side to the corresponding user on the UNIX side. In this case, it is necessary to have a NIS server (Network Information Service) running in your network. For further information on setting up user name mapping with NIS, see:

<http://www.microsoft.com/windows2000/sfu/>

## 2.1.2 Client file sharing

Once the designated MDC server in your SANergy environment is set up for file sharing, you can now set up the clients to get access to the shared storage.

### 2.1.2.1 Accessing a Windows share from a Windows system

From a Windows system you can map a shared drive by choosing **Tools -> Map network drive** in the Explorer window, as shown in Figure 22.

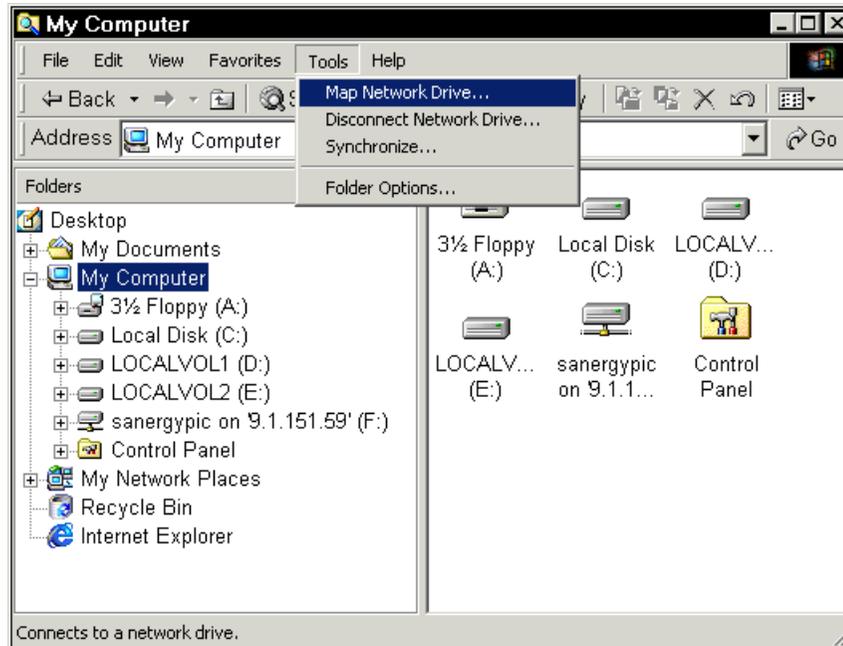


Figure 22. Map a shared network drive

You will see the screen in Figure 23.

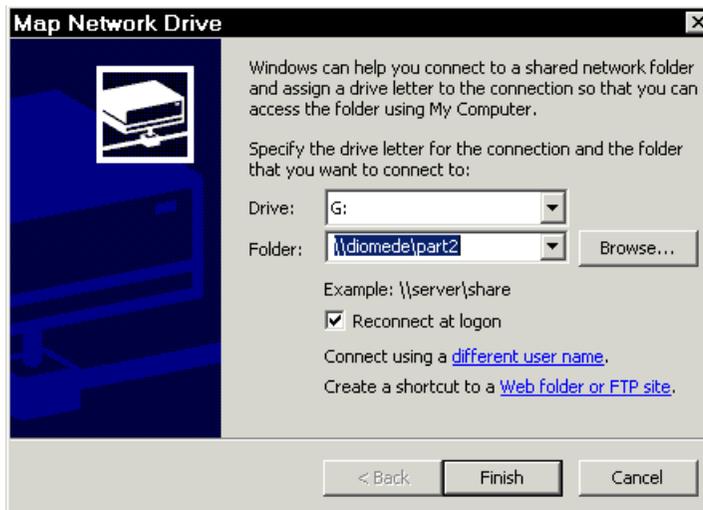


Figure 23. Type UNC path to map shared network drive

To map the shared drive, type in the field **Folder** the share name and click **Finish**. The connection will be established. A new drive appears in the Explorer window as shown in Figure 24.

**Remember**

Do not map the administrative shares using drivenames C\$, D\$ at this point for use with SANergy. Use the defined sharenames, since, as mentioned in the Note in 2.1.1.1, “CIFS server setup” on page 21, this may lead to problems getting fused reads/writes (that is, using the SAN) on the shared disk.

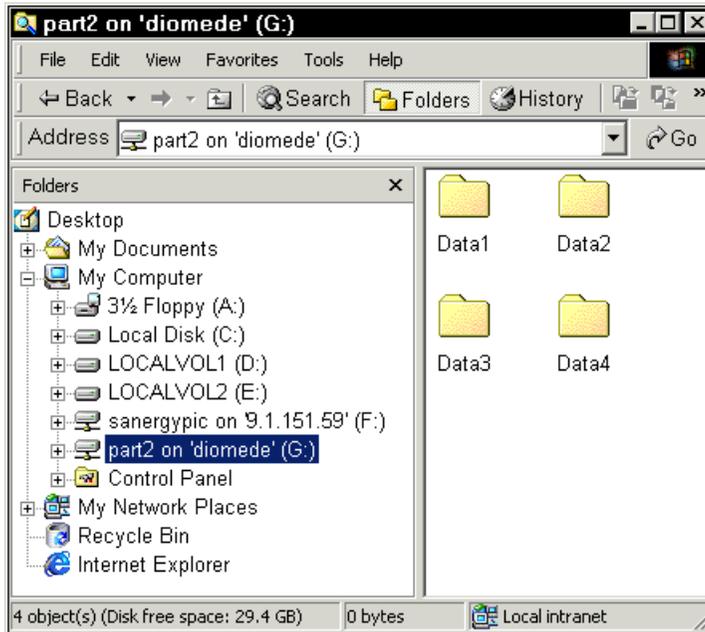


Figure 24. Access to a new share

### 2.1.2.2 Accessing a Windows share from a UNIX system

NFS client setup on UNIX is a simple process. We used Linux Red Hat 6.2 as our NFS client and SANergy host platform connecting to the Windows MDC. For the file sharing procedures on AIX please see 3.1.2.1, “UNIX NFS client setup” on page 67.

These are the system requirements:

- NFS installed on the Windows MDC as described in 2.1.1.2, “Sharing the storage via NFS” on page 25.
- TCP/IP connectivity from the SANergy UNIX host to the MDC

#### **Mount the NFS shares**

The steps to mount NFS shares exported from other hosts is similar in most UNIX environments, but not identical. These examples are given for the Linux operating system. If you are using a different UNIX variant, see the appropriate operating system documentation.

The mounting of NFS shares on Linux can also be done using the *linuxconf* graphical utility (see Figure 25) included in most Linux distributions, including Red Hat. Select **Access nfs volume**, then use the **Add** button in the **NFS**

**volume** panel to enter in the MDC hostname and SANergy host mountpoint as shown.

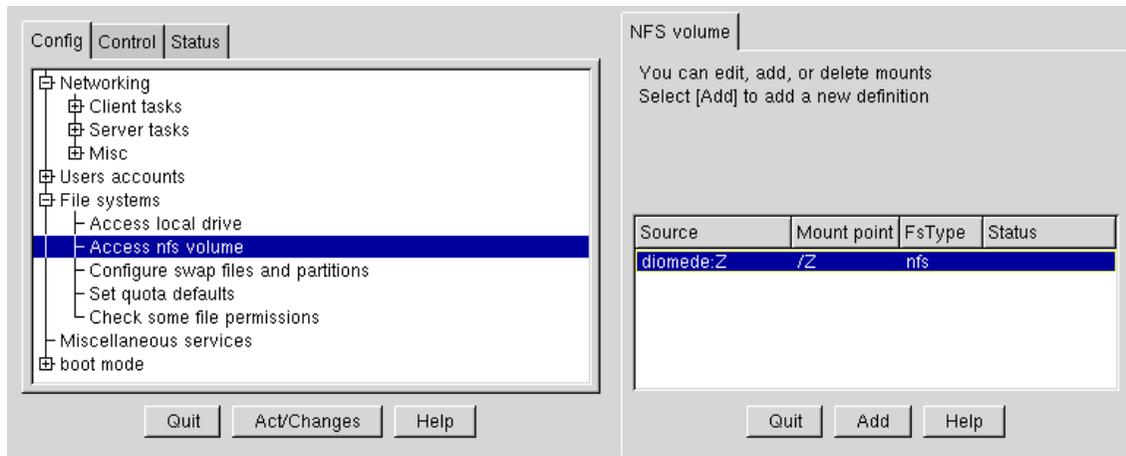


Figure 25. The linuxconf utility

Alternatively, for manual setup, edit the `/etc/fstab` file and create an entry for each share to be mounted. The entry will be in the form:

```
host:sharename mountpoint other-parameters
```

See the `fstab` man pages for details of parameters desired. For example, the following entry specifies mounting an NFS share from the MDC **diomede**:

```
diomede:Z /partition2 defaults 0 0
```

The share name parameter must match the drive letter of the exported NFS share on the Windows MDC when using SFU as the NFS server product.

Mount the NFS share, using the name you gave the entry in `/etc/fstab`:

```
# Mount /partition2
```

Test the mount point by logging on as the appropriate user account and attempting all intended access (read, write and execute). It is important to validate that the NFS sharing is working as intended before proceeding. This prevents you from perceiving problems that may be attributed to SANergy, which in fact are due to the NFS configuration.

## 2.2 Installing and configuring the MDC on Windows

In this section we describe the installation and configuration of the SANergy code for the Windows NT and Windows 2000 platforms. Since the procedure is identical on both operating systems, we will use the Windows 2000 computer **diomede** as our example.

We further assume at this point that the partition you want to manage with this MDC is properly set up on this machine. That means there is a connectivity over the SAN fabric to the storage system containing the partition, and that a drive letter has been assigned to the partition.

You can verify the status with the disk administrative tool. On Window NT, go to **Start > Programs > Administrative Tools > Disk Administrator**. On Windows 2000, choose **Start > Programs > Administrative Tools > Computer Management**. In the tree view, then select **Storage > Disk Management** to review the settings (Figure 26). Note the label of the drive you want to share with this MDC. In our installation the shared SAN disk will be **Disk3** with the label **MSSD12** and the drive letter **Z**.

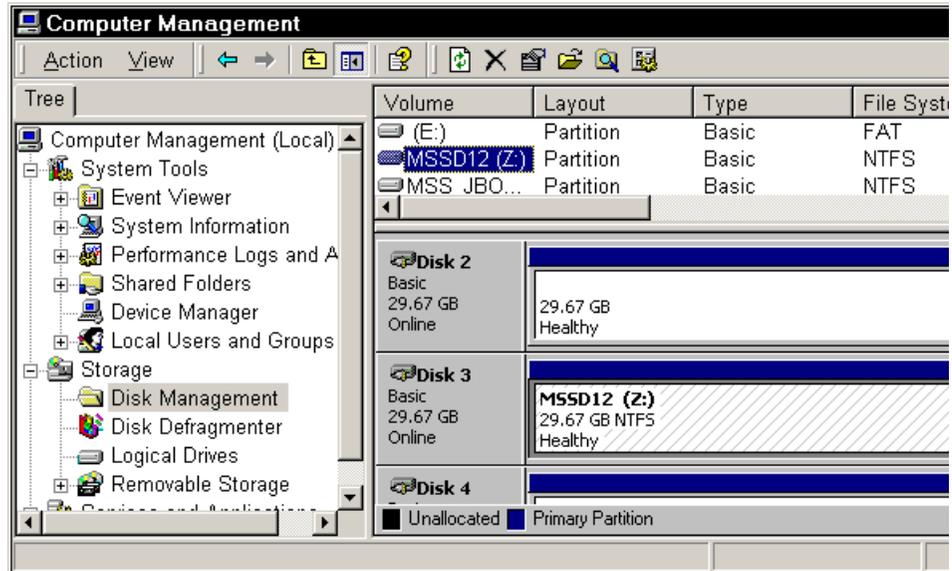


Figure 26. Disk Management on Windows 2000 MDC

You will first install the code from the original product CD which will include the required product key information for the license. Depending on release levels, you may then have to install a patch. You can find information on available patches and download them from the Web site:

<http://www.tivoli.com/support/sanergy/maintenance.html>

Before installing any patch, you should check the README file provided to see if there any special instructions associated with it.

Put the SANergy product CD into the CD-ROM drive — the installation setup will start automatically. Select the **Tivoli SANergy** option.



Figure 27. Initial SANergy installation screen

Follow the setup program and go through these steps:

1. Acknowledge license agreement
2. Choose installation directory
3. Choose program folder name
4. Type in user information.

A more detailed description of this procedure can be found in the *Tivoli SANergy Administrators's Guide*, GC26-7389. After these preliminary steps, the display shown in Figure 28 appears.



Figure 28. Select components to install

Here you can choose the specific program components to be installed. The SANergy software and documentation files are preselected by default. To enable SANergy for an environment monitored by SNMP, please explicitly install the SNMP package too. It is disabled by default.

After the install completes, some initial configuration screens appear automatically which have to be processed. First select the buses you want to be managed by SANergy (Figure 29). In our case, we have only one bus corresponding to our Fibre Channel HBA, but there might be multiple buses in our environment. Select only those which have access to the disk devices which you want to share volumes on.

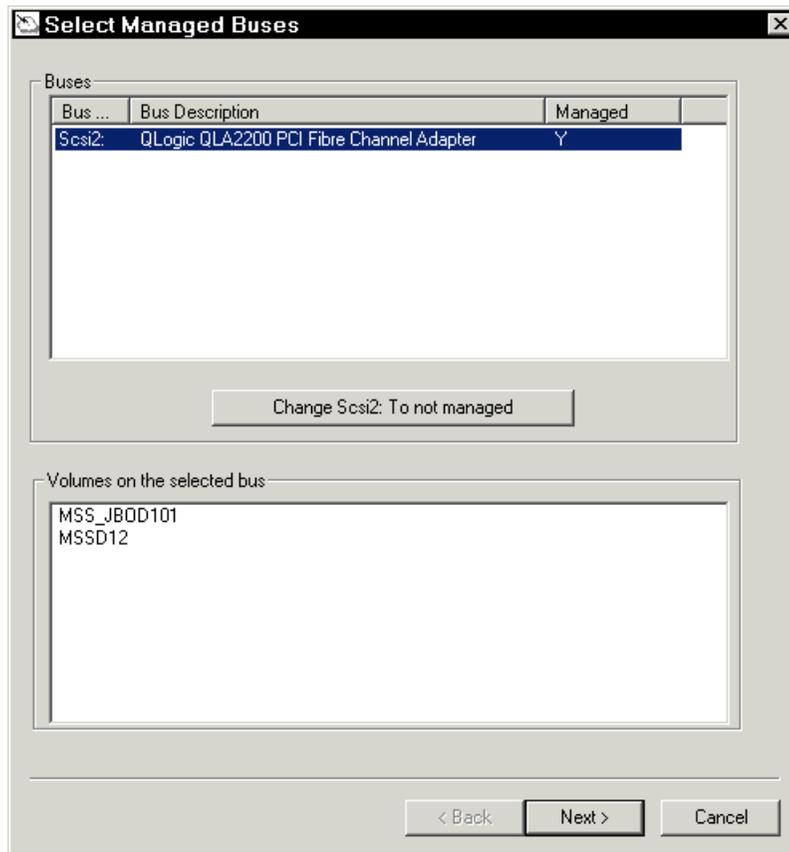


Figure 29. Select managed buses for Windows MDC

Select the bus that is connected to the storage which is to be shared and click on the **Change** button. Some HBAs have more than one bus per card. Be sure to select the correct one. When you click on the bus in the upper window you see in the lower screen the volume labels assigned to that specific bus. Only volumes with a drive letter assigned by the operating system are listed. After changing the bus from unmanaged to managed, you will see a pop-up window indicating that you should restart the computer. Click **OK** and proceed with the configuration. The restart will occur at the end of the configuration process.

In the next window you can assign devices to different MDCs (Figure 30).

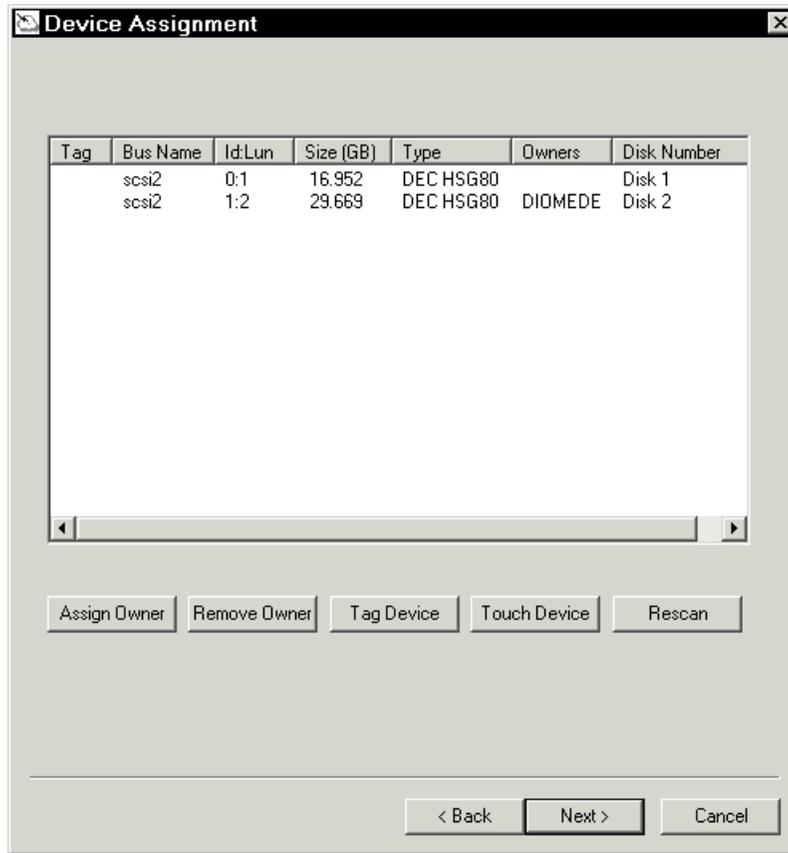


Figure 30. Device assignment for Windows MDC

Verify that the shared storage device is owned by the present computer (that is, the MDC which you are installing). To change the owner of a device, first remove the current owner (if any) by clicking **Remove Owner** and then assign the new owner with the button **Assign Owner**. In this example the screen shows two disks available on that bus. We set **diomede** as the owner of the first device.

At this point you can label the devices by clicking on **Tag Device**. The tag will just be a help for you to better identify the devices and has no further consequence. Another feature to help you associate the physical device among the listed LUNs is the **Touch Device** button. Select a device and click on this button to initiate I/O on the specified disk. This causes the selected disk's LED to flash.

The final installation step is to select the MDC for the shared storage (Figure 31).

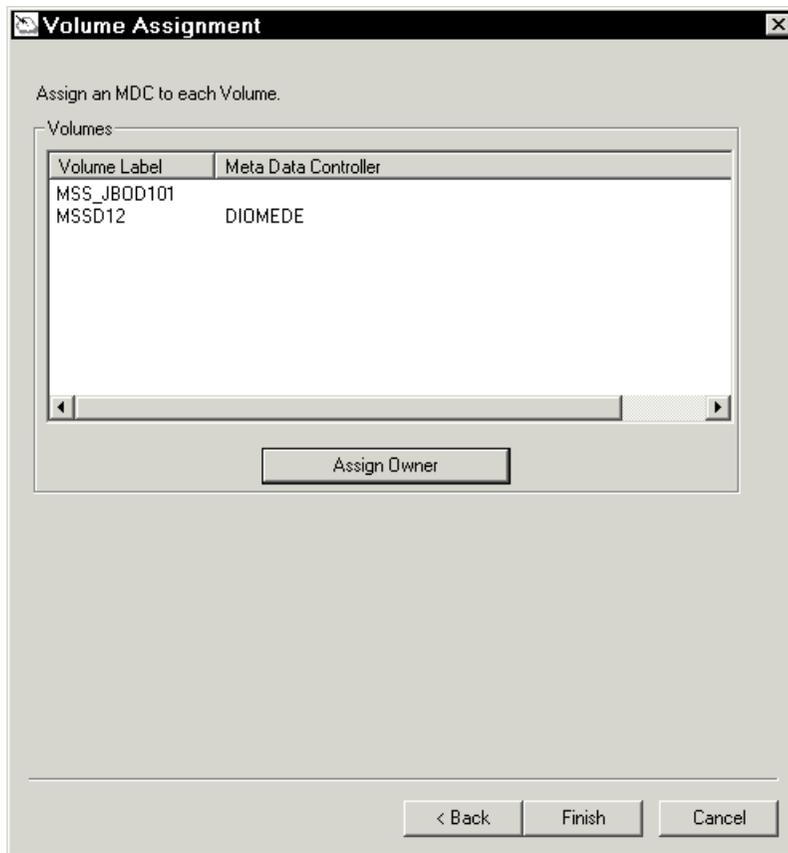


Figure 31. Volume assignment for Windows MDC

By default the owner of the storage device will become the MDC for a volume.

**Note**

You may have SAN storage devices attached to your present MDC that you do not want to share for whatever reason. Even so, it is necessary to assign ownership of these volumes to the MDC. Otherwise you will have no access to these volumes on the MDC itself. In this example the machine diomede will not be able to access the volume with the label **MSS\_JBOD101** but it will be able to act as MDC for the volume **MSSD12**.

After you have completed the volume assignment dialog box by clicking **Finish**, the system will ask you to reboot. After restart, you should install the latest patch available. The installation of the patch is identical to the main installation. All your configurations will remain set. After finishing installation, you can change these settings at any time using the SANergy setup tool. Start the program by choosing **Start > Programs > SANergy > SANergy Setup Tools**. The main window (Figure 32) contains tabs with the menu items you have configured during installation (**Select Managed Buses**, **Volume Assignment and Device Assignment**). Furthermore, there are two additional tabs: **Options** and **Performance Tester**, that allow you to tune, customize, and test the performance of the SANergy implementation. We will cover this topic in Chapter 4, “Performance” on page 89. To verify the MDC installation at this stage you can easily test the access to the SAN storage device by selecting **Performance Tester**.

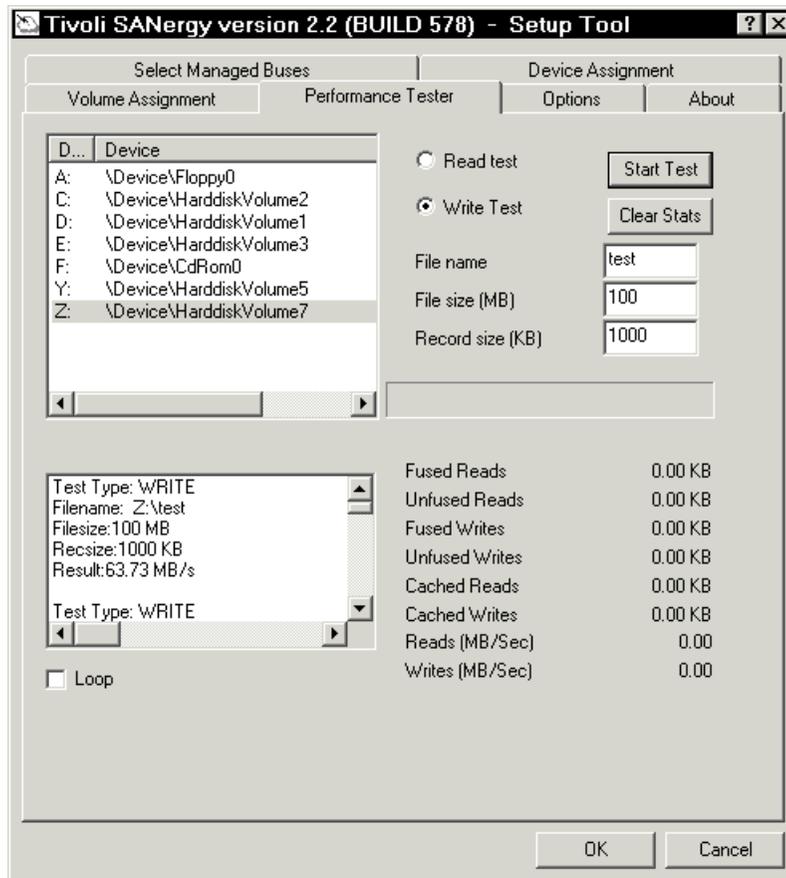


Figure 32. Performance Tester tab on Windows MDC

In the upper left-hand window, select the device owned by the MDC which will be the subject of the performance test (and which will be shared with SANergy hosts). To make sure that SANergy can write to this device, select the **Write Test** option, specify a file size and file name, and start the test with the **Start Test** button. This will run the test in a single iteration. To continuously run the test, check the **Loop** box.

SANergy now tries to write the defined file to the device. The bottom left-hand window shows the result of the test. You will not see any statistics on the right-hand side (for example **Fused Writes**), as these are only collected for SANergy host access (see Figure 39 on page 51 for an example). Do not proceed with the SANergy host installation if you have any connectivity problems at this point.

**Note**

The performance shown here seems to be mainly dependent on the I/O throughput of the storage device. Also, cache settings on the storage device can have increasing impact on data flow. That means that the first portions of data have high performance and then throughput falls back to the speed of physical disk performance after the cache is filled. To determine the performance that you will expect from hosts accessing the shared storage, run this test with a file whose size is larger than the cache settings on the storage device. Cache size and settings are discussed in detail in Chapter 4, “Performance” on page 89.

---

## 2.3 Installing and configuring SANergy hosts on UNIX and Windows

The final step in a SANergy implementation is the installation and configuration of the SANergy hosts. As described in section 2.1.2, “Client file sharing” on page 33, we are configuring both Windows and UNIX hosts on this Windows MDC (Figure 33). Compared with Figure 9 on page 21, we have now made the Fibre Channel connections from our SANergy hosts to the SAN to give them access to the shared storage devices.

Depending on how you chose to isolate the SANergy hosts in 2.1, “File sharing setup” on page 19, at this point you will either physically connect the SANergy host’s HBA to the SAN, or modify the switch zone so that it now has access to the disks to be shared. In our case we added the SANergy hosts to the zone already configured with the MDC and the shared disks. Depending on the SANergy host operating system, you may need to reboot it to gain access to the newly accessible drives.

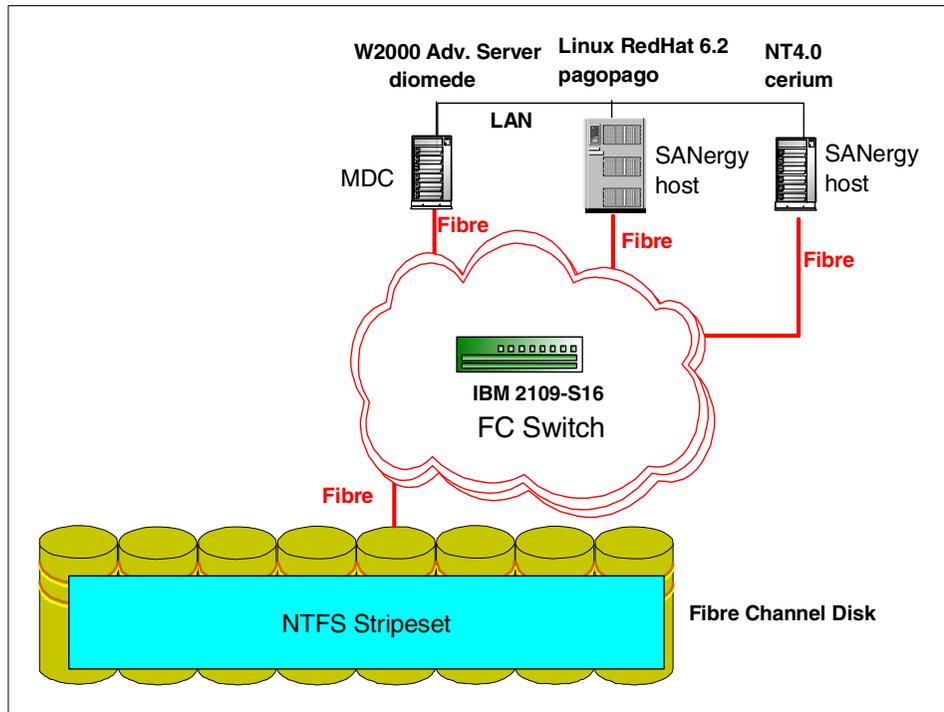


Figure 33. SANergy environment with a Windows MDC

### 2.3.1 Installing and configuring SANergy hosts on Windows

The installation of the Windows SANergy client software is identical to the installation of a Windows MDC. We document the installation of the SANergy software on our Windows NT machine cerium in this section. The specification of which machine is a SANergy MDC and which one is a SANergy host is done in the configuration procedure. An MDC is defined as the computer which owns a certain partition on the storage device. This same MDC could itself be a SANergy host for another partition controlled by another MDC in the SAN. We now demonstrate how to configure a machine to be a SANergy host for a partition owned by the MDC configured in section 2.2, “Installing and configuring the MDC on Windows” on page 37.

Before you run the setup, make sure the machine has proper connectivity to the shared SAN device. Check the Windows **Disk Administrator** tool and prove that this specific disk is listed. However, we do not recommend that you assign a drive letter and make it accessible to the operating system on the SANergy host because it is already owned by the MDC at this moment.

As soon as you establish physical connectivity from the host to the shared storage device, the Windows NT operating system will automatically assign a drive letter to that new device the next time you reboot. In Figure 34 you see the **Disk Administrator** tool before attaching the machine **cerium** to the SAN. Only the locally attached drive is visible.

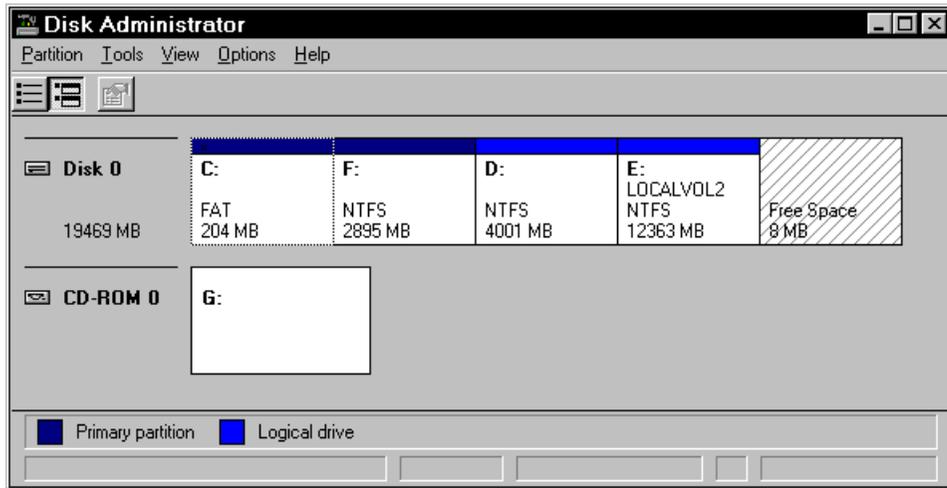


Figure 34. Disk Administrator tool before attaching the host to the SAN

After we connected the Fibre Channel cable from the HBA to the switch, thus connecting it to the SAN and rebooted, the system automatically added another device and provided it with a drive letter (Figure 35). This is because the MDC has already formatted that partition and this is recognized by this host.

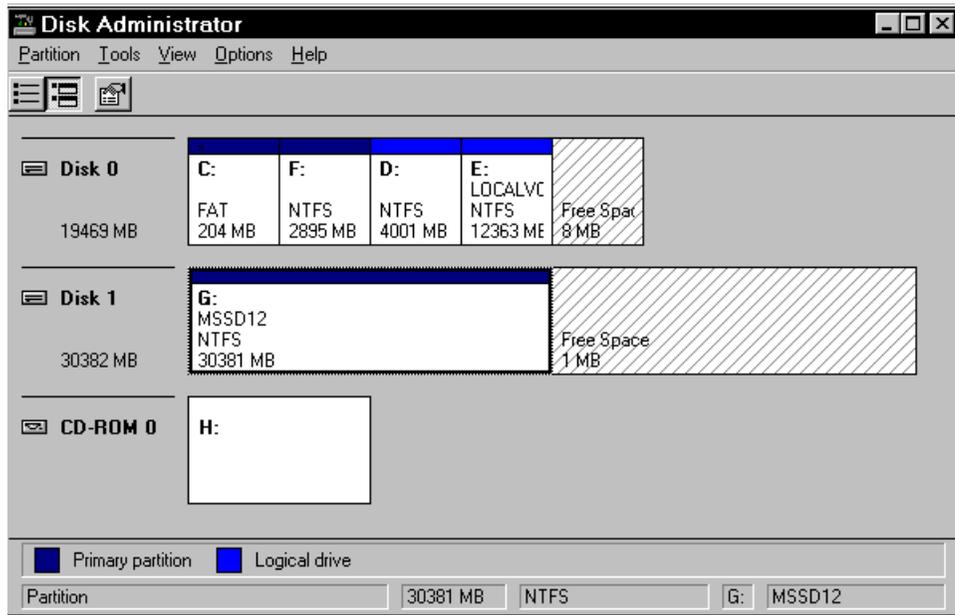


Figure 35. Disk Administrator tool showing a new device

We do not want this disk to be assigned a local drive letter, as we are using SANergy host access. To prevent NT from accessing and destroying data on that disk, we will unassign the drive letter. Right-click on the specified disk and choose **Assign Drive Letter** and then select the option **Do Not Assign Drive Letter**. The action will take effect immediately (Figure 36).

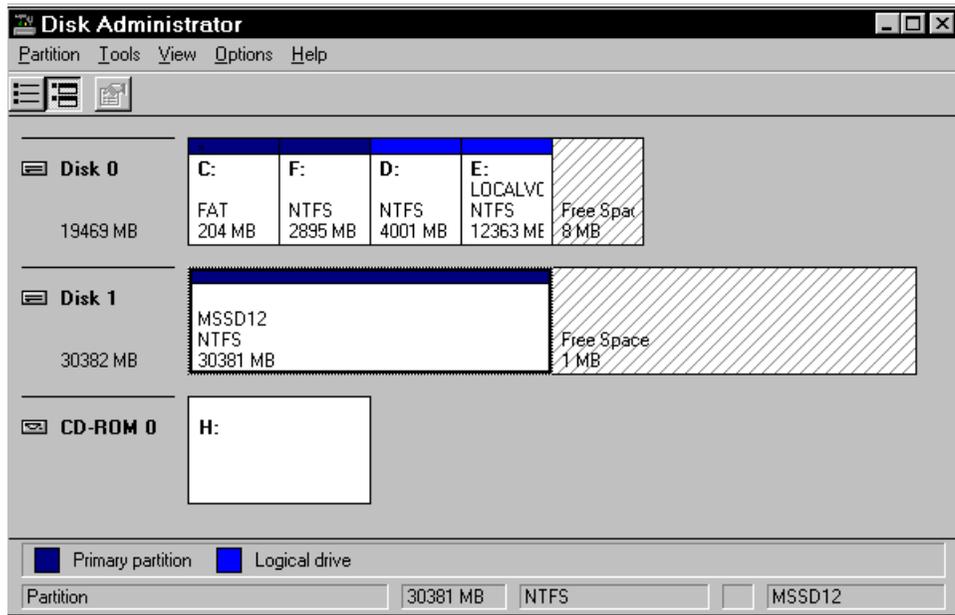


Figure 36. Disk Administrator after deleting drive letter

In the Explorer view you can now verify that you no longer have access to that disk. The disk icon for this device will have disappeared.

Now you can run through the installation process of SANergy as described in section 2.2, “Installing and configuring the MDC on Windows” on page 37. The difference between the SANergy host setup and the MDC setup in the configuration process comes with the configuration window as shown in Figure 37.

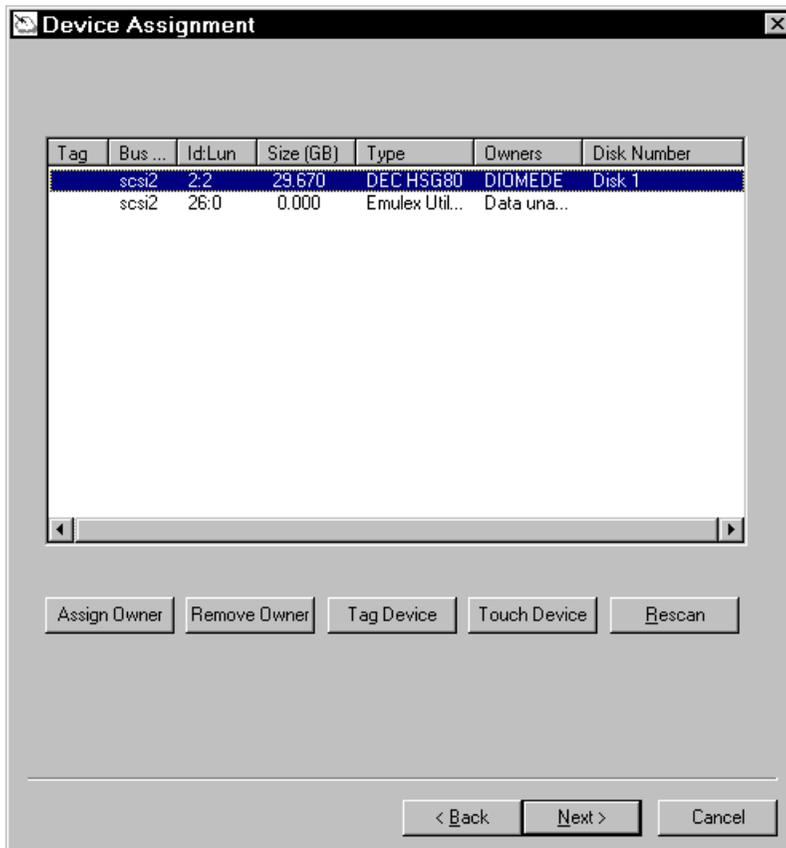


Figure 37. Device assignment to a SANergy host

The installation process identifies diomede as the owner of the SAN storage device.

Click **Next** to go to the window in Figure 38 which shows diomede also as the MDC (which we set up in the MDC installation in Figure 31 on page 42)

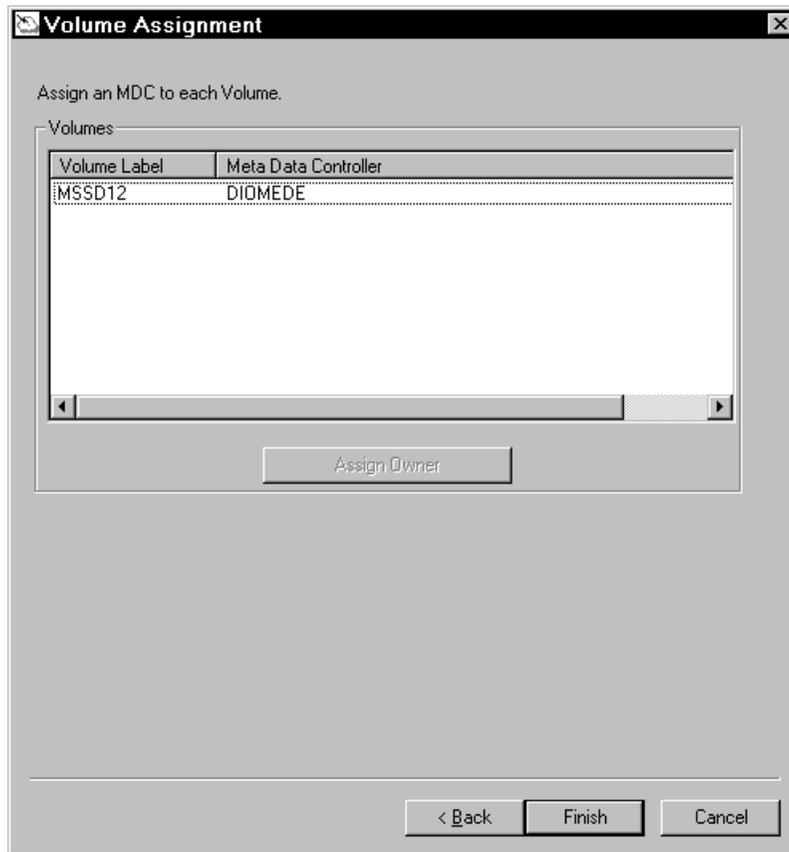


Figure 38. Volume Assignment on the SANergy host

Now the installation is completed and the host **cerium** will be able to fuse the shared volume on **diomede**. That means it redirects I/O to and from the shared volume to the host over the SAN. There is an easy way to test if the host can directly read and write data to the shared volume over the SAN. Open the SANergy setup tool. Open **Start > Programs > SANergy > SANergy Setup Tool** and select the tab **Performance Tester** (Figure 39).

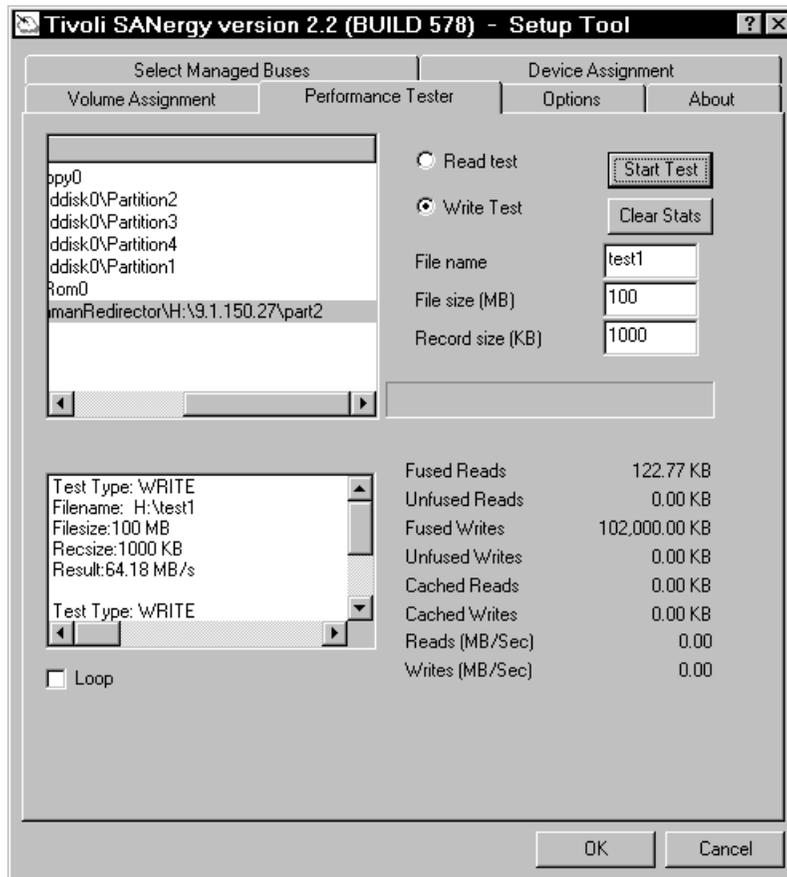


Figure 39. Verify installation with the performance tester

If the shared drive on the MDC is properly mounted on the host, you will see the mapped partition on the upper left window. Highlight this partition to perform a write test on the device. Select a dummy file name and a representative file size and click the **Start Test** button. On the right bottom side you can see the measured values for fused data throughput for this attempt. If you do not see any updated statistics for reads and writes, this means that the SAN is not being used for access.

## 2.3.2 Installing and configuring SANergy hosts on UNIX

This section covers the installation and configuration of SANergy on a UNIX host. This example is given using the Linux client and the GUI interface, where appropriate. An example of the UNIX host installation and configuration using the command line interface is included in 3.3.1, “Installing and configuring SANergy hosts on UNIX” on page 74. As a prerequisite for installing SANergy on UNIX, you should have a Netscape browser with JavaScript support installed.

### 2.3.2.1 Install SANergy base code

Insert and mount the SANergy CD-ROM (see your operating system documentation or `mount` man page for details on how to mount a CD on your version of UNIX). Change to the directory where you mounted the CD-ROM, and from there change to the directory containing the UNIX install script:

```
cd file_sharing/unix
```

Run the install script located in this directory:

```
./install
```

You will be prompted to validate the Operating System detected. The installation runs automatically and starts the configuration tool via your Netscape Web browser. The initial screen asks you to enter or validate the detected product key. Choose **Accept** to proceed to the main screen (Figure 40).

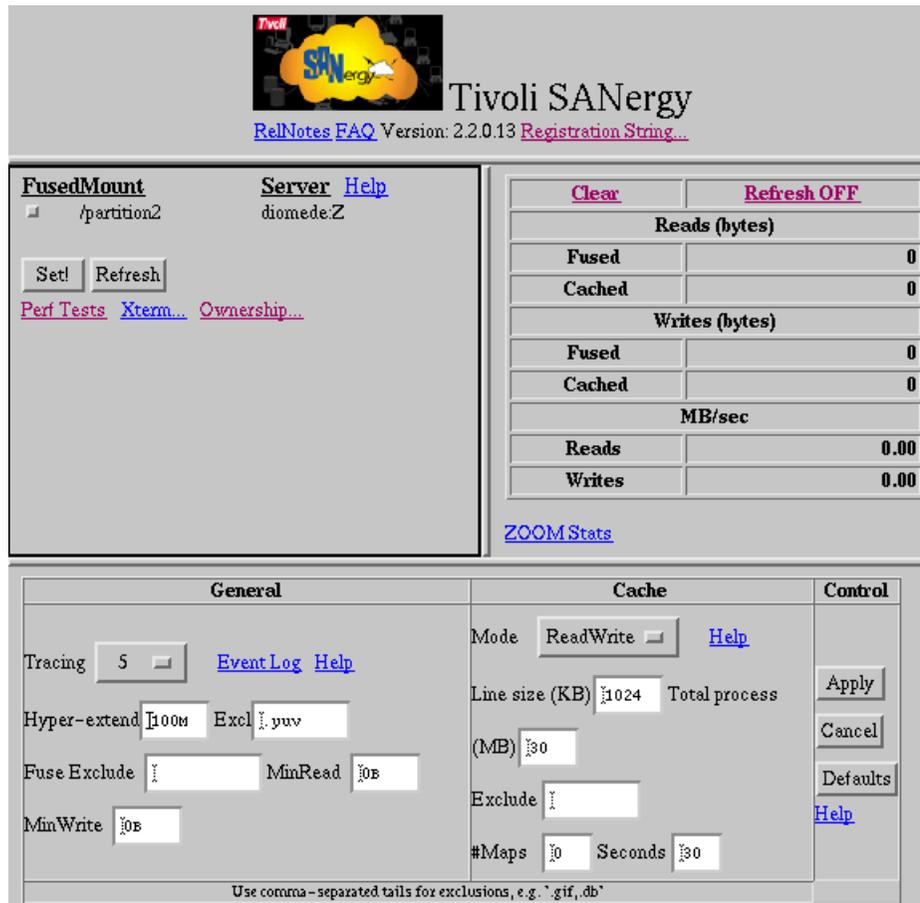


Figure 40. Tivoli SANergy main screen on Linux

For more detailed examples of basic installation refer to the *Tivoli SANergy Administrator's Guide*, GC26-7389.

### 2.3.2.2 Select the mountpoints to fuse

The upper left quadrant of the SANergy main window contains a list of all NFS shares currently mounted on the system (see Figure 41). Select those you wish to fuse and click on **Set!**



Figure 41. Mount point before fusing

Once you have selected to fuse the mount points, the buttons will be recessed, but there is no other indication that these file systems are now fused(Figure 42).



Figure 42. Mount point which is fused

In order to validate that fusing was successful, select the **Perf Tests** hyperlink to compare access speeds after fusing. Highlight the mount point to be tested from the **Select Volume** list box (see Figure 43). Click the **Write** button to initiate a write performance test. When the test is complete, the panel will contain the performance of the test. Also note that the statistics on the upper right quadrant will refresh to indicate the new values.

The screenshot shows a 'Performance Tester' window. On the left, there is a 'Select Volume' list box with options: /boot, /dev/pts, /mnt/cdrom, /part3, and /Z. The /Z option is selected. To the right of the list box, the text '/Z' and '31.39MB/s' is displayed. Below the list box are 'Read' and 'Write' buttons. Below these buttons, it says 'Wrote 288MBs in 9.17 secs' and 'Help Fuse panel'. On the right side of the window, there is a statistics table with 'Clear' and 'Refresh OFF' buttons at the top. The table shows the following data:

| Reads (bytes)  |               |
|----------------|---------------|
| Fused          | 0             |
| Cached         | 323,584       |
| Writes (bytes) |               |
| Fused          | 1,786,957,823 |
| Cached         | 457,728       |
| MB/sec         |               |
| Reads          | 0.00          |
| Writes         | 0.00          |

At the bottom of the statistics panel, there is a 'ZOOM Stats' link.

Figure 43. Performance Testing and Statistics panel

Once these tests have been done on all fused mount points, installation of the SANergy host software is complete.

### **2.3.2.3 Using SANergy with a process**

In order to run a process or application which will access SANergy managed file systems, it is necessary to first set some environment variables so that the SANergy libraries are used. In the SANergy installation directory (default:/usr/SANergy) there are two scripts to help you set these environment variables. If you are running a C-shell, use `SANergycshsetup`. If you are using a Korn-shell, use `SANergyshsetup`. We recommend including the appropriate invocation scripts in the startup scripts for all applications requiring access to SANergy managed file systems.

Once the environment variables are set correctly, start the process as normal. We give some more guidance on setting environment variables in different situations for Linux (which equally apply to other UNIX variants) in A.1.5, “SANergy and the SCSI disks” on page 266.



## Chapter 3. SANergy with a UNIX MDC

This chapter describes the implementation of a SANergy file sharing configuration with UNIX as the MDC. Installation and setup of SANergy on the MDC as well as on UNIX and Windows hosts is covered. We start with a discussion of file sharing setup on the MDC and different types of SANergy hosts.

---

### 3.1 File sharing setup

SANergy relies on either NFS or CIFS based file sharing. Therefore, it is strongly recommended that client and server file sharing, over a normal TCP/IP connection, be set up and tested before installing SANergy. The file sharing clients are those that will eventually become SANergy hosts. The server will become the MDC. Setup of an NFS server and Samba CIFS server are covered. SANergy host file sharing setup is covered following that.

The SANergy hosts do not need to be physically connected with access to the storage device over the SAN at this point. In fact, we recommend that at this stage, you allow only the machine which will be the MDC to have access to the shared disks. If the SANergy host or client machines have access to the shared disk, there is the risk of them writing their own signatures or otherwise trying to take ownership of the drives.

There are two ways to prevent the SANergy hosts from accessing the disks: You can either disconnect their HBAs from the SAN, or you can use the zoning capabilities of your fabric switch to prevent access to the switch. Zoning means grouping together ports on your switch so that the devices connected to those ports form a virtual private storage network. Ports that are members of a group or zone can communicate with each other but are isolated from ports in other zones.

The method that you choose will depend on your particular SAN topology, and what, if any, other access is required to SAN devices by the SANergy hosts. Disconnecting the HBA is a “shotgun” approach — simple but effective. Zoning the shared disks is more elegant but may be more complex to implement, and may not even be possible if you are in a non-switched environment, or your switch does not support zoning. In our setup, we used zoning for the IBM SAN Fibre Channel switch. Instructions on how to configure this are available in the redbook *Planning and Implementing an IBM SAN*, SG24-6116. We configured the MDC in a zone with access to the shared disks and put the SANergy hosts into another zone.

Once you have isolated the SANergy hosts from the disks, check that the LAN connection is properly configured, so that there is TCP/IP connectivity between SANergy MDC and hosts.

The following sections are based on the configuration in Figure 44, which shows only the MDC (sol-e) with SAN connection to the shared disk.

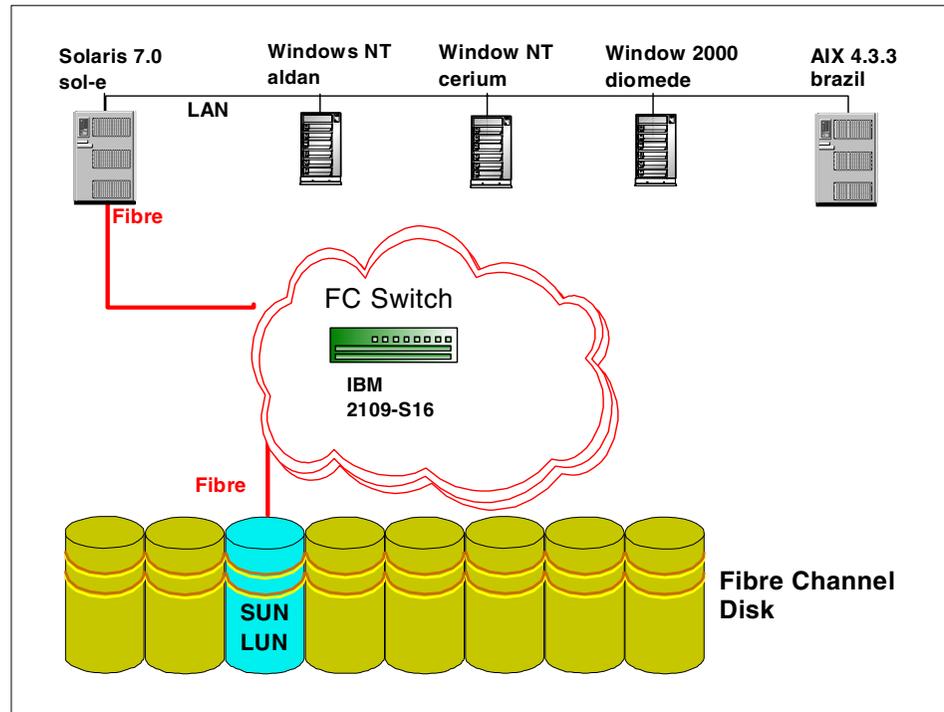


Figure 44. UNIX server file sharing configuration

### 3.1.1 UNIX server file sharing

UNIX NFS and Samba services are both described for our SANergy environment. Setup considerations are presented in this section. NFS is used to share file systems between UNIX systems. Samba is used to share a UNIX file system with Windows clients through the SMB, also known as CIFS, interface.

We use the SUN Solaris platform as our UNIX NFS server, later the SANergy MDC, and the hostname for our Solaris system is `sol-e`. The Solaris file system on the Fibre Channel disk LUN owned by `sol-e` was called `/d102`.

### 3.1.1.1 NFS server setup

The steps to prepare the NFS server on UNIX, with some SUN/Solaris specifics, are:

1. Connectivity from the UNIX system to the SAN disk(s) must first be established. Refer to your operating system device driver and HBA documentation for details on how to set this up.
2. Follow normal procedures of the UNIX operating system to define a file system. The Solaris `format` and `newfs` commands were used at this point to update the partition table and create the file system.
3. Verify that the file system is mounted; if it is not already, mount it. The file system we use in the following examples is `/d102`. The `mount` command with no parameters will display the file systems currently mounted. In Solaris, adding an entry for the new file system to the `/etc/vfstab` file will cause it to be mounted automatically each time the system is booted.

```
# mount
/proc on /proc read/write/setuid on Mon Apr  9 08:57:22 2001
/ on /dev/dsk/c0t0d0s0 read/write/setuid/largefiles on Mon Apr  9 08:57:22 2001
/usr on /dev/dsk/c0t0d0s6 read/write/setuid/largefiles on Mon Apr  9 08:57:22 2001
...
/d102 on /dev/dsk/c3t0d2s7 read/write/setuid/largefiles on Mon Apr  9 08:57:22 2001
# cat /etc/vfstab
#device      device      mount      FS      fsck  mount  mount
#to mount    to fsck    point      type    pass  at boot options
#fd          -          /dev/fd fd    -      no    -
/proc        -          /proc  proc   -      no    -
/dev/dsk/c0t0d0s1 -          -        swap   -      no    -
/dev/dsk/c0t0d0s0 /dev/rdisk/c0t0d0s0 /        ufs    1      no
..
/dev/dsk/c0t0d0s4 /dev/rdisk/c0t0d0s4 /tmp     ufs    2      yes
/dev/dsk/c3t0d2s7 /dev/rdisk/c3t0d2s7 /d102    ufs    2      yes
```

4. The permissions for directories, including the mount point, and files to be shared from the server must be set appropriately. On Solaris, we set the permission of the mount point to 777 with the UNIX `chmod` command. It should be noted that default permissions for files created on the NFS server may not allow access by users on other systems. In other words, the permissions may need to be reset after a file is created.
5. Next, the file system must be NFS exported. Exporting is the process of making the file system available to NFS clients. This is accomplished on Solaris with a share definition for the file system, which is made in the file `/etc/dfs/dfstab`. The share entry for our file system is shown on the last line:

```
# cat /etc/dfs/dfstab

#      Place share(1M) commands here for automatic execution
#      on entering init state 3.
#
#      Issue the command '/etc/init.d/nfs.server start' to run the NFS
#      daemon processes and the share commands, after adding the very
#      first entry to this file.
#
#      share [-F fstype] [ -o options] [-d "<text>"] <pathname> [resource]
#      .e.g,
#      share -F nfs -o rw=engineering -d "home dirs" /export/home2
share -F nfs -o rw=brazil:diomede:cerium -d "sol's 102 disk" /d102
```

6. Restart the NFS service if necessary. The Solaris commands to do this are `/etc/init.d/nfs.server stop` and `/etc/init.d/nfs.server start`. You can verify the NFS service daemons are running with the `ps -ef` command. The two server daemons are *mountd* and *nfsd*. The NFS exports are checked with the Solaris `share` command. See the following example:

```
# /etc/init.d/nfs.server stop
# /etc/init.d/nfs.server start
# ps -ef | grep nfs
root  159      1  0 19:00:04 ?        0:00 /usr/lib/nfs/lockd
daemon 161      1  0 19:00:04 ?        0:00 /usr/lib/nfs/statd
root 13455     1  0 07:48:41 ?        0:00 /usr/lib/nfs/nfsd -a 16
root 13453     1  0 07:48:41 ?        0:00 /usr/lib/nfs/mountd
root 15038 15026   0 09:23:51 pts/6    0:00 grep nfs
# share
-                /d102    rw=brazil:diomede:cerium  "sol's 102 disk"
```

The file system is now ready to be mounted by the NFS clients.

### 3.1.1.2 Samba CIFS server setup

Samba was used on our UNIX MDC to provide CIFS (otherwise known as SMB) access for Windows clients to the file system we were sharing. The level of Samba used on our Solaris system was 2.0.7, and we downloaded Samba from the Web site:

<http://www.samba.org/>

**Note:** Documentation on Samba is also available on this Web site.

We downloaded, installed, and successfully used the Samba 2.0.7 pre-compiled binaries — source code is also available. We followed the installation instructions at the Web site and in the README file. Samba will use a number of TCP/IP ports which you can see in the `/etc/services` file:

```
swat          901/tcp          # Samba configuration tool
netbios-ssn   139/tcp          # Samba
netbios-ns    137/udp          # Samba
```

After Samba is installed, the steps for sharing a volume through CIFS are as follows:

1. Check that the Samba SMB service is running by using the `ps -ef` command as shown in the following example. The presence of the `smbd` daemon process shows that the SMB service is running. If `smbd` is not running, it should be started now. On Solaris this is done by issuing the command `/etc/init.d/samba start`.

```
# ps -ef | grep smb
   root 17236 16941  0 11:37:03 pts/6    0:00 grep smb
sanuser1 16678  158  0 11:03:54 ?        0:00 smbd
```

2. Samba uses two UNIX users: one for administration and one for CIFS, or drive mapping, authentication. For administration, we used the Solaris root user. For CIFS authentication or login, we used the id `sanuser1`. When performing the subsequent steps, you will need to know the ids and passwords for both of these functions.
3. Samba configuration is done through the Samba Web Administration Tool, known as `swat`. This is accessed through a browser using `<samba_host>:901` as the URL. In this URL, 901 is the default port used for `swat` and may be different in your environment. The browser can be run from the system on which Samba is running or from another system that has network access to the Samba server. See Figure 45 for an example of accessing `swat` and the welcome screen.

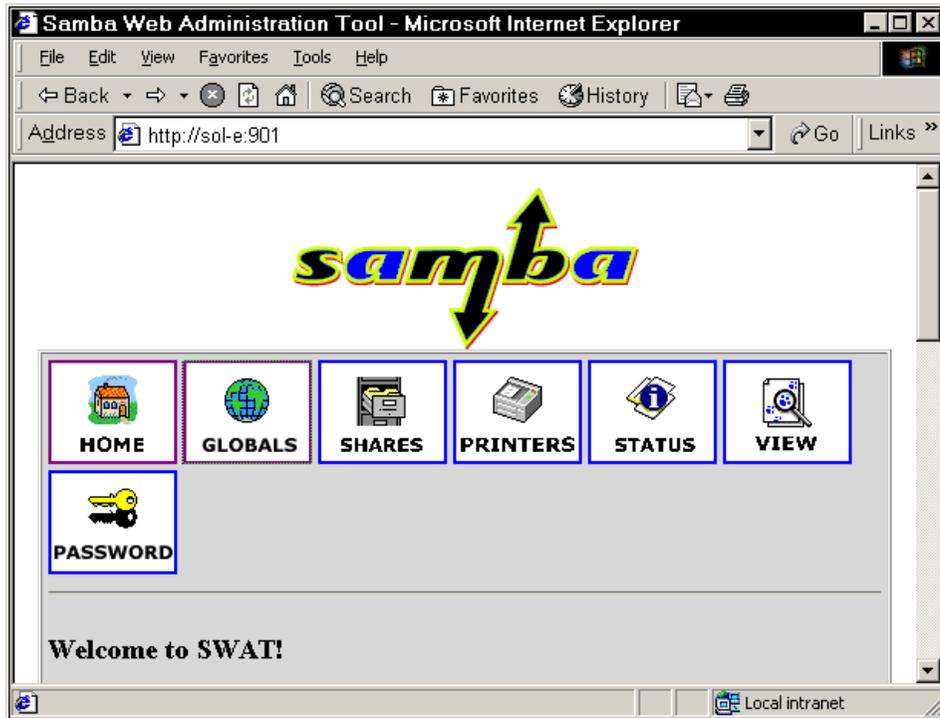


Figure 45. Samba swat home page

4. The next step is to configure the global settings. We used the default workgroup name and specified the userid needed for CIFS authentication. Again, this is a UNIX id that is defined on the Samba server. We restricted Samba access to hosts that were on one of two subnets, 193.1 or 9.1. We overrode the default log file specification because we wanted logs written to the `/var/log/samba` directory with the accessing host name as the extension of the log file.

After making the necessary changes, you would click the **Commit Changes** button at the top of this screens. Figure 46 shows the global settings.

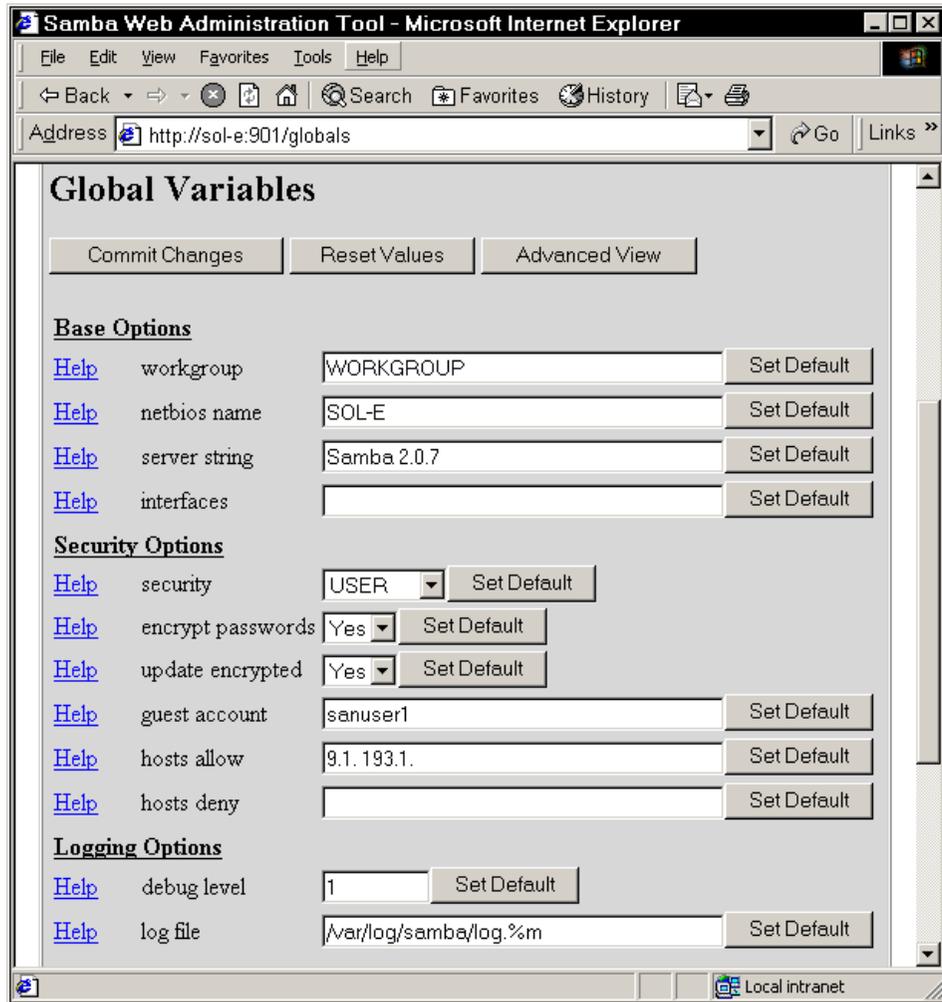


Figure 46. Samba global settings

- Now you need to define the shares. First enter the new share name in the entry field next to the **Create Share** button and then click the button. This will bring up a screen as shown in Figure 47 allowing you to specify various options for that share. The path parameter is the UNIX file system mount point. When all parameters are properly set, click the **Commit Changes** button. This example shows the options we defined to allow us to share the file system /d102.

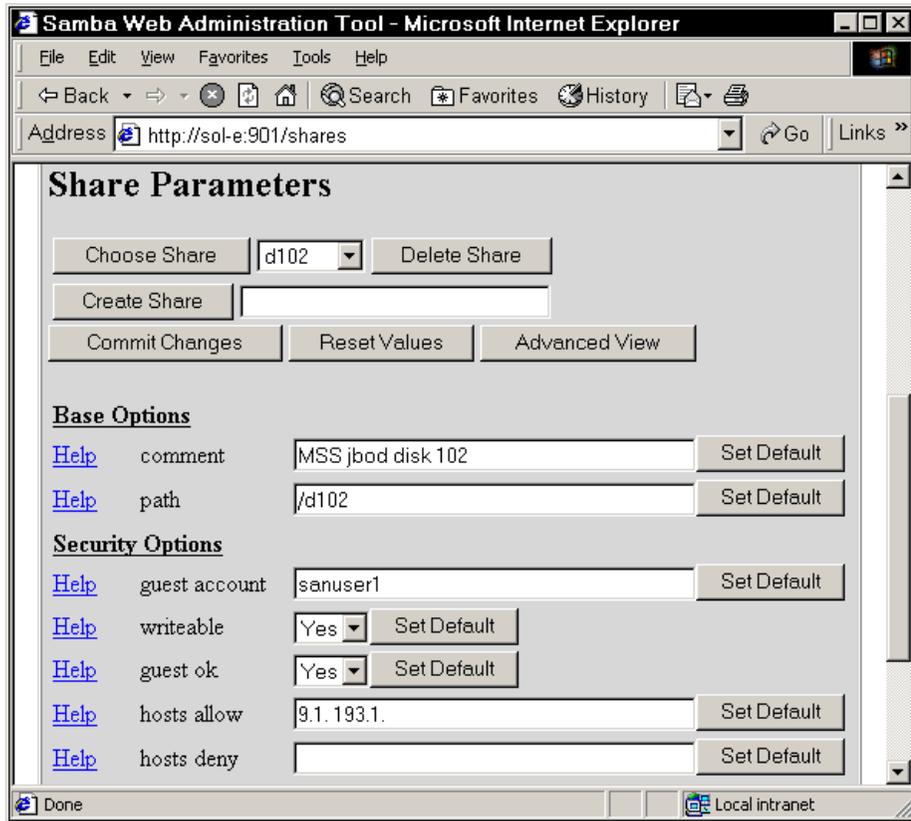


Figure 47. Samba share creation

6. Your Samba configuration can be reviewed by clicking the **View** button at the top of the screen. Figure 48 shows an example of this. You can also see these options by viewing the Samba configuration file, which in our example was `/opt/samba/lib/smb.conf`.

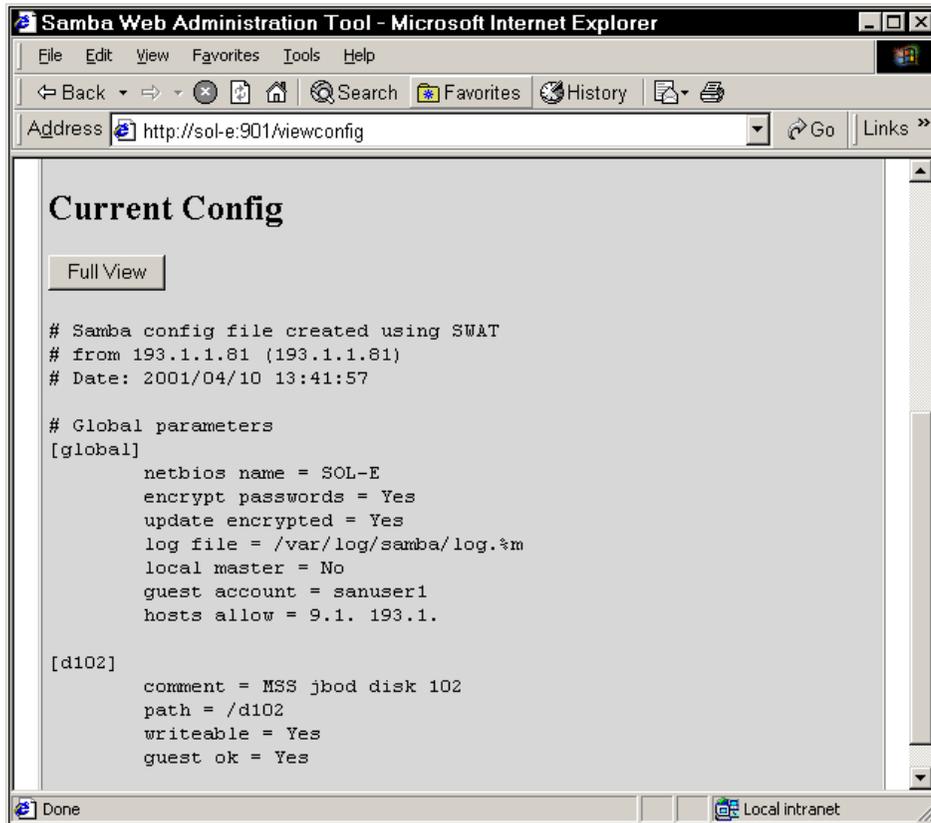


Figure 48. Samba configuration summary

7. The file systems shared out by Samba should now be ready to use from the CIFS clients. The running status of the Samba server and various controls are accessible from the *swat* Server Status screen, shown in Figure 49. The Samba daemon *smbd* can be stopped and restarted from this screen. The Active Connections, Active Shares, and Open Files information displayed are very useful in monitoring the configuration.

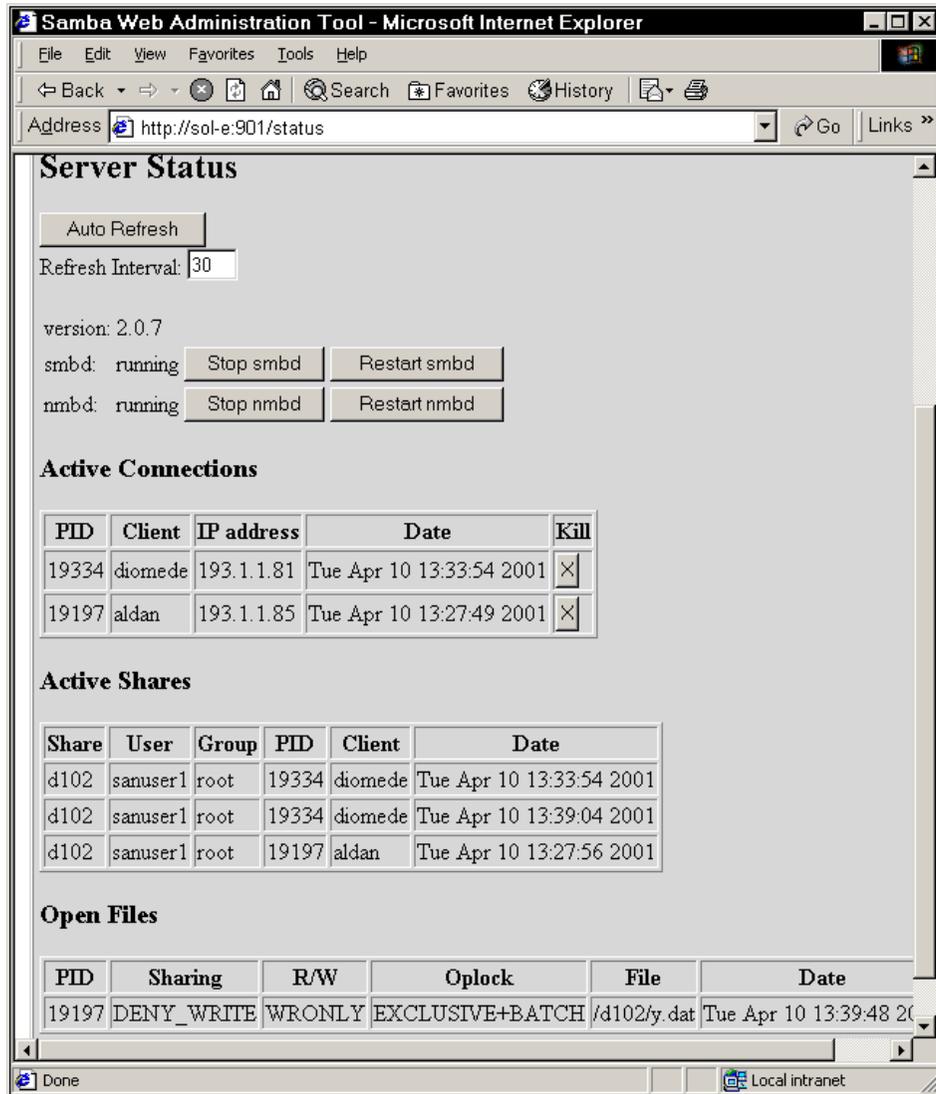


Figure 49. Samba server status

### 3.1.2 Client file sharing

This section shows how to configure clients to share files from the UNIX host.

### 3.1.2.1 UNIX NFS client setup

NFS client setup on UNIX is a simple process. We used AIX as our NFS client and SANergy host platform. The setup process consists of these steps:

1. On AIX, check that the fileset `bos.net.nfs.client` is installed with the `lspp` command.

```
# lspp -l bos.net.nfs.client
Fileset              Level  State      Description
-----
Path: /usr/lib/objrepos
bos.net.nfs.client   4.3.3.30 COMMITTED Network File System Client
Path: /etc/objrepos
bos.net.nfs.client   4.3.3.30 COMMITTED Network File System Client.
```

2. Create the mount point directory with the UNIX `mkdir` command. Change the permissions on this directory per your security requirements with `chmod`. The permissions of the mount point can be checked at any time with `ls -al <mountpoint>`. See the following example.

```
# mkdir /d102
# chmod 777 /d102
# ls -al /d102
total 16
drwxrwxrwx  2 root    system      512 Apr 10 09:36 .
drwxr-xr-x 48 bin     bin         1536 Apr 10 09:36 ..
```

3. Verify that the file system desired for mounting has been exported and is available to your client by issuing the command: `showmount -e <NFS server TCP/IP hostname>`. We see below that file system `/d102` has been exported by `sol-e` and has been made available to the system we are preparing, `brazil`.

```
# showmount -e sol-e
export list for sol-e:
/d102 brazil,dionede,cerium,rainier
```

4. We recommend that you use the `mount` command as described in the SANergy Administrator's Guide to mount the file system. In order for the file system to get mounted whenever the system is rebooted, this command should be put into a shell script that is run at system startup.

Alternatively, on AIX, the command `mknfsmnt` can be used to define NFS mount and cause the file system to be mounted at command execution and whenever AIX is booted. This is the second command shown here.

```
# mount -o acregmin=0,acregmax=0,actimeo=0 sol-e:/d102 /d102
#
# /usr/sbin/mknfsmnt -f '/d102' -d '/d102' -h 'sol-e' -m 'NFS' -t 'rw'
-w 'bg' -U '0' -u '0' -T '0' -A
```

5. Verify the file system has been mounted correctly by using a command such as: `df <mountpoint>` or `mount | grep <mountpoint>`. Both of these are illustrated below.

```
# df /d102
Filesystem      512-blocks      Free %Used    Iused %Iused Mounted on
sol-e:/d102     34986272 34359344   2%         17    1% /d102
# mount | grep /d102
sol-e    /d102          /d102          nfs3    Apr 10 10:32 bg,intr,acregmin=
0,acregmax=0,actimeo=0,rw
```

6. Check access to directories and files of the file system by issuing the command: `ls -al <mountpoint>`

```
# ls -al /d102
total 277076
drwxrwxrwx  4 root    system      512 Apr 10 10:03 .
drwxr-xr-x  48 bin     bin         1536 Apr 10 09:36 ..
-rwxr--r--  1 sanuser1 sanergy     25 Apr 09 18:57 aix.file
-rw-r--r--  1 nobody   system      0 Apr 09 12:32 aix.root.file
-rw-r--r--  1 nobody   system     10 Apr 09 14:26 aix.root2
-rw-r--r--  1 nobody   nobody      0 Apr 10 09:59 dt.fil
-rw-r--r--  1 sanuser1 sanergy     0 Apr 09 13:24 filesa
drwx----- 2 root    system     8192 Apr 09 11:16 lost+found
```

7. Switch to the UNIX user who will run SANergy using the `su` command. We used `sanuser1`.
8. Test the ability to write to the filesystem from that user by using a command such as `touch` or `vi`.

```
# su sanuser1
$ cd /d102
$ touch aix.sanuser1.file
$ ls -al aix.san*
-rw-r--r--  1 sanuser1 sanergy     0 Apr 10 2001 aix.sanuser1.file
```

The UNIX NFS client is now ready for SANergy.

### 3.1.2.2 Windows CIFS client setup

This section covers the mapping of the shared volume from the Solaris MDC sol-e to a Windows host. The volume was exported as a CIFS share by Samba. The share name is \\sol-e\d102. The host that will get access to this share is the Windows NT machine cerium.

From a Windows system you can map a shared volume by choosing **Tools-> Map network drive** in the Windows Explorer window. This opens the window in Figure 50.

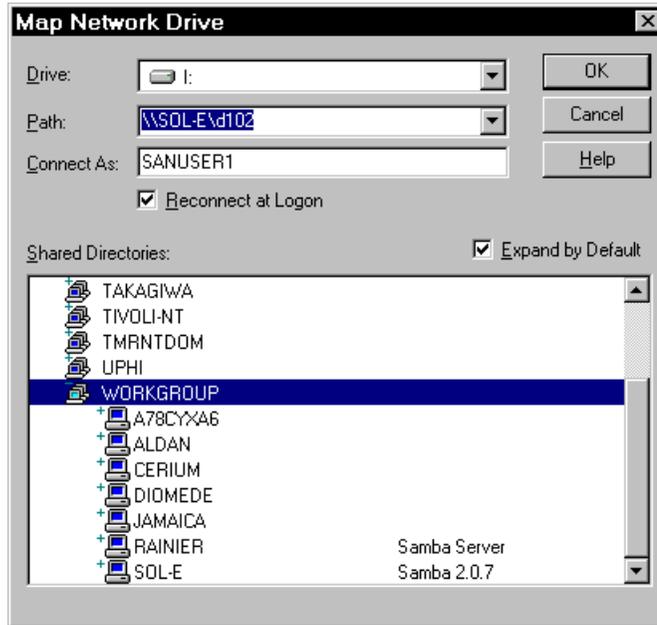


Figure 50. Map network drive on Windows systems

To map the shared volume, enter the share name in the field **Path**. The share name has the form: \\computername\sharename. Also, enter a username (in this example, sanuser1) in the field **Connect As**. This username must be defined as a valid CIFS user to the CIFS server on the machine to which you want to connect. In our environment, sanuser1 is defined as the **guest account** in the Samba configuration, as shown in Figure 46 on page 63.

Click **OK** to establish the connection. You will be prompted for the password for the user sanuser1. You can verify the access to the new share in the Explorer window. A new drive icon labeled **d102** was added to the view, as shown in Figure 51.

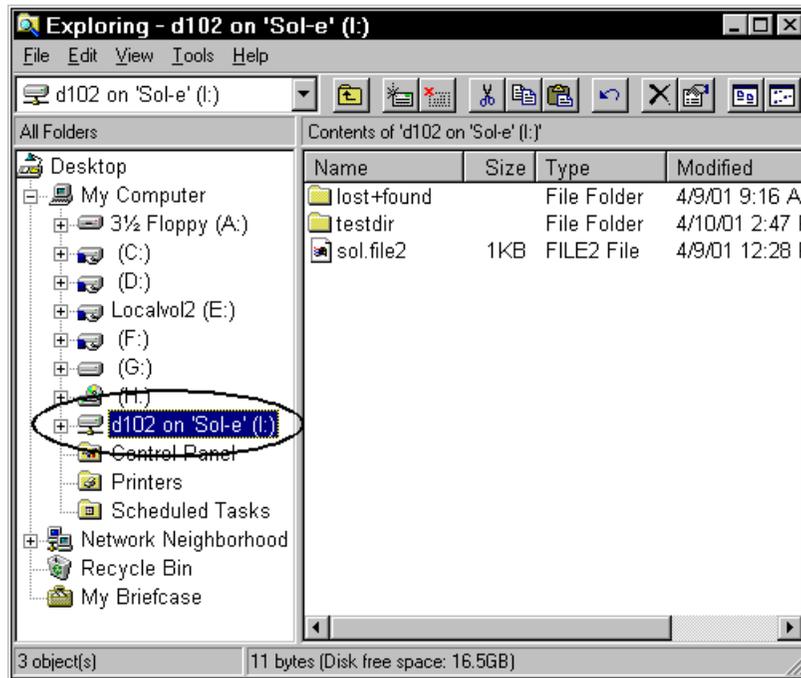


Figure 51. Access to a new share from a UNIX system

---

### 3.2 UNIX MDC installation and configuration

Implementation of a SANergy MDC on the UNIX platform is covered in this section. First we describe the installation of the SANergy base and patch with specific platform considerations. Then we describe the configuration of the SANergy MDC. Figure 52 illustrates our SANergy configuration.

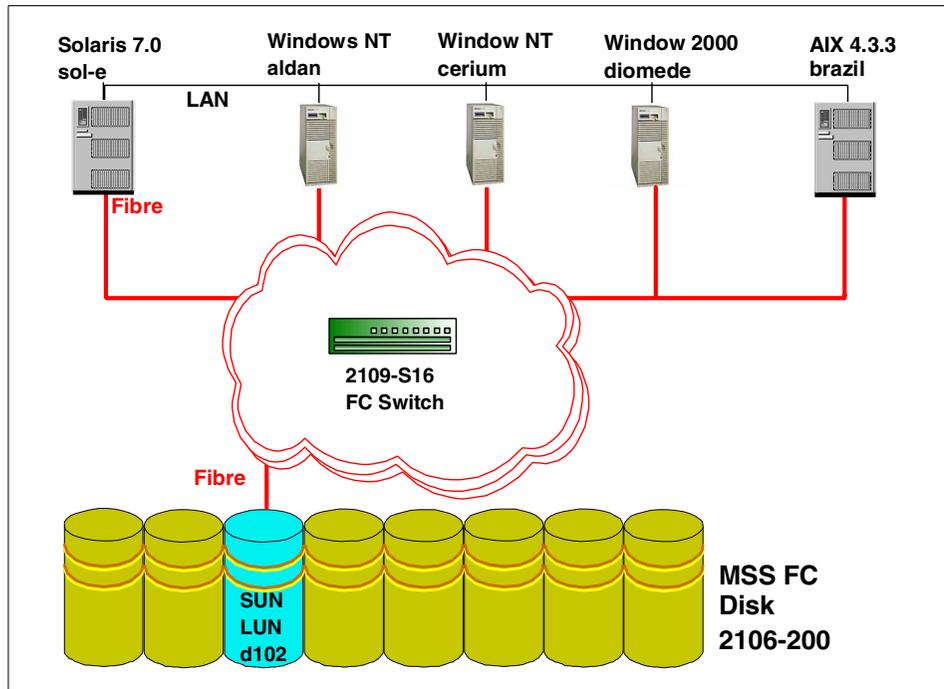


Figure 52. UNIX SANergy file sharing configuration

### 3.2.1 SANergy base installation

The base SANergy code for UNIX was installed on our Solaris MDC according to *Tivoli SANergy Administrator's Guide*, GC26-7389. The only exception to the procedure documented in the manual was the directory structure of the SANergy installation CD. The UNIX software is in the directory "file\_sharing/unix/" rather than "unix\_sanergyfs/".

### 3.2.2 SANergy patch installation

We then installed SANergy patch 13 on our MDC, bringing the code level to 2.2.0.13. The most recent SANergy patch is available through the Tivoli Web site at:

[www.tivoli.com/support/sanergy/maintenance.html](http://www.tivoli.com/support/sanergy/maintenance.html)

### 3.2.3 Linux system considerations

We discovered a number of hints and considerations for working with Linux. These are documented in Appendix A, "Linux with a SAN and SANergy" on page 261.

## 3.2.4 Solaris system considerations

There are a few things you need to consider when installing the SANergy MDC on Solaris.

### 3.2.4.1 Device tree

The SANergy software installation on Solaris over-writes the Solaris `/dev/dsk` and `/dev/rdisk` entries for disks it takes ownership of. We reconstructed the device structure by running the Solaris commands `drvconfig`, `disks`, and `devlinks`.

### 3.2.4.2 32-bit kernel

SANergy 2.2 must run in 32-bit mode on Solaris. Therefore, after performing the preceding steps, we set up our Sun system to use the 32-bit Solaris kernel as the default and then rebooted. The figure below shows how to set the 32-bit kernel as the default and how to check the setting.

```
# ./eeprom boot-file=kernel/unix
# ./eeprom | grep boot-file
boot-file=kernel/unix
```

## 3.2.5 MDC configuration

Next we describe some setup steps for configuring the MDC.

### 3.2.5.1 Samba usage setup

Samba must run with the SANergy environment set properly. This is accomplished by modifying the Samba start script to set the SANergy environment variable. The highlighted lines in the following figure are those we added for SANergy to the Samba startup script `/etc/init.d/samba`. You will need to make the same changes to the `/etc/rc2.d/S99samba` if you are using that to start Samba instead of `/etc/init.d/samba`.

```

#!/sbin/sh
#
# Start Samba SMB file/print services

# Set environment
PATH=/usr/bin:/sbin:/usr/sbin export PATH

# Samba directory
SAMBA_DIR=/opt/samba
SAMBA_SMBD_DEBUG_LEVEL=0
SAMBA_NMBD_DEBUG_LEVEL=0
SAMBA_SMBD_LOG=/var/opt/samba/log.smb
SAMBA_NMBD_LOG=/var/opt/samba/log.nmb

# Additions for SANergy
LD_PRELOAD=/usr/lib/libSANergy.so
export LD_PRELOAD

```

### 3.2.5.2 SANergy setup

The SANergy UNIX MDC setup procedure, after installing the software, is a matter of taking ownership of the volumes to be shared. The manual recommends using the GUI to take ownership. We were not able to set ownership from the GUI, so we used the SANergyconfig tool CLI instead. We issued the `owner` command for our Solaris disk to be shared. See the example below. It is recommended that the tag assigned, which is the third parameter (in our example, set to `d102`), should match the file system mount point. The reason for this is to make it easier for the SANergy clients to relate the disks they are fusing to the mounted file systems.

```

Enter Command: owner|c3t0d2s2|sol-e|d102
+++++
Processing command - owner|c3t0d2s2|sol-e|d102...
OK
=====

Enter Command: owner
+++++
Processing command - owner...
0      d102      c3t0d2s2      DEC      HSG80      V85SZG03713358
      sol-e
1      c3t0d3s2      DEC      HSG80      V85SZG03713358
      ALDAN
=====

```

This example also shows the output from a query of the owners of disks known to the UNIX SANergy system, by again issuing `owner`, this time with no operands.

---

### 3.3 SANergy host installation and configuration

Here we show installation and configuration steps for SANergy hosts on UNIX and Windows. Depending on how you chose to isolate the SANergy hosts in 3.1, “File sharing setup” on page 57, at this point you will either physically connect the SANergy host’s HBA to the SAN, or modify the switch zone so that it now has access to the disks to be shared. In our case we added the SANergy hosts to the zone already configured with the MDC and the shared disks. Depending on the SANergy host operating system, you may need to reboot it to gain access to the newly accessible drives.

#### 3.3.1 Installing and configuring SANergy hosts on UNIX

Installation and configuration of SANergy on the AIX platform is presented using the SANergy command line interface. As of SANergy Version 2.2, the Netscape based SANergy graphical user interface is not yet functional on an AIX host. The SANergy GUI for UNIX installation and setup was shown in 2.3.2, “Installing and configuring SANergy hosts on UNIX” on page 52.

##### 3.3.1.1 SANergy base and patch installation

The base SANergy code for UNIX was installed on our AIX host according to *Tivoli SANergy Administrator’s Guide*, GC26-7389. The only exception to the information in the manual was the directory structure of the SANergy installation CD. The UNIX software is in the directory “file\_sharing/unix/” instead of “unix\_sanergyfs/”.

The procedure for base installation relies on the Netscape browser being available to enter the registration key. If Netscape fails to run for whatever reason, the registration key will not be set, and SANergy will not be usable. Since Netscape could not be used for the installation on our AIX system, we had to enter the key through the command line interface, `SANergyconfig`. This should be done immediately after the installation completes. The following example shows the output of the SANergy `install` command, with the Netscape error displayed. Then it shows running the configuration command `SANergyconfig` to complete the installation correctly.



```

# lsdev -Cc disk
hdisk0 Available 10-60-00-8,0 16 Bit SCSI Disk Drive
hdisk1 Available 10-60-00-9,0 16 Bit SCSI Disk Drive
hdisk2 Available 10-68-01      Other FC SCSI Disk Drive
hdisk3 Available 10-68-01      Other FC SCSI Disk Drive
...
hdisk10 Available 10-68-01      Other FC SCSI Disk Drive
hdisk11 Available 10-68-01      Other FC SCSI Disk Drive
hdisk12 Available 10-68-01      Other FC SCSI Disk Drive
hdisk13 Available 10-68-01      Other FC SCSI Disk Drive

```

If the disks are physically connected to AIX but not accessible, that is, they do not show up with the `lsdev` command, you may be able to make them available to AIX with the `cfgmgr` command. In our case, the volumes we use with SANergy are `hdisk10`, `hdisk11`, `hdisk12`, and `hdisk13`. The relation of the `hdisk` numbers to the SANergy MDC volumes is shown in step 3 in this section.

The remote file systems, made available through NFS exports, should also be mounted if necessary before continuing.

#### Note

Unreliable, nonexistent or unnecessary NFS mounts should be eliminated before starting the SANergy configuration. Bad mounts or non-responsive NFS servers can cause very long delays when running SANergyconfig commands such as `fuse` or `owner`.

Here are the steps to configure the UNIX SANergy host:

1. Start the command line interface as shown here. Please note that it is important to set up the SANergy environment prior to issuing the `SANergyconfig` command. This can be done with the `SANergyshsetup` script:

```

# . /usr/SANergy/SANergyshsetup
# /usr/SANergy/SANergyconfig
Consider using the graphical 'config' program instead of this.
Enter Command:

```

2. Verify the version and that the registration key is set.

```

Enter Command: ver
+++++
Processing command - ver...
OK|2.2.0.12
=====

Enter Command: key
+++++
Processing command - key...
OK|KEY|UNIX-SFS-eQQY6Y3P78PeQ8P3QR-TSI
=====

Enter Command:

```

3. The UNIX disks can be related to the MDC volumes as follows. Only the MDC owned volumes will show a hostname in the right-hand column and the SANergy tag (for example, rd1, d11) to the left of the rhdisk entry. Although it is not required, the tag should easily tie back to the file system mount point. This information is not necessary for fusing, but helps in relating the hdisks to SAN volumes. In our case, we are accessing rhdisk13, which is tagged d102.

```

Enter Command: owner
+++++
Processing command - owner...
  0          rhdisk1 NO INFO
  1          rhdisk2 NO INFO
  2          rhdisk3 NO INFO
  3          rhdisk4 NO INFO
  4          rhdisk5 NO INFO
  5          rhdisk6 NO INFO
  6   rd1    rhdisk10   NO INFO rainier
  7   d11    rhdisk11   NO INFO ALDAN
  8   d12    rhdisk12   NO INFO DIOMEDE
  9   d102   rhdisk13   NO INFO sol-e
=====

```

4. Fusing is done as shown. The parameter passed to the fuse command is the mount point of the file system as it is mounted on the SANergy client.

```

Enter Command: fuse|/d102
+++++
Processing command - fuse|/d102...
OK
=====

```

- You can easily determine which volumes are under SANergy control by using the command `fused`. The output from the command shows the desired volume, `/d102`, is mounted. It also shows two other volumes mounted on this SANergy client from two Windows MDCs, `aldan (/G)` and `diomed (/Z)`.

```

Enter Command: fused
+++++++
Processing command - fused...
/d102,sol-e,/d102|/G,aldan,/G|/Z,diomed,/Z|
=====

```

- You can verify that reads and writes to the fused disk are handled by SANergy by using the `stats` command. In this test, we cleared the SANergy statistics, copied a large file from CD-ROM to the fused volume, and then displayed the statistics. The statistics show: `fusedwrites`, `fusedreads`, `cachewrites`, and `cachereads`, respectively.

```

Enter Command: clear stats
+++++++
Processing command - clear stats...
OK
=====

Enter Command: stats
+++++++
Processing command - stats...
0|0|0|0
=====
(Copied a large file to the SANergy volume here ...)

Enter Command: stats
+++++++
Processing command - stats...
411646|0|2496512|0
=====

```

- It may be desirable to limit SANergy usage to files of a minimum size. The `minfused` command controls this function. To illustrate its use, we first verified that the minimum fused size was 0 and the statistics were 0. We then ran the UNIX `cat` command against a 9+ KB file on a fused volume and displayed the SANergy statistics again. Next we set the minimum fused size to 10 KB and reran the `cat` command. Finally, we displayed the statistics again and saw that this time SANergy did not handle the reads. The pertinent SANergyconfig commands and output are as follows.

```

Enter Command: minfused
+++++
Processing command - minfused...
OK|MINFUSED|0|0
=====

Enter Command: stats
+++++
Processing command - stats...
0|0|0|0
=====
                                     <A cat command was issued at this point on another screen.
Enter Command: stats
+++++
Processing command - stats...
0|19630|0|4096
=====

Enter Command: minfused|10k
+++++
Processing command - minfused|10k...
OK|MINFUSED|10240|10
=====
                                     <The same cat command was issued at this point.
Enter Command: stats
+++++
Processing command - stats...
0|19630|0|4096
=====

```

The unchanged statistics show that SANergy is no longer managing the file tested, since it was less than 10K in size.

**Attention**

The SANergy environment variables must be set in each and every session using commands whose I/O is to be managed by SANergy. In other words, the SANergy libraries must be used. Instructions for accomplishing this can be found in *Tivoli SANergy Administrator's Guide*, GC26-7389. Briefly, either the SANergyshsetup or SANergycshsetup scripts must be run; which depends on the UNIX shell you are using.

- 8. SANergy can be told to stop managing a volume, that is, fusing can be disabled, by issuing the `unfuse` command:

```
Enter Command: unfuse|/d102
+++++
Processing command - unfuse|/d102...
OK
=====
```

9. To make SANergyconfig settings permanent on a UNIX host, they need to be placed into the SANergyconfig.txt file, which resides in the SANergy installation directory. On the AIX platform the default is /usr/SANergy; on the Solaris platform, it is /opt/SANergy. An example of our SANergyconfig.txt is shown below. The SANergyconfig.txt file is executed automatically when SANergy is started. It can also be executed at any time by issuing the command from either the /usr or /opt directory as appropriate: SANergy/SANergyconfig<SANergyconfig.txt .

```
cat SANergyconfig.txt
# SANergyconfig.txt - input file for SANergyconfig
# rules - no tabs and no leading spaces and # are comments
#
# The owner command should be first in the list.
#
# nfs volumes to accelerated by SANergy
# Example:
#     fuse|/ddd           where /ddd is the nfs mount point
#     fuse|/eee
#
log|0|/usr/SANergy/mclog
fuse|/d102
exit
```

**3.3.2 Installing and configuring SANergy hosts on Windows**

The platform used here is Windows NT 4.0. The SANergy host cerium will use CIFS, otherwise known as drive mapping, to access the data from the MDC sol-e.

Before we ran the SANergy installation we made the shared SAN disk available to the NT machine. After a reboot we start the **Disk Administrator** tool to verify that there is proper connectivity over the SAN to that volume. Open **Start-> Programs-> Administrative Tools-> Disk Administrator** in the Windows task bar. This first access after changing the physical configuration on the device initializes a rescan of the disks. The pop-up window (Figure 53) will appear, indicating that Windows found a disk with no signature. You are asked if you want to write a signature to this new disk.

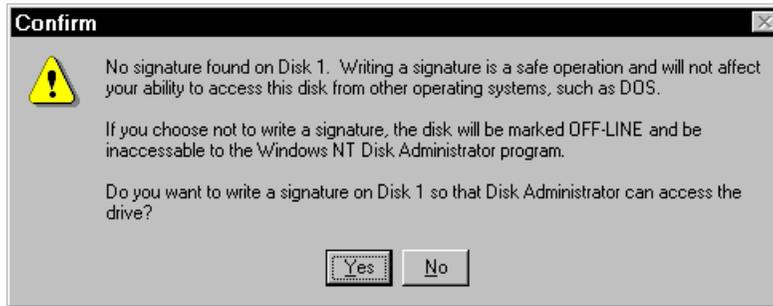


Figure 53. Create signature on new device

Click the **No** button to prevent Windows NT from writing any data to the disk which is already owned by the UNIX MDC. This will lead you to the window shown in Figure 54.

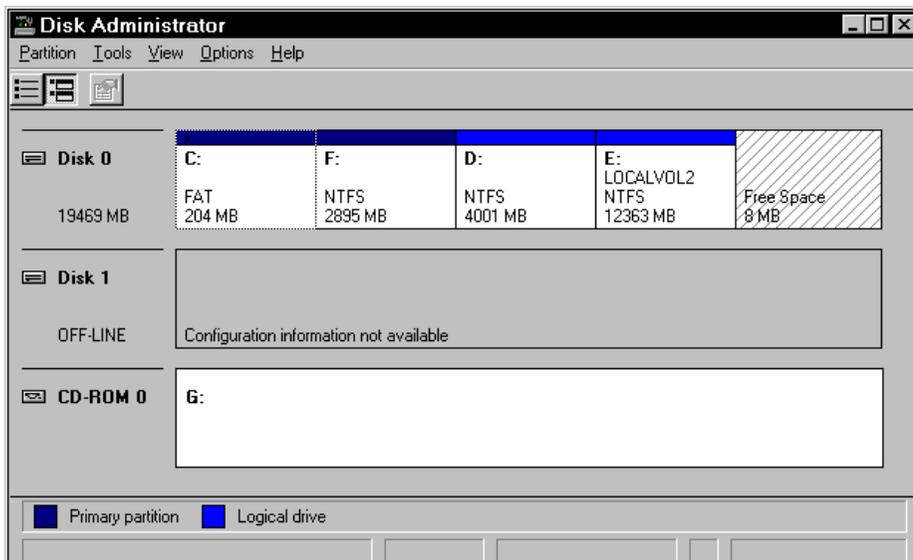


Figure 54. UNIX formatted disk in the Disk Administrator view

The new drive is offline, and Windows NT cannot see the label or the format of the volume at this point. This is what we want to see right now.

To install the SANergy software, start the *setup* program from the CD and run through the installation procedure which is a straightforward process. For more detailed information about the installation, see the *Tivoli SANergy Administrator's Guide*, GC26-7389.

After the installation is completed, the InstallShield automatically starts a configuration dialog which begins with the device assignment in Figure 55.

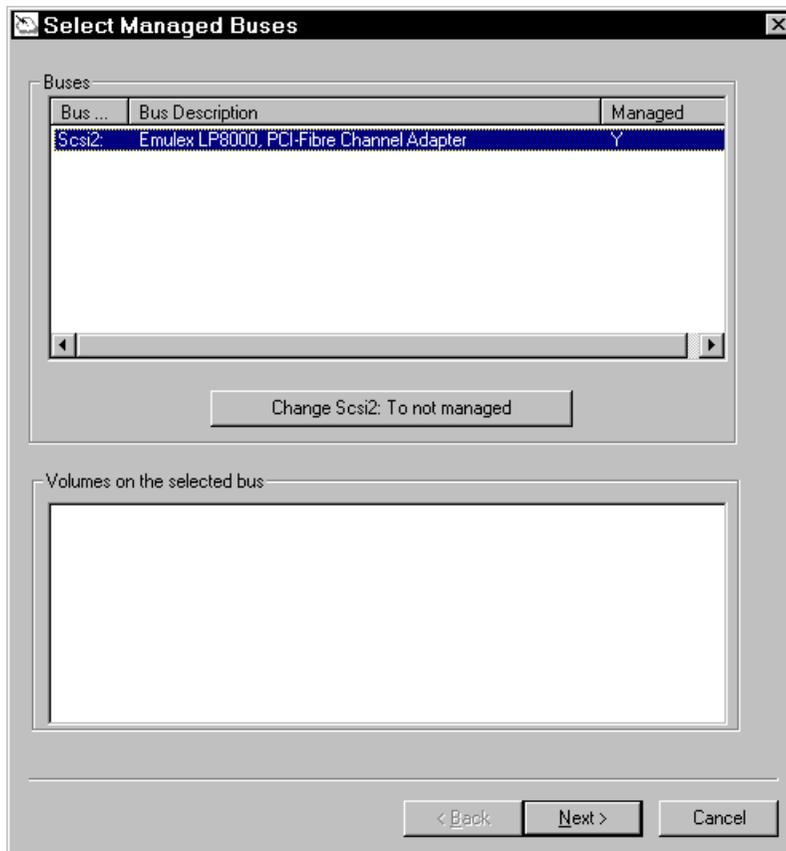


Figure 55. Select Managed Buses dialog box

By default, all buses are unmanaged by SANergy. You have to enable the bus with the shared SAN device attached. Highlight the appropriate bus and click the **Change** button. Click **Next** to get to the **Device Assignment** window (Figure 56).

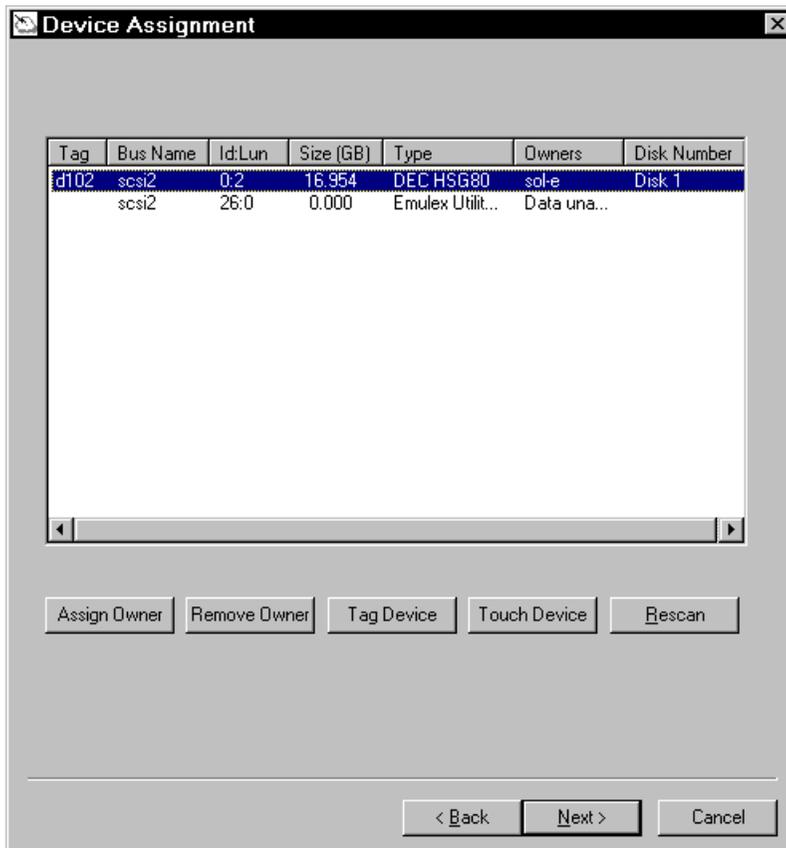


Figure 56. Device Assignment on the Windows host

The SANergy configuration process identifies the current owners, if there are any, of the SAN storage devices attached to this machine. The owner of the device in this example is sol-e. As we are setting up a SANergy host at this time, cerium does not take ownership of any devices. Proceed to the final configuration dialog box (Figure 57) by clicking **Next**.

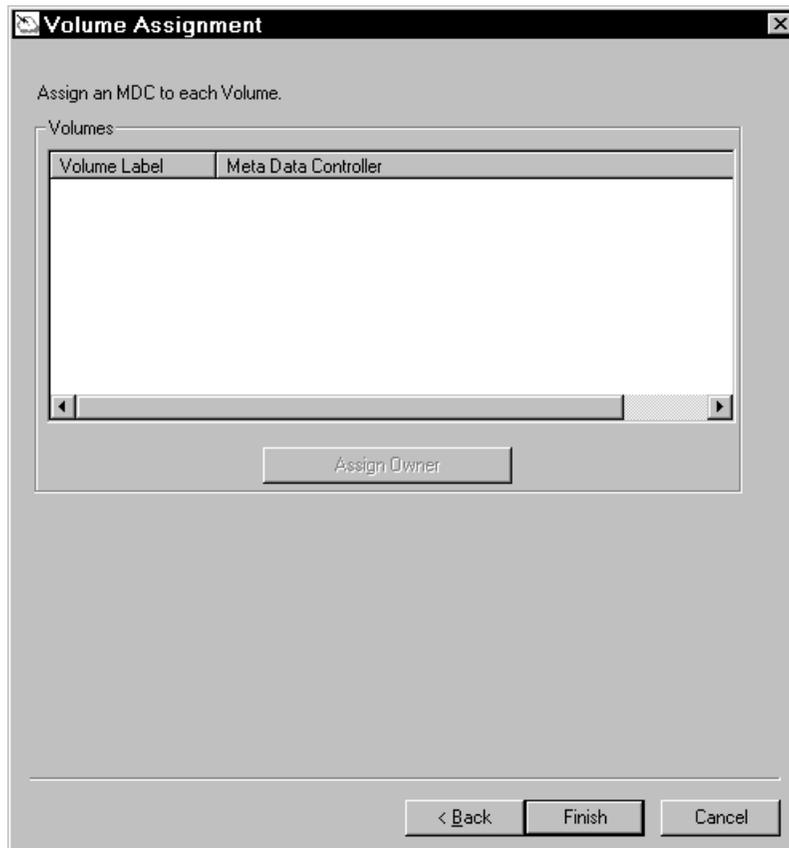


Figure 57. Volume Assignment on the Windows host

As Windows NT cannot read any volume label from the SANergy managed bus, this window is empty, and that is because the volume is owned by a UNIX MDC. When you have a Windows MDC running, you will see the volume and the MDC names listed.

The installation and configuration is completed by clicking on **Finish**. The system will reboot automatically. Check for the latest SANergy patch available on [www.tivoli.com/support/sanergy/maintenance.html](http://www.tivoli.com/support/sanergy/maintenance.html). The patch can be installed at this point.

There is an easy way to test if the host can directly read and write data to the shared volume over the SAN. Open the SANergy setup tool. Open **Start- > Programs -> SANergy- > SANergy Setup Tool** and select the tab **Performance Tester** (Figure 58).

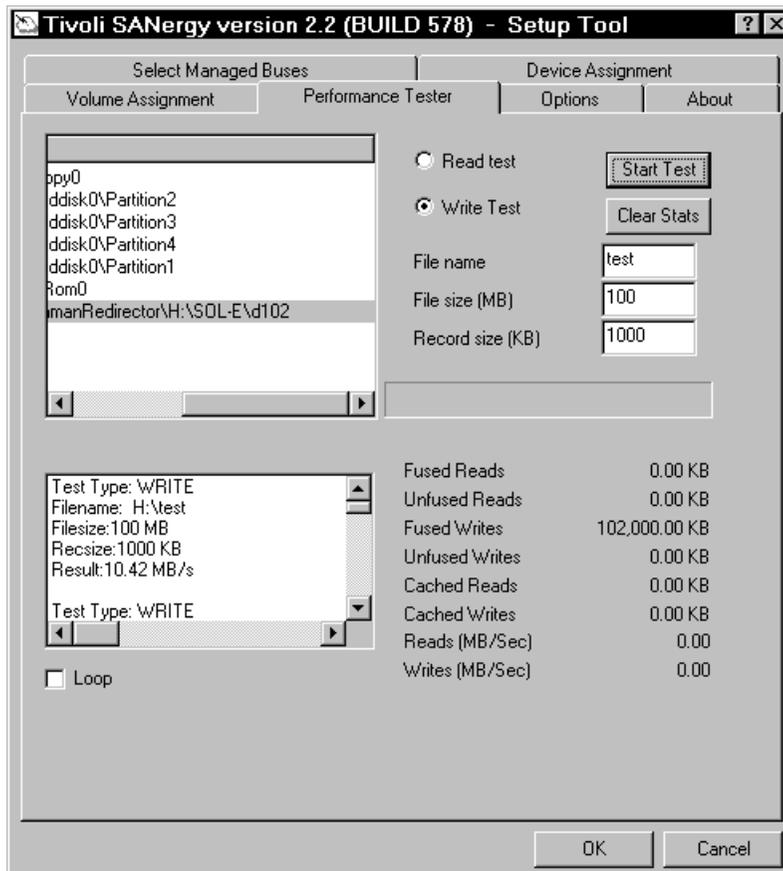


Figure 58. Verify installation with the performance tester on the Windows host

If the shared volume on the MDC is properly mounted on the host, you will see the mapped partition on the upper left window. Select the partition, the SAN attached device, against which to perform a write test. Type in a dummy file name, choose a file size in the appropriate fields, and press the **Start Test** button. On the right side, at the bottom, you can see the measured values for fused data throughput for this test.







---

## Chapter 4. Performance

In this chapter we discuss the possibilities for customizing and tuning your SANergy implementation, to tailor it to the environment and the needs of applications, so that the data throughput over the SAN is optimized.

We consider the standard SANergy parameters and options. Also, a new SANergy feature called ZOOM is now available. Details are provided in Appendix C, "Using ZOOM to improve small file I/O performance" on page 283.

---

### 4.1 General performance considerations

Before we present SANergy-specific tuning parameters, we will first focus on more general tools and applications available to the operating system and file sharing applications. These more generic functions help tune SANergy (as well as other applications) in a complex and heterogeneous environment and may help determine the performance bottleneck of an installation.

#### 4.1.1 UNIX NFS, LAN tuning, and performance problem diagnosis

Normal NFS and local area network (LAN) tuning considerations apply to both SANergy UNIX MDCs and hosts. This section covers a few of the commands and options that should be considered for NFS and network tuning in a SANergy environment. It also discusses some tools useful in diagnosing performance problems. The NFS documentation for your specific UNIX platforms should be consulted for additional information and detailed command syntax.

It should be kept in mind that normally SANergy is only using NFS and the LAN for sending and receiving metadata, and it is therefore less dependent on them than non-SANergy NFS based applications. In other words, if you are experiencing performance problems with SANergy and the disk volumes in question have actually been fused, NFS and the LAN are most likely not the source of the performance problem. Please also keep in mind that SANergy provides a performance benefit only for files of a minimum size and cannot handle the I/O in some configurations. See 1.5.2, "SANergy limitations" on page 14 for a more detailed discussion of this.

#### 4.1.1.1 NFS server daemon

The NFS `nfsd` daemon services client requests for file system operations on the NFS server. Therefore, it may have an impact on the performance of SANergy in certain environments.

The `nfsd` command starts the daemon. On Solaris the `nserver` parameter of this command controls the maximum number of concurrent requests the NFS server can handle, and the default value is 16. On Linux, the parameter is `nproc`, and the default is 1. Different UNIX implementations provide a variety of different control parameters for `nfsd`. You should consult the documentation for your MDC platform for other performance related settings.

#### 4.1.1.2 NFS mount command

This command is relevant only for UNIX SANergy UNIX hosts. The `mount` command recommended in the SANergy Administrator's Guide is:

```
mount -o acregmin=0,acregmax=0,actime=0 host:/share /mnt
```

The `mount` option `actime` specifies the length of time, in seconds, that cached attributes of files and directories are held before they are checked for changes. In this case, the attributes are checked after zero seconds, purged if changed, and new attributes retrieved from the back file system. The options `acregmin` and `acregmax` are actually redundant here, in that the option `actime=0` sets the value of both of them as well as those of the `acdirmin` and `acdirmax` parameters to 0.

#### 4.1.1.3 nfsstat command

The `nfsstat` command is a useful tool for diagnosing NFS related problems on both the NFS server and the clients, that is the SANergy MDC and hosts.

##### ***NFS server***

NFS statistics on the server can be displayed with the `nfsstat` command. `nfsstat -s` will show statistics such as calls received from clients and calls rejected, that is, bad calls. The latter information may help diagnose a problem where a client cannot communicate with the server. Calls can be rejected for a number of reasons. Here is an example of `nfsstat -s` on Solaris:

```

# nfsstat -s

Server rpc:
Connection oriented:
calls      badcalls  nullrecv  badlen    xdrCALL   dupchecks
4744      0         0         0         0         3405
dupreqs
0
Connectionless:
calls      badcalls  nullrecv  badlen    xdrCALL   dupchecks
0         0         0         0         0         0
dupreqs
0

Server nfs:
calls      badcalls
4744      0

```

### ***NFS client***

NFS performance problems on the client can result from calls that are rejected or have timed-out. These and other statistics can be displayed with the `nfsstat -c` command.

The next example we present shows a SANergy MDC that is not responding to a fuse request from our SANergy AIX host brazil. We first verify that the host knows of the disk to which we want to fuse, d12 on diomedes, by issuing the `owner` command. We then issue the `fuse` command. Z is the mount point of d12.

```

Enter Command: owner
+++++
Processing command - owner...
 0          rhdisk1 NO INFO
 1          rhdisk2 NO INFO
 2          rhdisk3 NO INFO
 3          rhdisk4 NO INFO
 4          rhdisk5 NO INFO
 5          rhdisk6 NO INFO
 6      rd1  rhdisk10      NO INFO rainier
 7      d11  rhdisk11      NO INFO ALDAN
 8      d12  rhdisk12      NO INFO DIOMEDE
 9      d102 rhdisk13      NO INFO sol-e
=====

Enter Command: fuse|/Z
+++++
Processing command - fuse|/Z...
NFS server diomedes not responding still trying

```

We then use the `nfsstat -c` command to help isolate the cause of the problem. The following statistics were observed shortly after the SANergy message above was displayed.

```
# nfsstat -c

Client rpc:
Connection oriented
calls      badcalls  badxids   timeouts  newcreds  badverfs  timers
86706     7         0         0         0         0         0
nomem     cantconn  interrupts
0         7         0
Connectionless
calls      badcalls  retrans   badxids   timeouts  newcreds  badverfs
43        2         5         0         0         0         0
timers    nomem    cantsend
0         0         0
```

We displayed these statistics again after waiting a while longer for the fuse to succeed.

```
# nfsstat -c

Client rpc:
Connection oriented
calls      badcalls  badxids   timeouts  newcreds  badverfs  timers
87001     302        0         0         0         0         0
nomem     cantconn  interrupts
0         302        0
Connectionless
calls      badcalls  retrans   badxids   timeouts  newcreds  badverfs
43        2         5         0         0         0         0
timers    nomem    cantsend
0         0         0
```

By comparing the statistics, we can see that the values of **badcalls** and of **cantconn** are increasing. The value of **cantconn** is the number of times the call failed due to a failure to make a connection to the server. This turned out to be the ultimate performance problem: the NFS server and SANergy MDC, diomede, was down.

#### 4.1.1.4 netstat command

The `netstat` command can be used to show statistics on the system's interface to the network and active sockets for a protocol. When an NFS server is operating under a heavy load, the server will sometimes overrun the interface driver output queue. This can be checked with the `netstat` command. Using the option `netstat -i` will show input or output errors, including statistics on dropped packets, listed in the columns labeled **Ierrs** and **Oerrs** below. This output comes from our SANergy MDC sol-e.

```
# netstat -i
Name Mtu Net/Dest Address Ipkts Ierrs Opkts Oerrs Collis Queue
lo0 8232 loopback localhost 1103921 0 1103921 0 0 0
hme0 1500 sol-e sol-e 299707 1 76993 0 4155 0
```

### 4.1.2 Operating system tools and considerations

There are various UNIX operating system tools to aid in assessing performance problems. Two of these are covered here, `iostat` and `vmstat`. We explain a consideration specific to the AIX environment that affects the behavior of SANergy. At the end of this section, we discuss Windows MDC resource requirements.

#### 4.1.2.1 I/O load and performance

The `iostat` command can be used with SANergy in two ways:

- It provides another way to verify that I/O operations are occurring directly to the fused disks instead of over NFS.
- It can be used to show the I/O rates and performance of the fused disks on the UNIX system.

An example of the output from an AIX `iostat` is shown below. In this example, a file was being copied to /d102, which is hdisk13, while we were running `iostat` with a two second interval.

The `iostat` command is a standard part of Solaris, AIX, and Red Hat Linux 7.0. It is available as an add-on package in earlier releases of Red Hat Linux; the package is called `sysstat` and is accessible through the Web site [www.rpmfind.net](http://www.rpmfind.net).

```
# iostat -d hdisk13 2
```

| Disks:  | % tm_act | Kbps   | tps   | Kb_read | Kb_wrtn |
|---------|----------|--------|-------|---------|---------|
| hdisk13 | 0.1      | 11.3   | 2.8   | 20014   | 264548  |
| hdisk13 | 0.0      | 0.0    | 0.0   | 0       | 0       |
| hdisk13 | 20.4     | 2260.3 | 565.1 | 0       | 4532    |
| hdisk13 | 38.0     | 3878.0 | 969.5 | 0       | 7756    |
| hdisk13 | 29.0     | 3072.0 | 768.0 | 0       | 6144    |
| hdisk13 | 31.0     | 3058.0 | 764.5 | 0       | 6116    |
| hdisk13 | 0.0      | 0.0    | 0.0   | 0       | 0       |

#### 4.1.2.2 System load

General information about the load on a UNIX system can be displayed using the `vmstat` command. A heavily loaded system may cause NFS or SANergy performance problems. This command is a standard part of Solaris, AIX, and Red Hat Linux.

#### 4.1.2.3 AIX Virtual Memory Manager

The AIX Virtual Memory Manager (VMM) may affect the observed behavior of read and write activity to SANergy managed disks. One example is sequential read-ahead. If you are performing an operation which results in a pattern of sequential reads to a file, VMM will cause data to be read into memory in advance of requests by the application. One outward manifestation of this behavior is that `iostat` on a SANergy host may show high activity to a fused volume for a while, then no activity, even though the application is still executing and reading data.

#### 4.1.2.4 Windows MDC resource requirements

The MDC is essentially a server that needs to process file requests very quickly; it does not need to handle much bandwidth. Faster processor speeds and Random Access Memory (RAM) are more important than the bus architecture. Providing the MDC with enough RAM to hold the entire Windows NTFS Master File Table (MFT) will help to process file requests more quickly. How large is the MFT? This depends on how many total files there are on the disk volumes and how fragmented the files are.

#### **Windows NT**

To determine the current size of the MFT of an NTFS volume on Windows NT, type the following `dir` command on the NTFS volume (where `x` is the SANergy volume or drive whose MFT size is to be determined):

```
X:\>dir /a $mft
```

Remember that the value reported by the DIR command may not be current.

## Windows 2000

To determine the current size of the MFT on Windows 2000, use **Disk Defragmenter** to analyze the NTFS drive, and then **View Report**. This displays the drive statistics, including the current MFT size and number of fragments. The exact path is as follows:

**Start -> Programs -> Administrative Tools -> Computer Management**

Then select **Disk Defragmenter**, choose the Volume, and click the **Analyze** button. When the analyze operation is complete, click the **View Report** button. Figure 59 shows the results, scrolled down to display the information on the Master File Table.

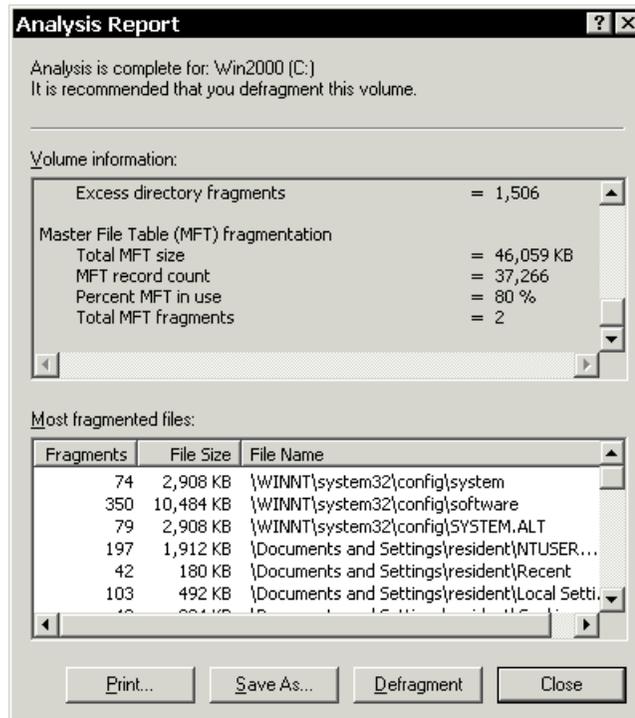


Figure 59. Windows 2000 disk defragmenter analysis report

### Additional information on the MFT

More information on the MFT can be found in the Microsoft support article entitled: *How NTFS Reserves Space for its Master File Table (MFT)*.

This article is available through the Microsoft support Web site:

[www.microsoft.com/support/](http://www.microsoft.com/support/)

## 4.2 SANergy performance parameters

In this section we describe the parameters of the SANergy software that allow for performance tuning. On a Windows platform, all the settings can be done on the SANergy Setup Tool from the **Options** tab.

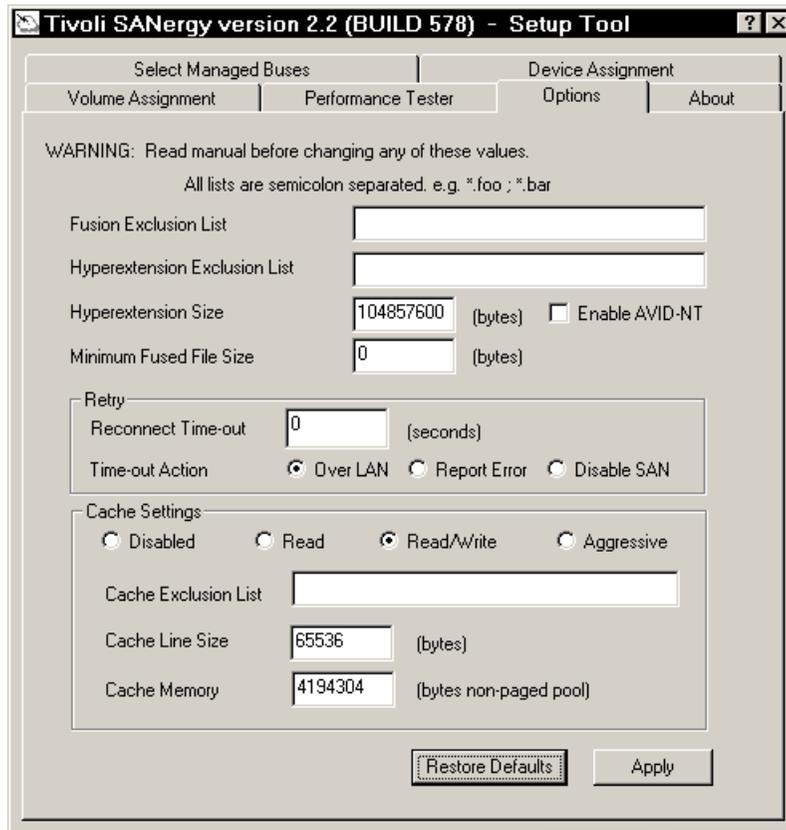


Figure 60. Options tab on the SANergy Setup Tool on Windows platforms

Click **Start- > Programs- > SANergy -> SANergy Setup Tool** and select the **Options** tab to get to the screen in (Figure 60).

On the UNIX platforms, all of these parameters can be set using the `SANergyconfig` command in the SANergy installation directory. Note that you should have the SANergy environment properly set up prior to issuing the command. Run the `SANergyshsetup` script in the active session to accomplish that. Additionally, some of the parameters can be set using the configuration GUI in Figure 61.



Figure 61. Configuration tool on UNIX systems

Every time you change a parameter you can activate the new setting by clicking the **Apply** button.

#### 4.2.1 Cache settings

Some applications desire direct-to-disk raw performance. But the majority of applications need “buffered” access. “Buffering” means that the data transfer of read and write operations passes through the system memory where it is stored for a short time. Applications can access data from the system memory much faster than from disk, and disk allocation time can be saved by this process. It allows the application developers to issue smaller I/Os and still get the greatest possible performance out of a storage system.

A small set of applications specify “no buffering”. Those programs typically do very large I/Os and are extremely performance sensitive (such as high speed video or imaging). In these cases, an extra buffer copy of the data would be an unnecessary overhead. Typically, all other applications can benefit from “buffering”. SANergy is able to perform larger I/Os than requested by an application and keep the unused data in memory in case it is required again in the future. SANergy refers to this activity as *caching*.

The SANergy cache is a per file cache. That is, the data in the cache is associated with a particular file at all times, and not just with the disk blocks. In this terminology, we have the concept of a *cache line*. A cache line is an area of memory associated with a certain region of a file. The size of the cache lines can be tuned in the GUI on all operating systems. Each file can allocate up to three cache lines (in all releases of SANergy Version 2.2 and later). As a file is read, cache lines are added. Once the maximum number of cache lines is reached, the least recently used line is deleted and a new one is opened.

The number of cache lines cannot be tuned with the standard interfaces (GUI and CLI). Nevertheless, for special applications on Windows NT and Windows 2000, that take extremely high advantage of data stored in the cache, you can change the limit of cache lines up to 20 by editing the registry value. The corresponding registry key is the parameter **CacheNumLines** in the [HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\mcfsr\Parameters] directory. Those applications perform frequent random reads in one area of a single file.

In addition to the cache line size, there is another parameter to regulate the total amount of memory consumed by all cache lines, the **cache memory**. This value can be tuned from the GUI by entering the value in the field **Cache Memory (bytes non-paged pool)** on Windows and **Total process (MB)** on UNIX. Once the total of all files open across a process have consumed this amount of memory, no new cache lines will be added for new files, and the I/O will start to go directly to disk.

For performance tuning with cache settings, it is necessary to know how the applications in your environment handle I/O requests. Performance tests always depend on the configuration of the whole environment. A certain set of parameter values can perform well in one scenario but perform much worse in a different one.

The first step is to decide how much memory you want to assign to SANergy for performing cache operations. Let us take 30 MB as an example. This is configured through the **cache memory** value. This value corresponds to the volume of the box in Figure 62.

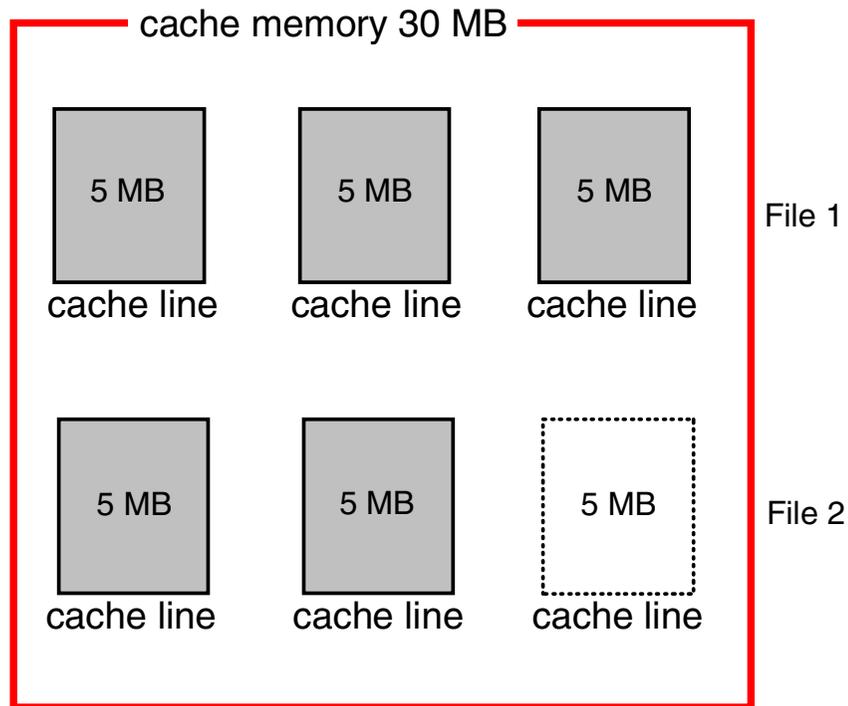


Figure 62. Cache settings with high cache line size

The next step is to set the **cache line size** parameter. To do this you would adjust the field **Cache Line Size (bytes)** on Windows and **Line size (KB)** on UNIX. This corresponds to the size of the rectangles inside the cache memory box. At a given cache memory size, a high cache line size should give high I/O performance when a few large files are accessed simultaneously. Also, larger I/O requests can be satisfied by the SANergy cache when a high cache line size is set.

In our example in Figure 62, we chose a cache line size of 5 MB per cache line. That means we can have a maximum of 6 cache lines for all cached files. If the first two files processed ask for the maximum of their allowed 3 cache lines, no further files could be cached at the same time. On the other hand, single I/O requests of sizes up to 5 MB can be satisfied within the cache with these settings.

**Note**

If an I/O request of an application is larger than the cache line size, the data will not be cached. To ensure that a certain application benefits from SANergy cache algorithm, set the cache line size higher than the typical I/O requests of this application.

A good setup for high performance of smaller files, but many at a time, would be the configuration shown in Figure 63.

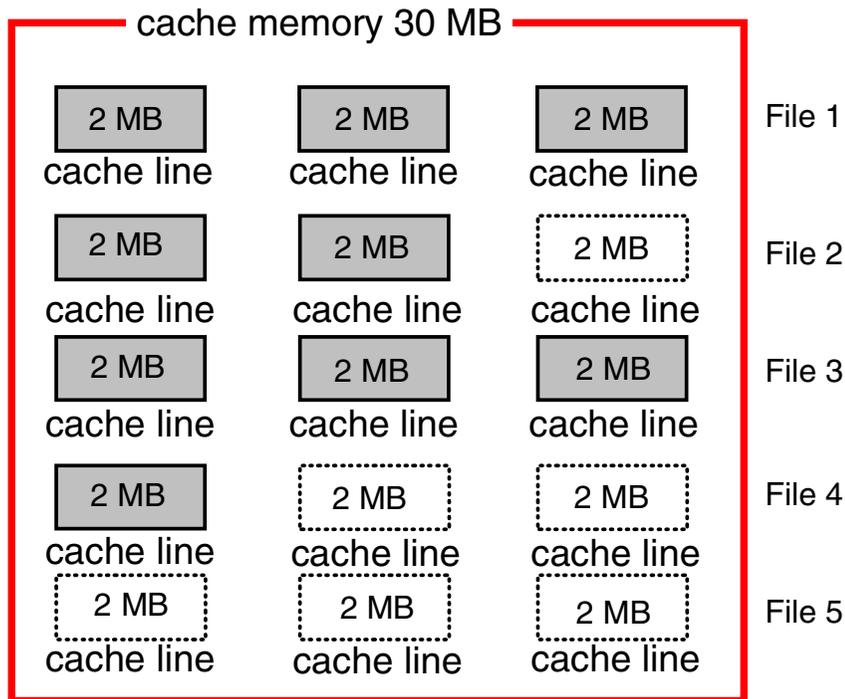


Figure 63. Cache settings with smaller cache line size

In this example we set the cache line size to 2 MB, while still using the same cache memory size of 30 MB. That makes a maximum of 15 cache lines in the reserved memory. Now there can be I/Os processed on parts of up to 5 files simultaneously if all files reserve the maximum of 3 lines, but the cached I/O size per request is lowered to 2 MB.

There is a difference between the cache processing on UNIX and Windows NT or Windows 2000. On Windows platforms, the cache is machine-wide or system-wide, while on UNIX it is per-open-file-instance. This means that on Windows, all processes benefit from the cache lines already present, if multiple applications request the same area of a file. However, it also means that the maximum number of cache lines for a file (the default is three) can be reached sooner, as many processes will share all the cache lines available for a file.

Particular files can be excluded from caching by specifying the file type in the **Cache Exclusion List** (Windows) or **Exclude** (UNIX) field on the GUI.

On both UNIX and Windows platforms, you can specify the cache mode. You can choose between **None**, which disables caching; **Read**, to cache only files opened with read-only access; and **ReadWrite**, to cache all files except for those opened with shared-write access. The ReadWrite option is the default. On Windows systems, there is another mode called **Aggressive** caching. When this mode is set, SANergy will buffer all files even if they are with shared-write access. UNIX systems do not have the option of opening a file with the shared flags, therefore the Aggressive caching option is neither provided nor relevant.

#### 4.2.2 Hyperextension

Before files are written to the shared file system, a certain amount of space has to be allocated for this file by the MDC (as the owner of the file system) to enable the SANergy host to write data into the file. In order to minimize the metadata traffic sent between the host and the MDC, the space requested to be allocated for a file at a time should be in large chunks rather than small ones.

The **Hyperextension Size** tells SANergy the size to be allocated for a file in a single request from a SANergy host. Whenever the SANergy host has written this amount of data into a file and writing is not yet finished, a further request to extend the file is issued to the MDC. In general, a large value for hyperextension size is beneficial, however large hyperextension values require more free space on the file system, especially when many files are opened concurrently for write access.

#### Note

On earlier releases of SANergy with Sun Solaris using the UFS file system, the hyperextension process had a considerable impact on performance, since UFS not only allocates the space, but fills the allocated space with dummy data on each request. That means the amount of data transferred to the disk while writing a file is doubled, once from the MDC (the dummy data) and then from the SANergy host to the disk (for the actual file data).

In order to minimize the slowdown of I/O when using a Solaris MDC with a UFS file system, we recommend setting the hyperextension value to a reasonably smaller value to avoid allocating space that is not needed by the host. However, for SANergy at V2.2.1.5 and later, this issue is now rectified and setting the hyperextension value to a low value is no longer necessary.

To prevent SANergy from hyperextending special files that do not take advantage of hyperextension, you can specify the type of this file in the **Hyperextension Exclusion List** field. Normally you would not exclude files from hyperextension, because this feature accelerates throughput.

However, there might be situations where you mainly deal with large files but have some very small files to transfer as well. In this environment, a high hyperextension size results in good performance for the large files, but when transferring the small files, time is wasted for space allocation. The allocated space is not needed for the small files and is released afterwards, so performance suffers. In this scenario it might be better to exclude the type of the small files from hyperextension, since Windows systems are very fast in general when hyperextending.

#### 4.2.3 Minimum fused file size

By default, files of all sizes on a SANergy managed volume are fused on the host. You can set a low threshold to prevent small files from using fused I/O. You can specify a file size in bytes in the field **Minimum Fused File Size**. Files smaller than this size will be accessed using conventional LAN-based I/O from the MDC and the host. The reason to do this is to avoid the large metadata overhead of fusing small files — that is, those files where the amount of actual file data being transferred is comparable to the size of the metadata. Fusing the access to these files with SANergy could result in minimal or no performance benefit, compared with using conventional I/O. A typical recommendation here is to exclude all files smaller than 10 KB from being fused.

#### 4.2.4 Fusion exclusion list

Apart the file size, you also can specify which type of files are going to be fused. Enter in the field **Fusion Exclusion List** the list of wild-card file names separated by semicolons that you do not want to be fused (for example, \*.html to exclude Web files). You can easily use this feature to measure the difference between data transfer over LAN and over SAN by temporarily excluding certain files from fusion and comparing the throughputs before and after.

#### 4.2.5 Logging

Enabling logging for SANergy is useful for monitoring and problem diagnosis. However, the overhead of logging has a direct impact on performance. SANergy allows you to specify the level of logging provided to enable a balance between collecting useful information and achieving reasonable performance. A value of 0 disables logging and is the value recommended for normal operation. Higher levels are used for debugging problems. In our limited testing, setting logging to level 5 degraded performance more than 45% compared to level 0. Setting the logging level is done using the `SANergyconfig log` command on a UNIX MDC.

##### 4.2.5.1 Setting logging parameters on UNIX

The `SANergyconfig log` command has four parameters:

- **Logging level:** This parameter can be a value in the range of 0 to 6, where 0 is off and 6 provides the most detail. The default value is 0.
- **Log destination:** This parameter can be the keyword “screen” for console output or a fully qualified file specification, such as `/usr/SANergy/mclog`.
- **Number of logs:** This parameter controls the maximum number of files that logging can generate.
- **Size:** This parameter controls the log file size.

If you issue the `SANergyconfig log` command without any parameters, it will return the current log settings.

##### 4.2.5.2 SANergy logging on Windows

Control of SANergy logging on the Windows platform is currently limited. The level of detail of logging and the other logging parameters cannot currently be set. SANergy log messages are generated in the Windows System event log, and they can be viewed with the Event Viewer.

#### 4.2.6 Datagram threads (dgram) parameter

The SANergyconfig `dgram` command sets the number of MDC datagram threads used to listen for client requests. The value is 5 by default. This parameter may affect performance in some situations. Given our limited workload configuration, we were not able to exceed the capability of even one datagram thread. We ran three concurrent host sessions writing data to the volume owned by our **sol-e** MDC. Varying the `dgram` setting between 1 and 5 did not appear to affect performance at all. If you do change this value, you need to stop and restart SANergy for it to be effective.

#### 4.2.7 Error handling

On Windows hosts by default, whenever a problem occurs that prevents SANergy from fusing data over the SAN (such as a connectivity loss), the data request will be re-directed over the LAN. This fallback is normally effective in most environments. You can also choose to disable the SAN for further transactions when SANergy runs into a time-out by selecting **Disable SAN** in the **Retry** panel (Figure 64), which is part of the main window (Figure 60 on page 96).



Figure 64. Retry panel on the option tab

Transactions will continue to be processed over the LAN in this case until you rectify the SAN problem and reboot the system. The time to wait until invoking the Time-out action defined can be set in the field **Reconnect Time-out**.

If you select the Report Error option, then if the SAN times out, the transfer will stop and SANergy will report an error to the application. In this case, the fallback path (the LAN) will not be used. You would select this option if you do not want SANergy traffic ever to be diverted back to the LAN — for example, if you do not want to increase the LAN traffic bandwidth.

---

## Chapter 5. Advanced SANergy configuration

Until now, we have concentrated on the basic SANergy setup. However, SANergy can also add value in more complex scenarios by providing heterogeneous file and volume sharing while exploiting the speed of a SAN. This chapter describes two common advanced SANergy implementations:

- SANergy MDC in a Microsoft Cluster Server (MSCS) configuration
- SANergy sharing striped volumes on a Windows MDC

---

### 5.1 MSCS clustering for the MDC

It is often desirable to configure a SANergy MDC for high availability. Where the data being shared or applications being served are highly critical to the enterprise, it is vital that access to them over the SAN is always available, regardless of single points of failure. Making an application or service highly available typically means duplicating the important resources so that the impact to the end users of any planned or unplanned failure in an individual system or subsystem failure is minimized. Workload on a failed system can be transferred or passed over (transparently or nearly transparently) to another system which is still available.

There are two main ways to create a SANergy high availability solution. You can use the special HA component available as a SANergy feature, or you can use a native high availability product for the MDC operating system (for example, Microsoft Cluster Server on Windows NT or Windows 2000).

We recommend using a native clustering solution, as it is more tightly integrated with the operating system. It may also provide additional benefits, for example, automatic failover of the share referencing the SANergy shared volume with MSCS.

This section documents the steps necessary to install SANergy on a cluster running Microsoft Cluster Server (MSCS). This cluster will be built using Windows 2000 Advanced Server; however, the installation steps should be identical on any MSCS compliant operating system.

### 5.1.1 Base configuration

Figure 65 shows our basic testing and validation configuration. We built a cluster named musala from nodes bonnie and clyde. The nodes were all running Windows 2000 Advanced Server. Our SANergy host in these tests was elbrus, running Windows 2000 Server. Each of these machines could detect three drives in our SAN. One of these drives was used as the quorum resource for the MSCS cluster (SAN\_Quorum). The other two were to be used as SANergy shared volumes (SANergy\_Disk1 and SANergy\_Disk2).

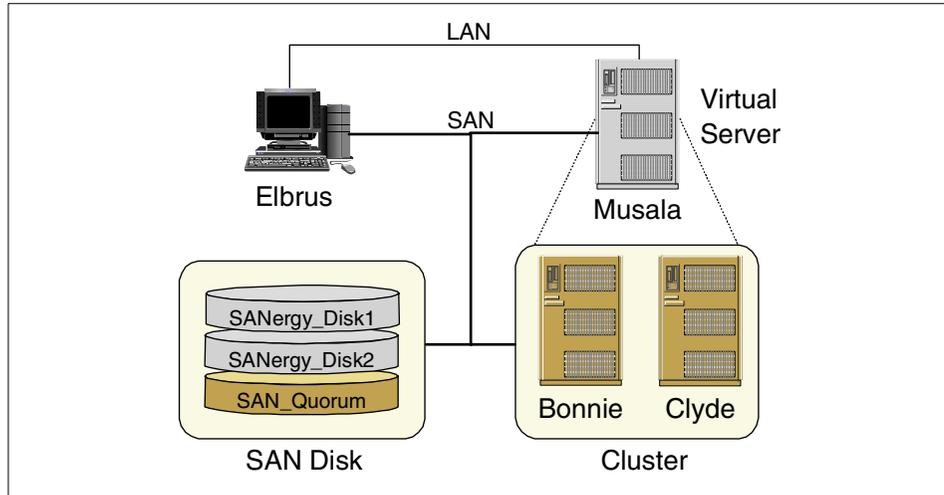


Figure 65. Testing cluster configuration

MSCS defines one of its shared drives as a *quorum resource*. This resource contains data vital for the operation of the cluster and is always reserved by the node hosting the cluster group. It is generally recommended that the quorum resource be an entire volume, instead of just one partition on a volume. This prevents potential problems that could occur if the quorum resource had to be moved between nodes, thus requiring the entire volume to be relocated.

The other shared disks are used by different applications hosted on the cluster. Normally, these are defined as physical disk resources and are controlled by MSCS. To integrate MSCS and SANergy, we have to take special steps to allow them to work together.

### 5.1.2 How to integrate SANergy and MSCS

MSCS works by allowing you to define resources of various types that require failover protection. These resources can be grouped together so that they operate as a single unit. An example of resource types are server names (which appears to the outside world as an ordinary server), disks, and applications. A group could include a server name, an application, and the disks that support the application. MSCS will guarantee the availability of this group of resources by monitoring them and making certain they are operating together on one of the physical machines in the cluster (a node).

In addition to making certain that the group of resources is available, MSCS ensures that they are not running on more than one machine at any given time. If a group were active on more than one machine, many sorts of problems and failures would occur.

MSCS prevents the mounting of a physical disk resource on more than a single machine at a time. It accomplishes this by use of the SCSI reserve/release commands. Due to the fact that MSCS attempts to limit access to physical disk resources to a single host, a special type of resource must be used for those disks which will be shared by using SANergy. SANergy then takes on the role of allowing sharing of the resources while making certain that data corruption does not occur.

The software component that make SANergy cluster-aware is the SANergy MSCS module. This module is included with the base SANergy package and enables the definition of SANergy volumes within MSCS.

### 5.1.3 Installing and configuring SANergy with MSCS

#### Note

Since some readers will already be familiar with the processes described in this section, a concise outline of those steps is presented here. The steps are then presented in greater detail in the following sections.

1. Uninstall MSCS, if it is already running. This is necessary, as a previous MSCS installation will have ownership of the disks you wish to be shared via SANergy, which will prevent SANergy from accessing them. An alternative to performing a complete uninstall is to delete the physical disk resources that correspond to the volumes you wish to share via SANergy. These will later be redefined as SANergy Volume resources.
2. Make certain that each node in the cluster has access to the disks to be shared. Each node will need to have the same drive letter for each disk.
3. Reinstall MSCS. Make certain the quorum drive is on its own disk, not on a partition. DO NOT allow MSCS to manage the disks, except for the quorum drive.
4. Install the SANergy file sharing component on each node in the cluster as documented in Chapter 2, "SANergy with a Windows MDC" on page 19. Use special names for assigning volumes to their MDCs. When defining the MSCS quorum disk, use the special name **?FREE** for the MDC. When defining the volumes to be shared via SANergy, use the special name **?CLUS** to denote they are not owned by any specific machine, but rather the cluster.
5. Install SANergy MSCS component on each node in the cluster. When installing on the final node, SANergy will register the new resource type to the cluster.
6. Define each volume to be shared to MSCS, using the **SANergy Volume** resource type.
7. Define a **File Share** resource for each disk. Make this resource dependent upon the corresponding SANergy Volume resource.
8. Map the CIFS share from the SANergy hosts sharing the volume.
9. Validate SANergy functionality by running performance tests from both the MDC (direct access to the disks) and the hosts sharing the volume (access via SANergy).

### 5.1.3.1 Uninstall MSCS

It is critical that SANergy and MSCS do not both attempt to manage the same disks. The best method to ensure this is to uninstall MSCS, if it is currently on the machine. Start the uninstall process by using the following options: Select **Start -> Control Panel -> Add/Remove Programs -> Add/Remove Windows Components**. You will be presented with the *Windows Component Wizard* (see Figure 66).

Uncheck the **Cluster Server** check box and click **Next**. The Windows Component Wizard will proceed and uninstall MSCS. Do not be alarmed if the wizard makes changes to the other Windows components checked. If you did not specify any other changes except for uninstalling MSCS, the wizard will not modify other settings.



Figure 66. Removing previous MSCS installation

### Note

Although it is cleaner to perform an uninstall of MSCS, it may not be desirable or possible in some situations. As an alternative, you can simply delete the physical disk resources from MSCS that you wish to be shared by SANergy. Later, you will re-define these resources as SANergy Volumes.

If you decide to use this method, be certain to note which groups the physical disk resources were members of, as well as any dependencies related to those resources.

#### 5.1.3.2 Make certain all nodes have access to SAN disks

All of the MSCS nodes that will be used as MDCs should obviously have access to the volumes to be shared. Configure your SAN components so that the machines have access and can mount the volumes. Once this is done, validate that the volumes are visible in the *Disk Management* applet in *Computer Management*. You can start Computer Management by selecting **Start -> Programs -> Administrative Tools -> Computer Management**.

You should see the disks to be shared (see Figure 67). You may or may not see the volume information, depending upon whether the disks are already being accessed by other hosts.

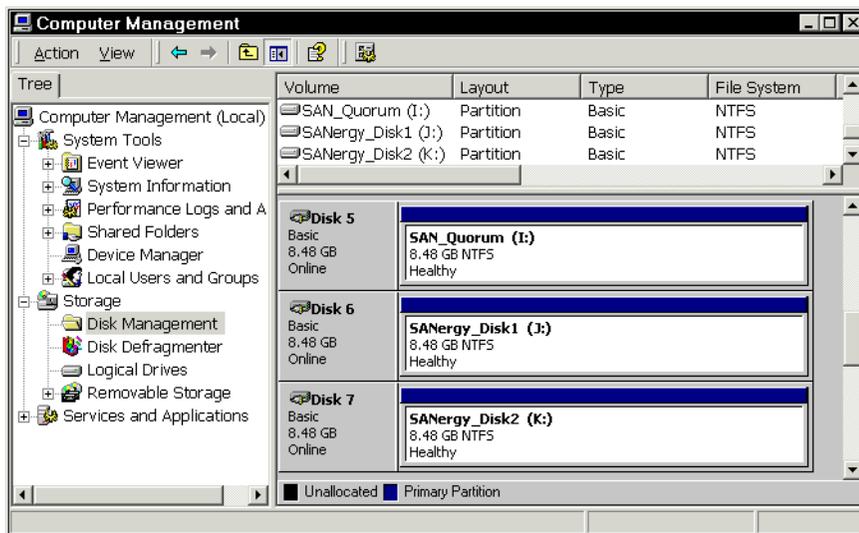


Figure 67. Validating that machine can access SAN disks

All nodes in the cluster should use the same drive letters to represent each disk. This is necessary to later build other cluster resources that access the disk, such as file share resources. To set the drive letter, right-click a volume and select **Change Drive Letter and Path** from the pop-up menu (see Figure 68).

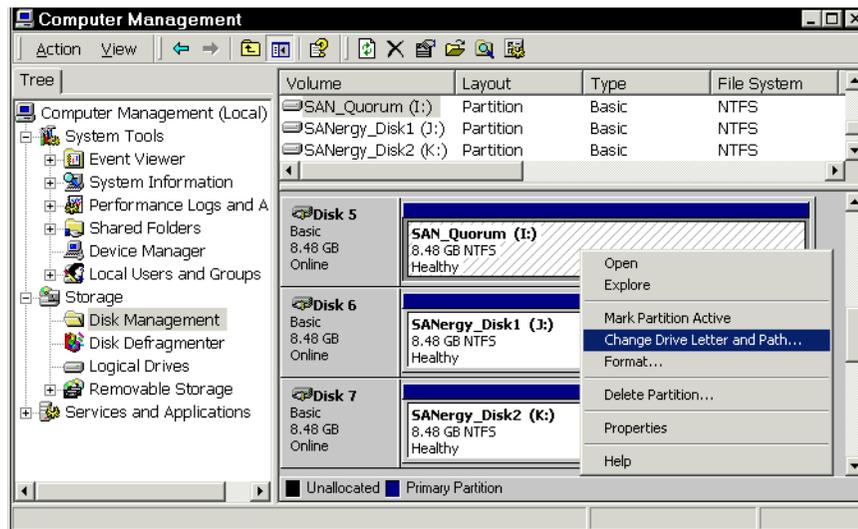


Figure 68. Assigning drive letter to volume

You will also need to note the label and volume serial number assigned to the volumes to be shared later. It is important that you gather this information now, as it will not be possible to do this later without uninstalling software. Open a command prompt (select **Start -> Programs -> Accessories -> Command Prompt**). Issue the `vol` command for each drive. Note both the label and volume serial number of the disk. If the `vol` command gives an error, you may need to run this command from whatever machine currently owns the volume.

```
C:\>vol j:
Volume in drive J is SANergy_Disk1
Volume Serial Number is 92D4-A4FB

C:\>vol k:
Volume in drive K is SANergy_Disk2
Volume Serial Number is 66D0-117B
```

### 5.1.3.3 Install MSCS

Select **Start button -> Control Panel -> Add/Remove Programs -> Add/Remove Windows Components**. You will be presented the *Windows Component Wizard* (see Figure 69).

Check the **Cluster Server** check box and click **Next**. The Windows Component Wizard will proceed with the installation of MSCS. Do not be alarmed if the wizard makes changes to the other Windows components checked. If you did not specify any other changes except for installing MSCS, the wizard will not modify other settings.



Figure 69. Install MSCS

It is important that you do not allow MSCS to manage the disks to be shared via SANergy. When presented with a list of disks to be managed, move those disks to be shared via SANergy to the *Unmanaged Disks* column (see Figure 70).

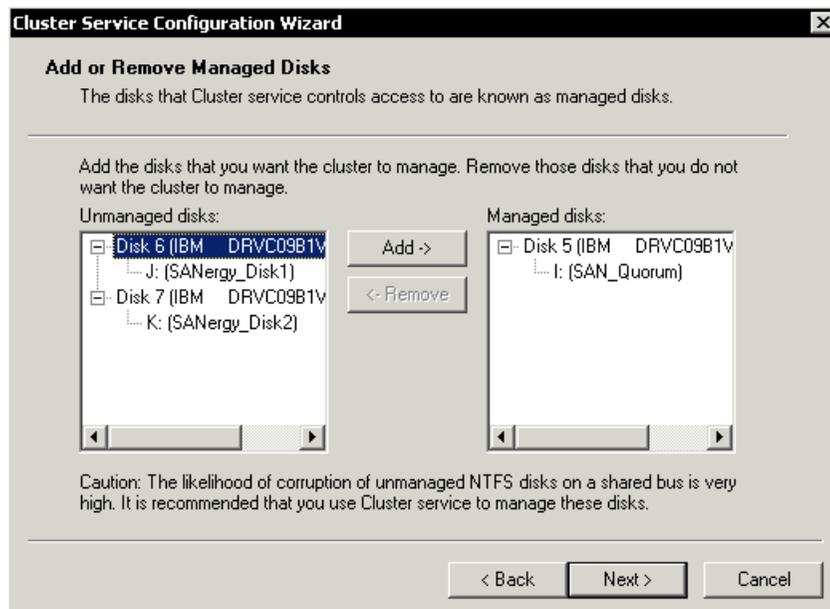


Figure 70. Identifying managed disks

Make certain the disk to be used as the quorum resource is managed. In our examples, we selected one of the disks on the SAN to be the quorum resource (see Figure 71).

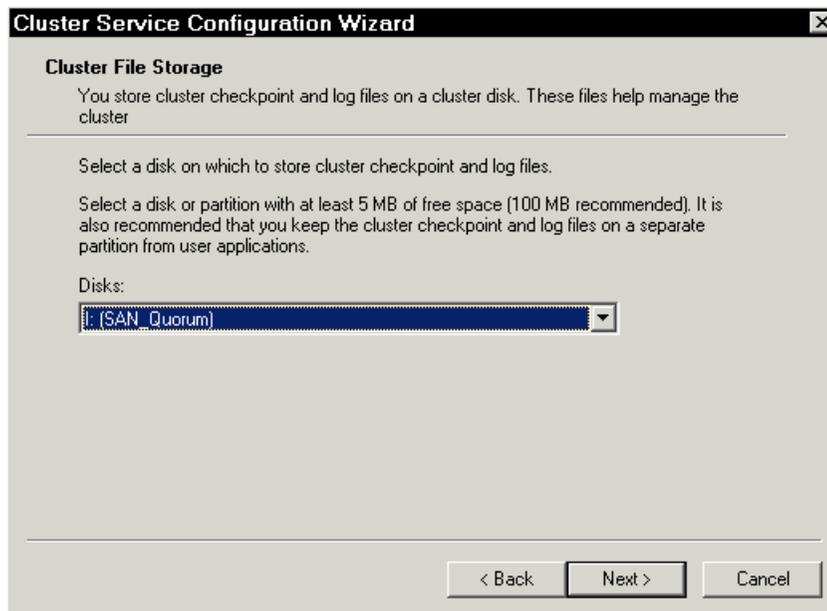


Figure 71. Define quorum disk

All other MSCS parameters and settings can use the same values as any other MSCS configuration in your environment.

#### 5.1.3.4 Install SANergy on cluster nodes

Install the SANergy base-code as normal. Select the **Tivoli SANergy** button from the main install screen (see Figure 72). Chapter 2, "SANergy with a Windows MDC" on page 19, contains detailed information on the installation and configuration of SANergy on Windows machines. The only difference, when installing the SANergy base-code on a clustered MDC, involves the names used to identify which MDC manages a volume.



Figure 72. Install Tivoli SANergy file sharing

When the installation of SANergy is complete, the SANergy Setup Tool will automatically be displayed. As usual, identify which SCSI buses are to be managed by SANergy — depending on your configuration, some of the volumes on the bus may not be shared using SANergy. Select **Next** to proceed. The next window allows you to specify the machines to own the individual disk devices. For those devices that will be shared from the MDC running on the MSCS cluster, the ownership name is not critical. Just accept whatever is currently assigned (see Figure 73). Select **Next** to proceed to the Volume Assignment window.

| Tag  | Bus Name | Id:Lun | Size (GB) | Type            | Owners       |
|------|----------|--------|-----------|-----------------|--------------|
| ssa0 | scsi3    | 0:0    | 8.480     | IBM DRVC09B1... | CLYDE        |
| ssa1 | scsi3    | 0:1    | 8.480     | IBM DRVC09B1... | CLYDE        |
| ssa2 | scsi3    | 0:2    | 8.480     | IBM DRVC09B1... | CLYDE        |
| ssa3 | scsi3    | 0:3    | 8.480     | IBM DRVC09B1... | CLYDE        |
| ssa4 | scsi3    | 0:4    | 8.480     | IBM DRVC09B1... | CLYDE        |
|      | scsi3    | 0:5    | 0.000     | IBM DRVC09B1... | Data unav... |
|      | scsi3    | 0:6    | 0.000     | IBM DRVC09B1... | Data unav... |
|      | scsi3    | 0:7    | 0.000     | IBM DRVC09B1... | Data unav... |

Figure 73. Device assignment — defaults are acceptable

The name assigned to the MDC for volumes to be shared is critical. For the quorum drive, use the special name **?FREE** to prevent SANergy from attempting to manage this disk. The MSCS cluster must manage the quorum resource. For those disks to be shared by SANergy, use the special name **?CLUS** to inform SANergy that these are cluster resources and are not owned by any specific machine (see Figure 74). SANergy will then automatically use whichever node owns the SANergy Volume resource as the MDC for that volume.

Using this process, you will install SANergy file sharing, as well as any current SANergy updates necessary, on all nodes in the cluster.

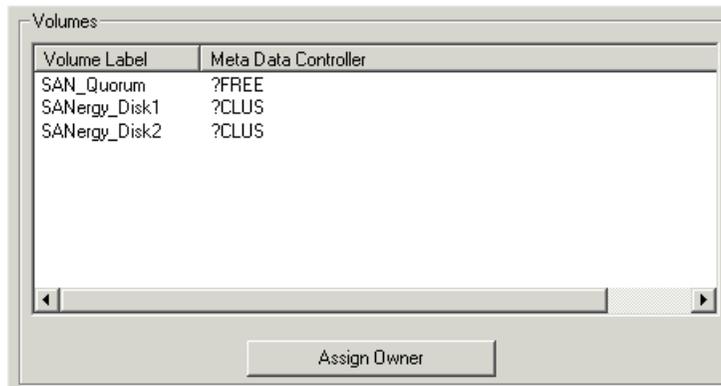


Figure 74. Special names on volume assignment

#### 5.1.3.5 Install SANergy MSCS

Install the SANergy MSCS component on each node in the cluster. This can be done at the same time as the installation of the SANergy base-code. From the SANergy main installation menu, select SANergy MSCS (see Figure 72 on page 115). From the SANergy MSCS installation menu, simply select **Install** (see Figure 75).

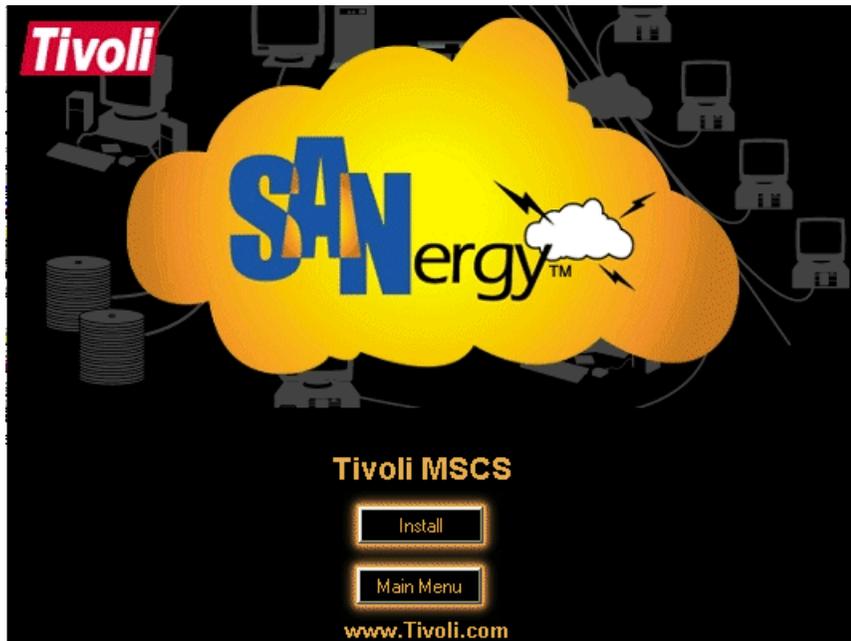


Figure 75. Tivoli SANergy MSCS install menu

You will then proceed to the normal installation of the SANergy MSCS component. For every node, except the last, select **Cluster Node Setup** as the installation type (see Figure 76). For each node, it is very important that you install the component in the same location.

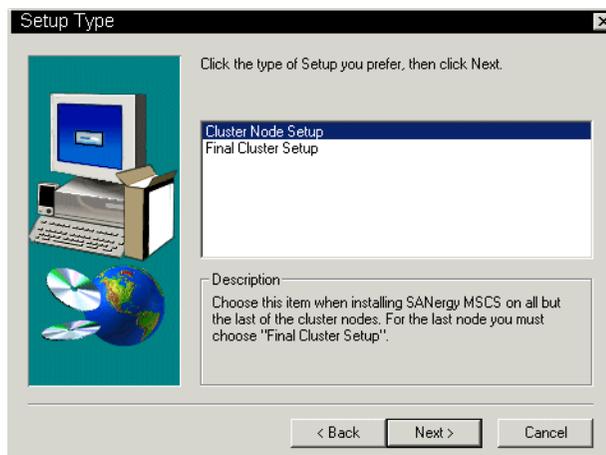


Figure 76. Installing SANergy MSCS on cluster node other than final node

When installing on the final node, select **Final Cluster Setup** as the install type (see Figure 77).



Figure 77. Installing SANergy MSCS on final node in a cluster

In the final step of this installation, a window will be displayed to register the SANergy Volume resource type to MSCS. Select **Install** from this window to proceed (see Figure 78). In our configuration, for example, we used a **Cluster Node Setup** on *bonnie* (the first machine we installed SANergy MSCS upon); and on *clyde* we specified a **Final Cluster Setup**.

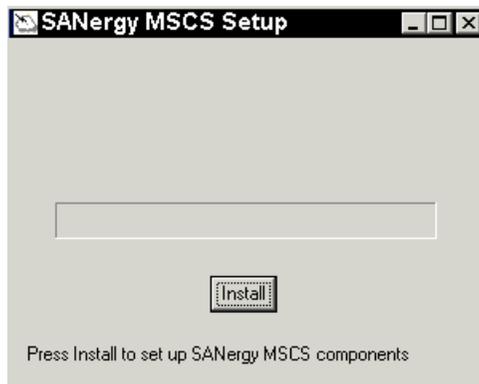


Figure 78. Last window of SANergy MSCS install on final cluster node

After the installation is complete, you can validate that the SANergy Volume resource is available in your cluster by listing the **Resource Types** from the Cluster Administrator (see Figure 79). Start Cluster Administrator by selecting **Start -> Programs -> Administrative Tools -> Cluster Administrator**.

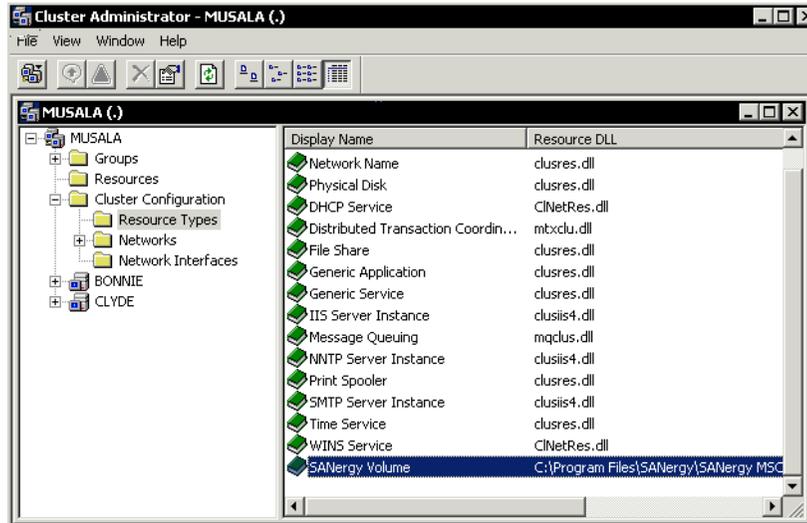


Figure 79. Validate that SANergy Volume resource type is available

### 5.1.3.6 Define SANergy Volume cluster resources

To allow shared SANergy volumes to be shared from an MSCS cluster, you must define them as a resource to a cluster group. Start Cluster Administrator and click the right mouse button on the right pane of the window. Select **New -> Resource** from the pop-up menu (see Figure 80).

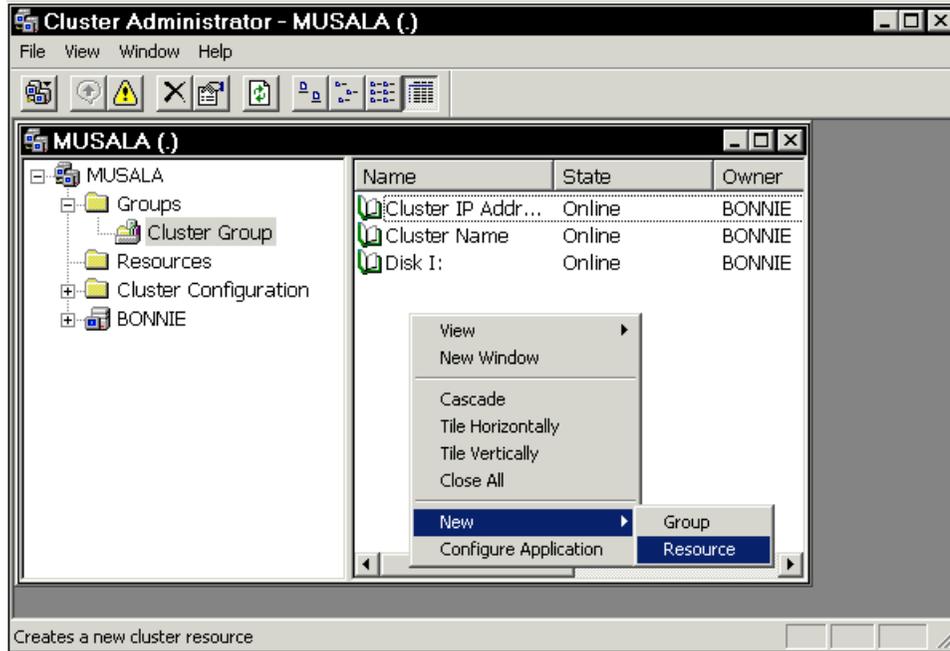


Figure 80. Adding a new cluster resource

A New Resource wizard will then start. The first window will allow you to enter the name of the resource (see Figure 81). This can be any valid resource name, since SANergy does not require a special naming convention. In our examples, for clarity, we gave the resource the same name as the volume's label. The resource type will be **SANergy Volume**. Also, you can select which group will own this resource. We selected the Cluster Group, but you can use any group you wish. Select **Next** to continue.

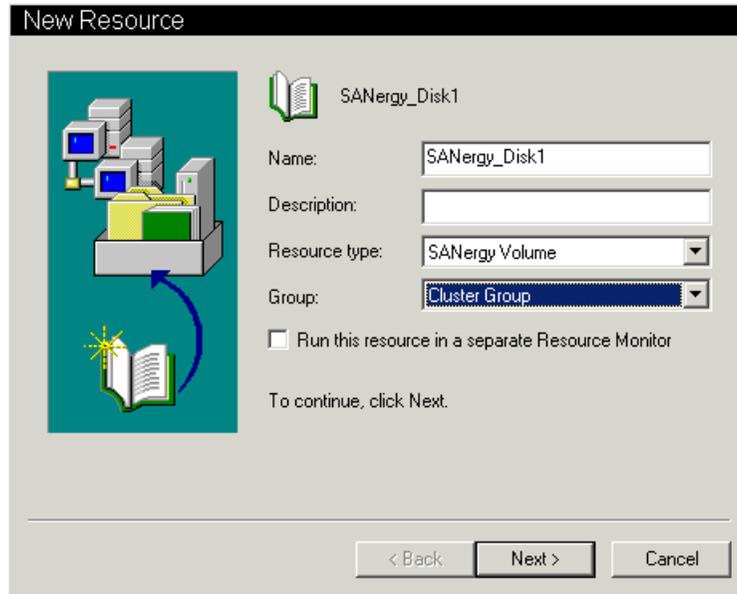


Figure 81. Defining a new resource

You will now see a window which allows you to specify which nodes can host the resource (see Figure 82). By default, all nodes are selected (bonnie and clyde in our configuration). Modify the list of nodes, if desired, and click **Next** when done.

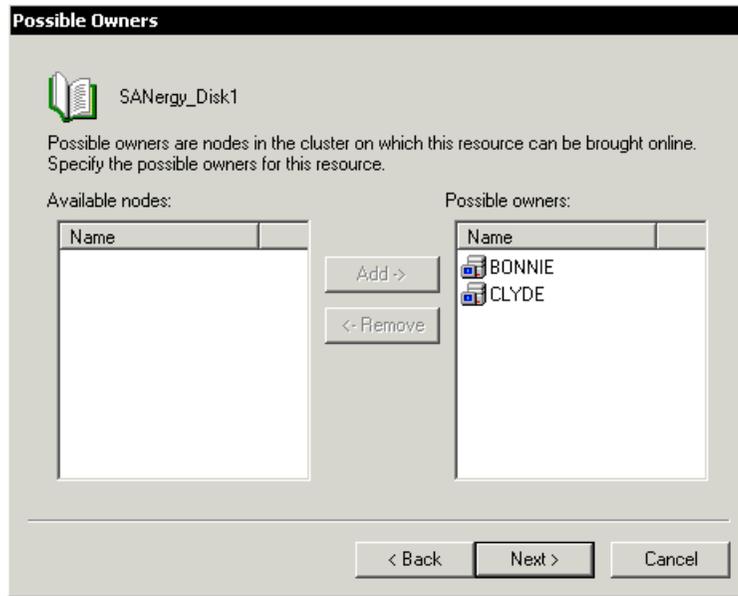


Figure 82. Identifying which nodes can support the resource

The next window will allow you to specify any dependencies that this resource may have on other resources (see Figure 83). Normally, there are no dependencies, so select **Next** to proceed.

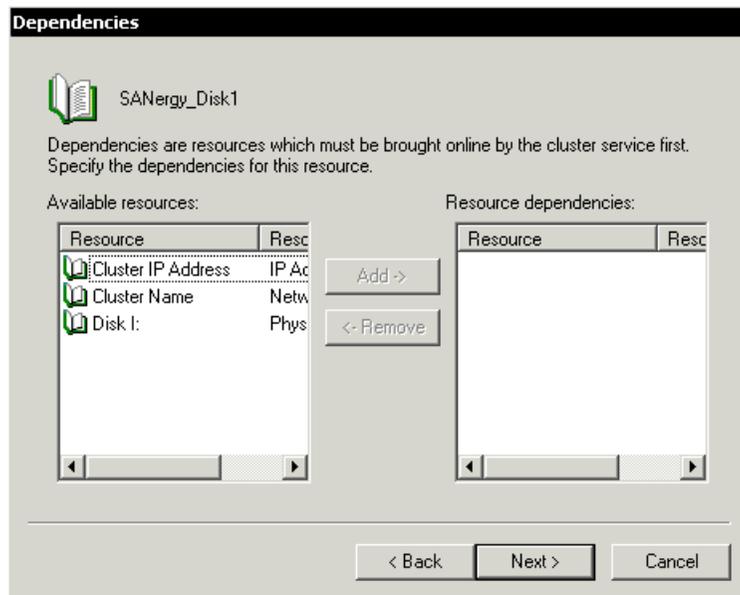
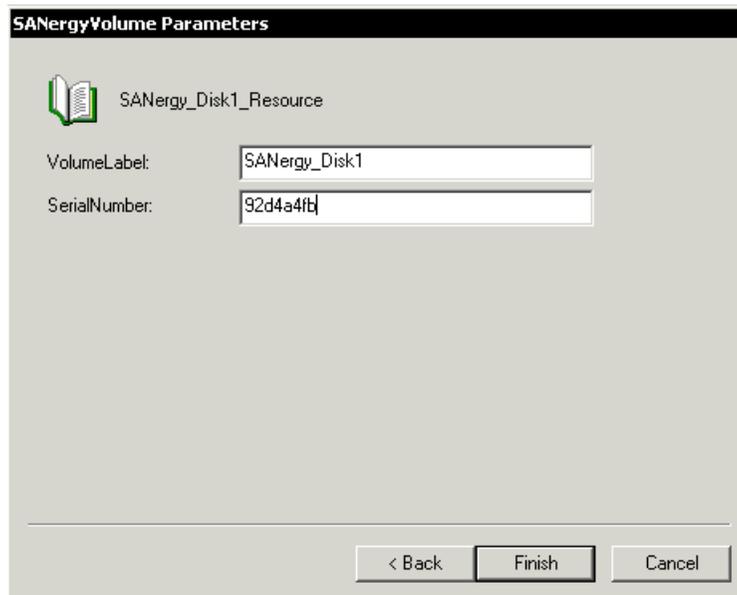


Figure 83. Identifying resource dependencies

Finally, the New Resource wizard will prompt you for information about the volume being shared (see Figure 84). You will need to enter the volume label and serial number which is information you gathered before installing SANergy (see 5.1.3.2, “Make certain all nodes have access to SAN disks” on page 110).



The image shows a dialog box titled "SANergy Volume Parameters". It contains a small icon of a book and the text "SANergy\_Disk1\_Resource". Below this, there are two input fields: "VolumeLabel:" with the value "SANergy\_Disk1" and "SerialNumber:" with the value "92d4a4fb". At the bottom of the dialog, there are three buttons: "< Back", "Finish", and "Cancel".

Figure 84. Setting SANergy Volume resource parameters

You can now select **Finish** to complete the task of creating the new SANergy Volume resource. Create a resource for each volume you wish to share. In our configuration, we created SANergy Volume resources for 2 disks, SANergy\_Disk1 and SANergy\_Disk2. As a reminder, you will never need to create a SANergy Volume resource for the quorum disk, as that disk is exclusively managed by MSCS.

After you have defined the resources, you can test them by bringing them online to the cluster. In Cluster Administrator, right-click the resource and select **Bring Online** from the menu (see Figure 85).

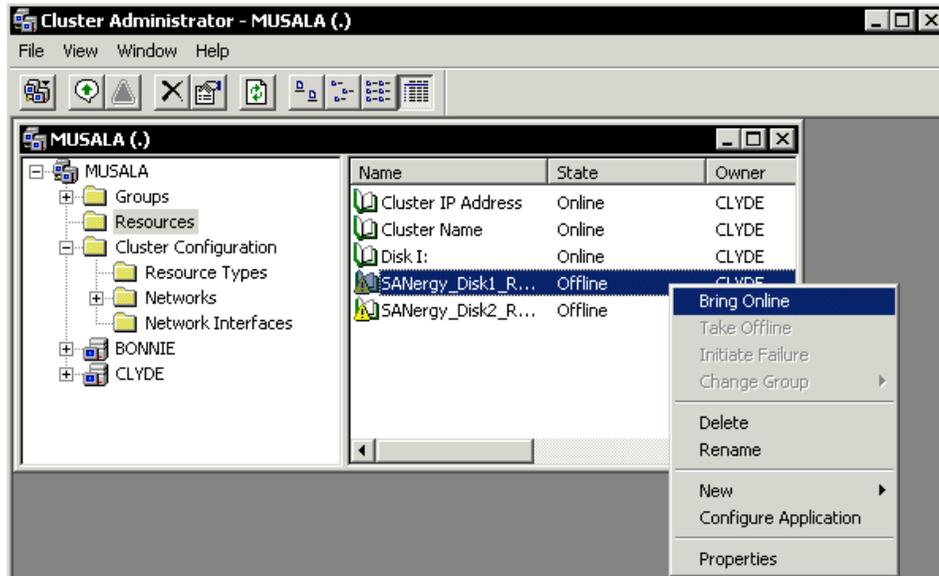


Figure 85. Bringing SANergy Volume resources online

If you have any problems, Cluster Administrator should be able to identify the cause. The computer's event logs may also provide diagnostic information. The most common causes of problems are incorrect values in the **Volume Label** and **Serial Number** parameters for the resource. You can view those by right-clicking on the resource and selecting **Properties** and then **Parameters**.

### 5.1.3.7 Define File Share cluster resource

Now that you have a SANergy Volume resource available, it is necessary to define a CIFS share to allow SANergy hosts to access the volumes. When defining a CIFS share on a MSCS cluster, you do not use the normal techniques. The normal methods define shares that are associated with a specific machine. When defining a share for a cluster, you will need to create a File Share resource. By defining the share as a cluster resource, it can be relocated by MSCS, as needed.

Create a new resource by clicking the right mouse button while the cursor is on the right pane of the Cluster Administrator window (see Figure 80 on page 120). Define the share as a **File Share** resource in the same group as the corresponding SANergy Volume resource (see Figure 86).

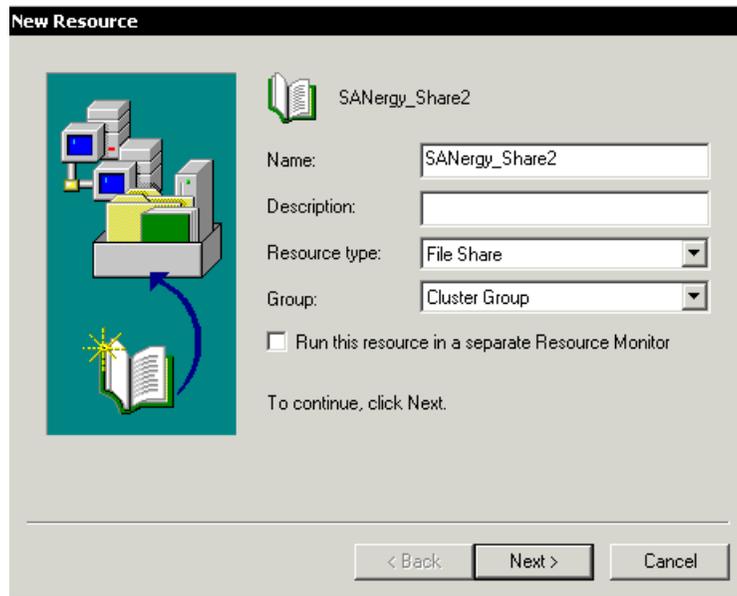


Figure 86. Defining a File Share resource

Allow the same nodes that can host the SANergy Volume resource to host this resource, as well.

When defining dependencies for the File Share resource, be certain to make it dependent upon the SANergy Volume it will reside upon (see Figure 87). In our example, the CIFS share SANergy\_Share2 will serve the data on the volume SANergy\_Disk2. We therefore make this File Share resource dependent upon the volume, since it should not start until the volume is online to the cluster.

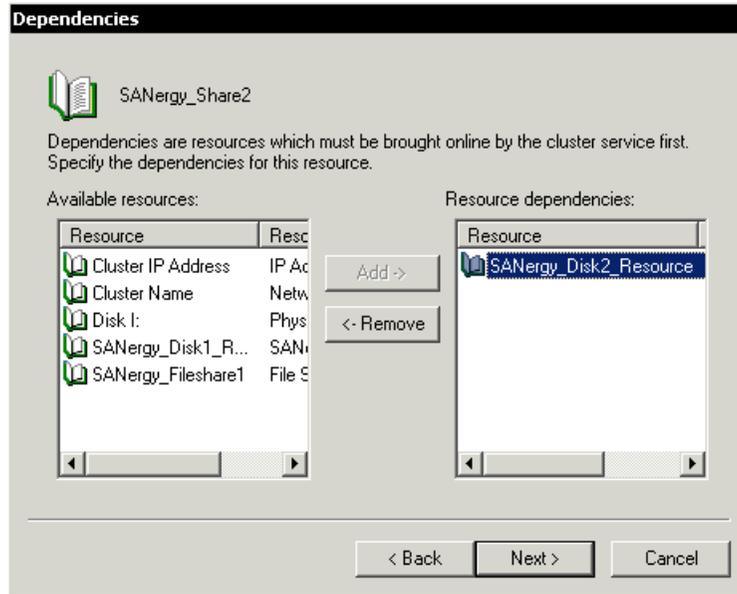


Figure 87. Making File Share dependent upon SANergy Volume resource

The File Share Parameters window of the New Resource wizard will ask you for the parameters specific to this resource. Be certain that the **Path** parameter corresponds to the drive letter that this volume is assigned on each node. In our example, the disk SANergy\_Disk2 has the drive letter K: on every node in the cluster (see Figure 68 on page 111). The path shared by SANergy\_Share2 should then be specified as **K:\** (see Figure 88).

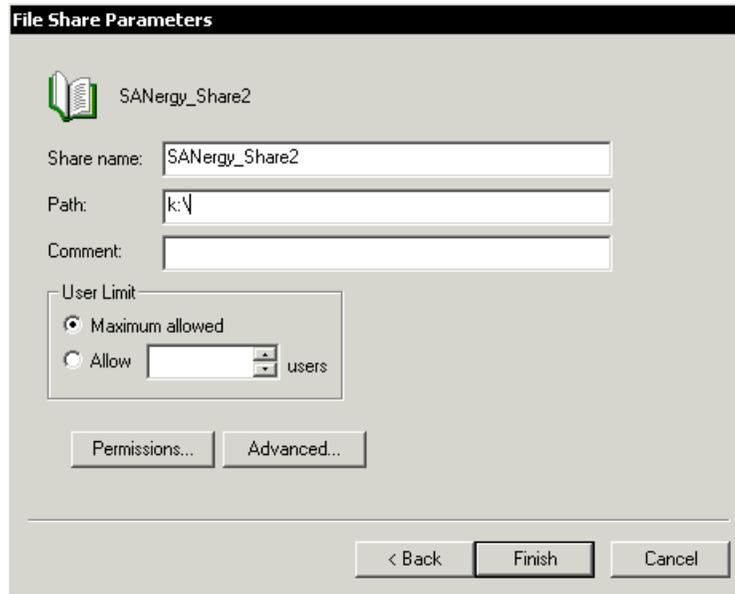


Figure 88. Defining CIFS share parameters

### 5.1.3.8 Map the CIFS share from SANergy host

In order for SANergy to fuse traffic from hosts, those hosts must mount the CIFS share being served from the SANergy MDC. In the case of the clustered MDC we just configured, those hosts must mount the CIFS share we defined to this cluster (see Figure 89). This share name will be specified using the UNC name of `\\cluster_name\file share`.

After you have mapped the CIFS share, validate both functionality and security settings by creating, modifying and deleting a file on that share.

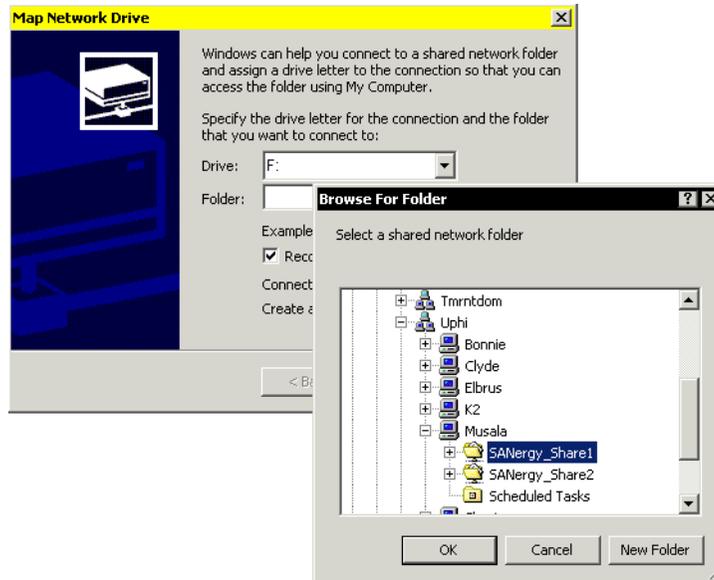


Figure 89. Map like any other CIFS share

### 5.1.3.9 Validate configuration

At this point, I/O to the mounted CIFS share should now be fused and operating at SAN-level performance instead of LAN speeds. Double-check this by starting the SANergy Setup application and performing write and read tests to the mapped drive (see Figure 90). Under ideal circumstances, you should see the same performance for the mapped drive that you find from the MDC node that owns the disk and does direct I/O to the disk.

For SANergy performance related information, please review Chapter 4, “Performance” on page 89 of this Redbook.

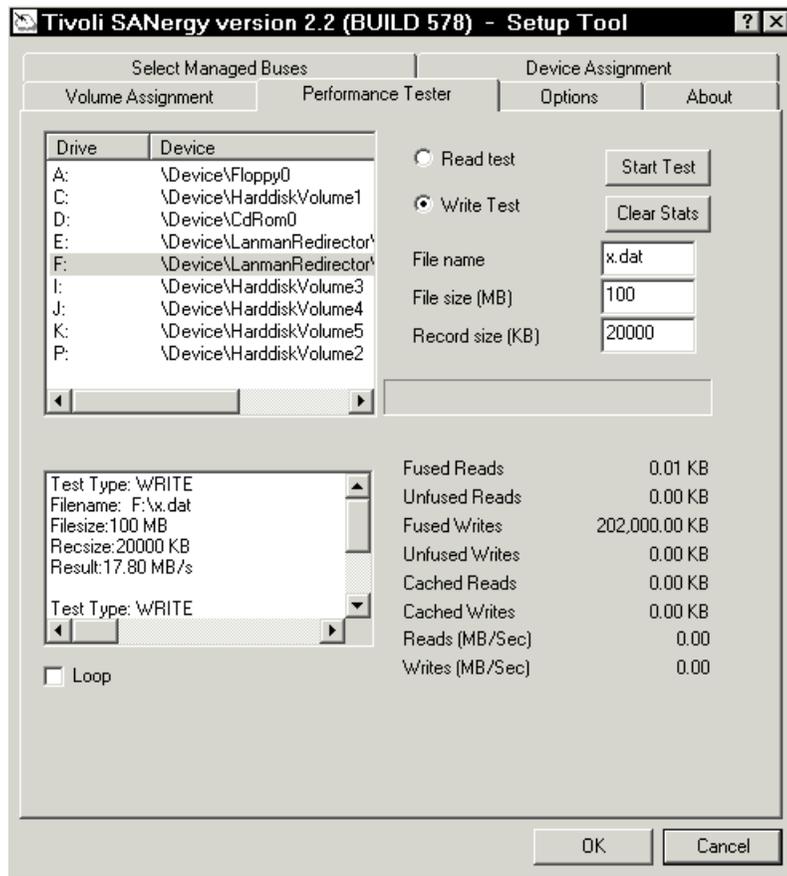


Figure 90. Validating the installation using SANergy Setup

You should also validate that the resources fail over successfully. From whichever node is currently hosting the group, right-click the group and select **Move Group** (see Figure 91).

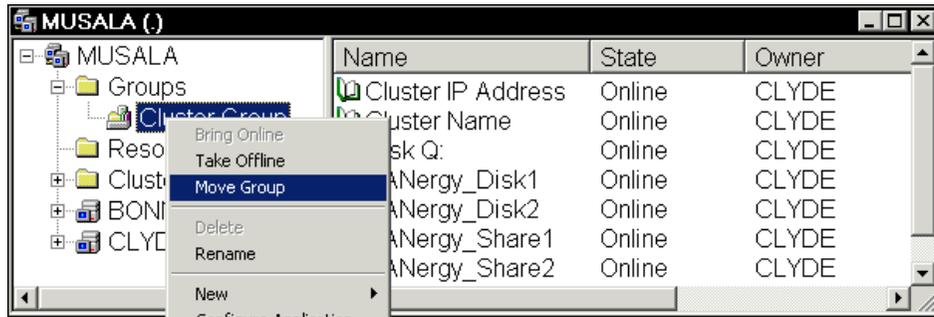


Figure 91. Move MSCS group to test failover

If you have the actual group selected in the left pane of Cluster Administrator, you will see the individual resources go offline on the current node (bonnie) and then online on the new node (clyde), as illustrated in Figure 92 and Figure 93.

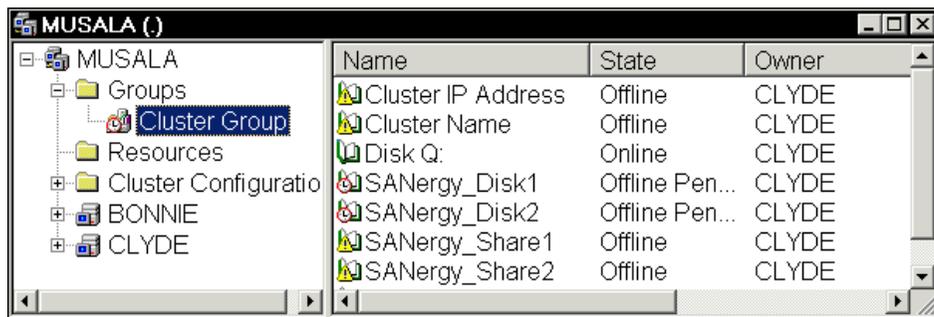


Figure 92. MSCS group going offline on current node

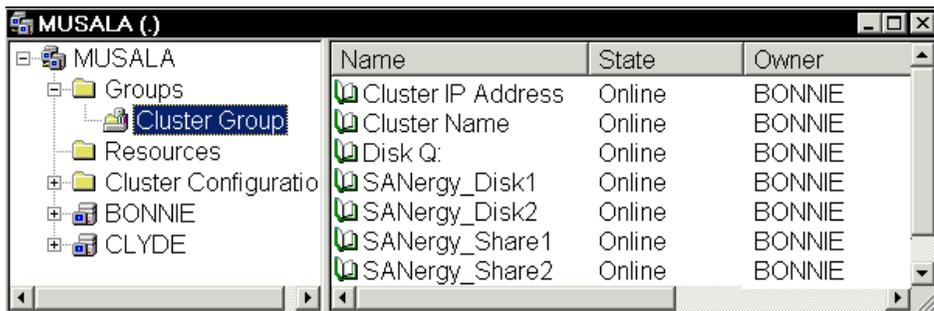


Figure 93. MSCS group online on the other node

We tested our configuration by running a continuous performance test on the SANergy host (Elbrus) and then initiating a failover. The resources moved successfully. The performance test on Elbrus failed temporarily, but could immediately be restarted. This failure was caused by the network resource \\musala\SANergy\_Share1 temporarily being duplicated during the failover. This is a characteristic of MSCS, not SANergy.

---

## 5.2 Sharing Windows Dynamic Disks and Stripe Sets

SANergy allows you to share Windows 2000 Dynamic Disks or Windows NT Stripe Sets. *Stripe Set* is Microsoft terminology for their implementation of a striped RAID volume without parity (RAID 0), on Windows NT. While Windows NT implements simple Striped Sets, Windows 2000 enhances this capability with the addition of the *Dynamic Disk* technology. Dynamic disks allow the adding and removing of disks at any time, without taking the entire volume offline. This technology is similar to the logical volume management applications available on UNIX platforms.

A Dynamic Disk can be formatted as a striped volume. The term *Striped Volume* will be used in this document to refer to either an NT Stripe Set or a Windows 2000 Dynamic Disk formatted as a striped volume.

The ability to share striped volumes can significantly enhance the performance and manageability of a configuration. For example, many SAN disk subsystems have the capability to create hardware-based RAID 5 (striping with parity) volumes. While RAID 5 offers the advantage of fault tolerance, it does so at the expense of slower write I/O operations.

However, this limitation can be partially offset by the use of striped volumes. By creating a striped volume comprising several of the RAID 5 volumes, a fault tolerant implementation with faster read and write I/O is possible. This configuration also has the advantage of offering a single logical volume to be managed instead of several, smaller volumes. If using a Windows 2000 dynamic disk for the striped volume, the manageability is enhanced by the ability to dynamically add or remove disk volumes.

Since SANergy does not alter the design or capabilities of an operating system or application (other than redirecting LAN I/O over the SAN), there are limitations to sharing striped volumes. Since Windows NT Stripe Sets and Windows 2000 Dynamic Disks are proprietary and incompatible, it is not possible for a Windows NT machine to access a Windows 2000 Dynamic Disk. Therefore, Windows NT hosts cannot be part of a SANergy configuration where a Windows 2000 Dynamic Disk is being shared. Note that there is backward compatibility, so that Windows 2000 hosts can safely share Windows NT Stripe Sets.

For similar reasons, at the current time, SANergy cannot share the Windows software implementation of striped volumes *with* parity (RAID 5) or spanned Dynamic Disks. Obviously, hardware RAID implementations do not have this limitation, as the operating systems and applications are unaware of the striping on the actual hard disks.

We discuss the implementation of SANergy and Windows striped volumes in the following three sections:

- SANergy and Windows NT Stripe Sets
- SANergy and Windows 2000 Dynamic Disks
- SANergy performance using striped volumes

## 5.2.1 SANergy and Windows NT Stripe Sets

This procedure documents the steps for sharing an NT Stripe Set with SANergy. Figure 94 shows a basic outline of the configuration we used in creating this section of the Redbook. Two Windows NT 4.0 machines were used — *rainier* was configured as the SANergy MDC, and *pagopago* was the SANergy host. Five disk volumes were available to both via the SAN. These disks were merged into a Windows NT Stripe Set.

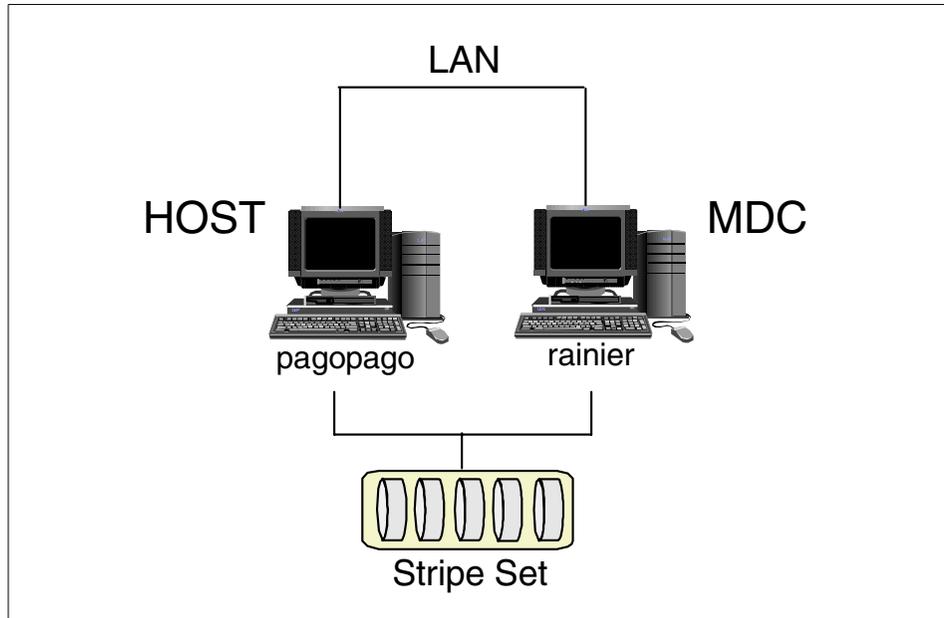


Figure 94. Windows NT Stripe Set testing configuration

### Note

Since some readers will already be familiar with the processes described in this section, a concise outline of those steps is presented here. The steps are then presented in greater detail in the following sections.

1. Gain access to SAN disks from MDC.
2. Create a Stripe Set using SAN disks. Format the new volume using NTFS.
3. Back up the disk configuration via Disk Administrator.
4. Install SANergy and any updates. Give MDC ownership of the volume and devices.
5. Create a CIFS share for this volume.
6. Gain access to the SAN disks from the SANergy hosts. You may have to reboot to get current view since last changes on MDC.
7. Note the current disk configuration via Disk Administrator.
8. Restore to the host the disk configuration backed up from the MDC. This will restore the Stripe Set information.
9. Check the host's disk configuration and fix any discrepancies caused by restoring MDC's configuration.
10. Mount the CIFS share served from the MDC.
11. Install SANergy and any updates.
12. Validate installation by testing for fused performance.

#### 5.2.1.1 Connect the MDC to the SAN

The disks to be attached via the SAN must be visible to Disk Administrator. At this time, you need to make certain there are no partitions defined on those disks.

### 5.2.1.2 Create a Stripe Set

While the disks are visible in Disk Administrator, select all of the disks to be included in the Stripe Set. This can be done by pressing the Control key and clicking the left mouse button on each disk, one at a time. When all of the desired disks are selected, build the Stripe Set by selecting the menu item **Partition -> Create Stripe Set** (see Figure 95).

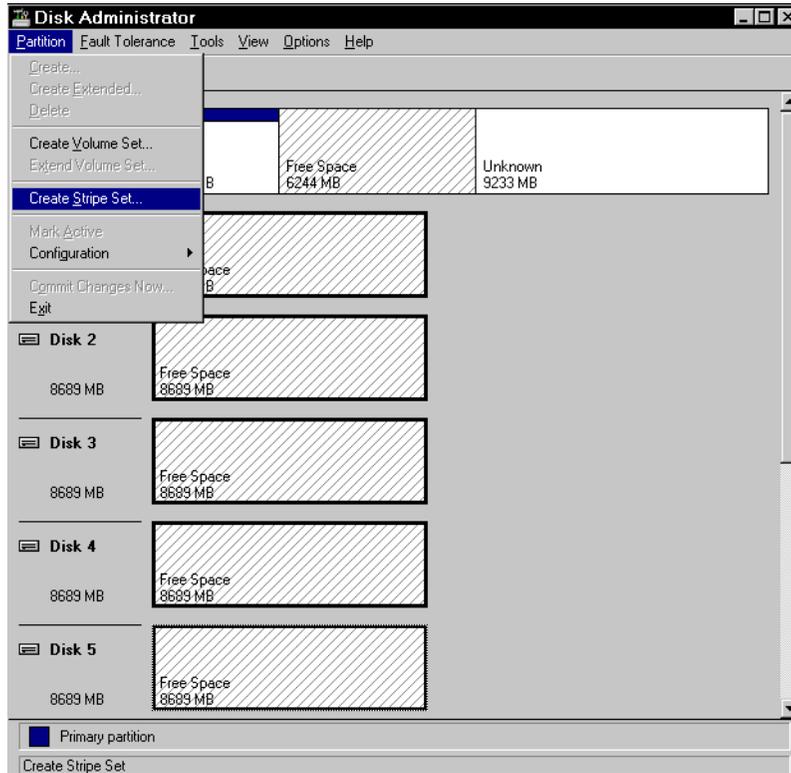


Figure 95. Create a Stripe Set using Disk Administrator

Commit the changes to your disk configuration by right-clicking on one of the disks and selecting the menu item **Commit Changes** (see Figure 96). Before you can proceed any further, you will need to reboot the NT machine.

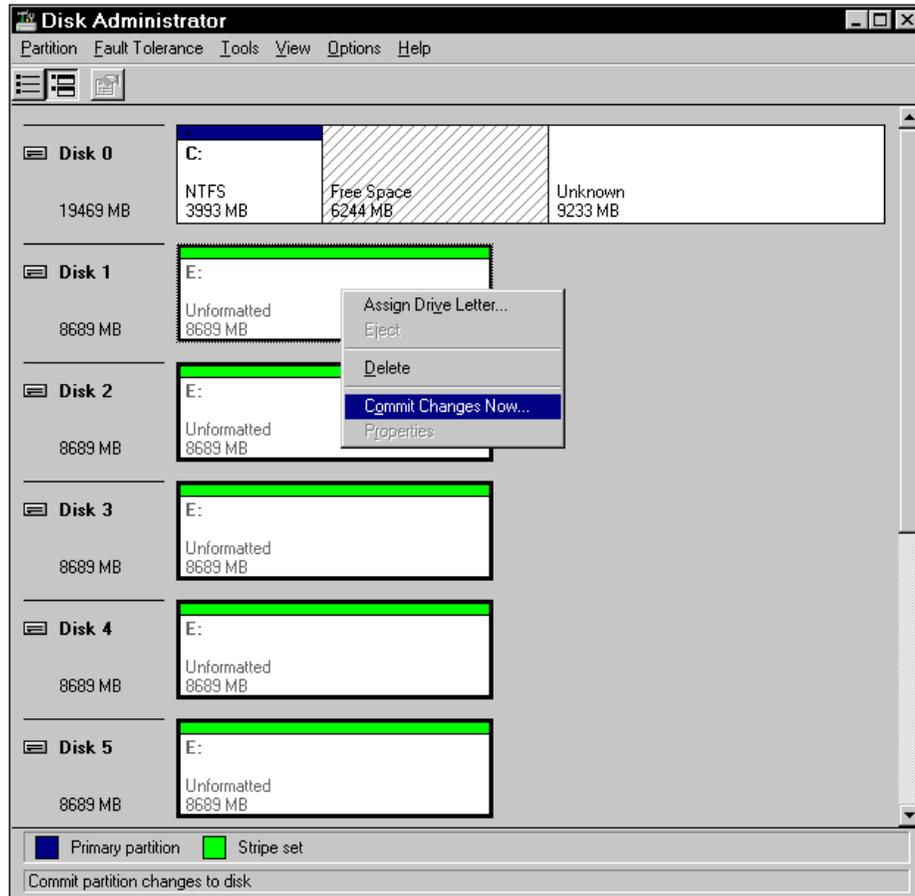


Figure 96. Commit changes to disk configuration

After the machine is restarted, open Disk Administrator again. Format the Stripe Set by right-clicking on one of the disks of the Stripe Set and selecting **Format** from the pop-up menu (see Figure 97).

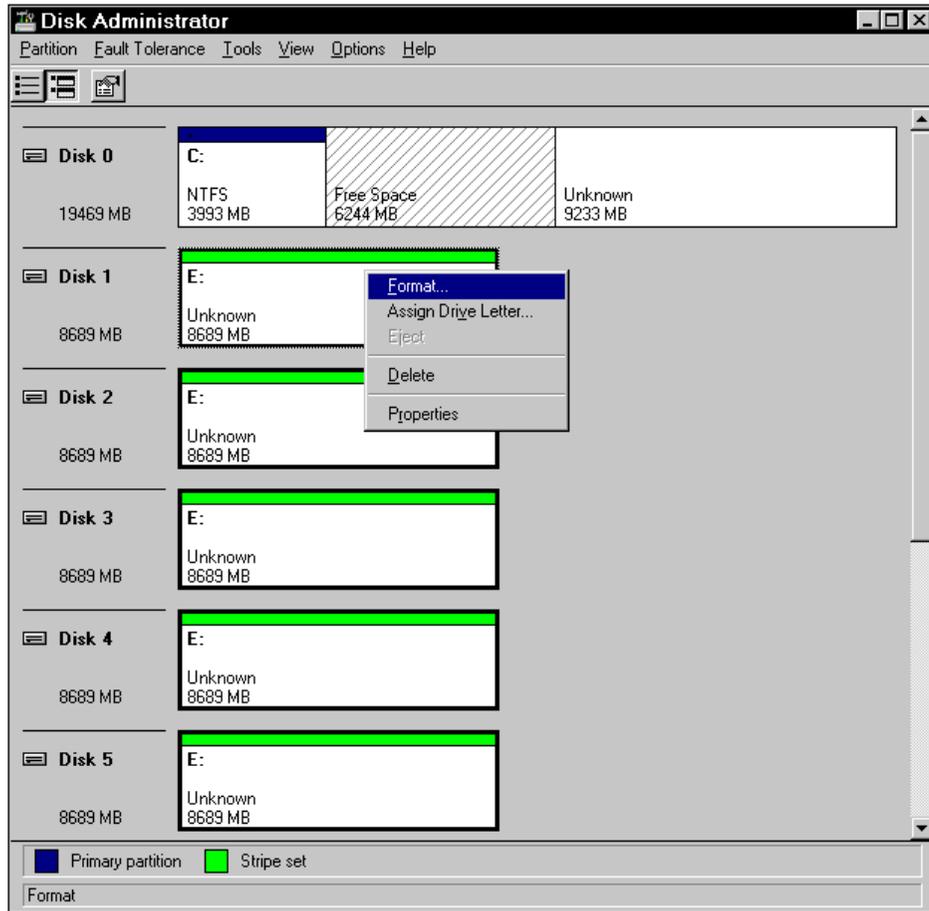


Figure 97. Format Stripe Set

Select **NTFS** for the file system and give it a label compliant with your internal standards (see Figure 98).

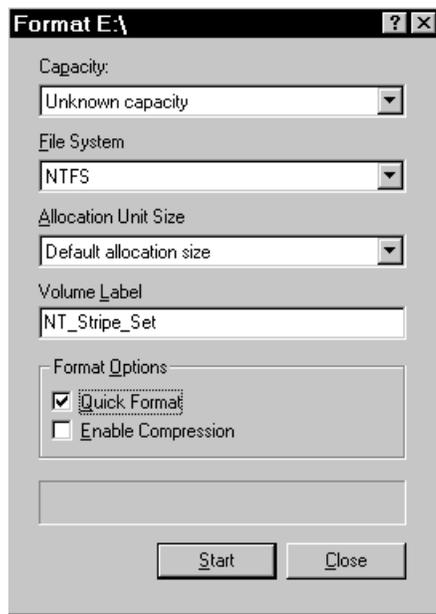


Figure 98. Format parameters

### 5.2.1.3 Back up the disk configuration

Once the Stripe Set has been formatted, back up the MDC's disk configuration using Disk Administrator. Select the menu item **Partition -> Configuration -> Save** (see Figure 99). You will need a formatted floppy disk to store the configuration.

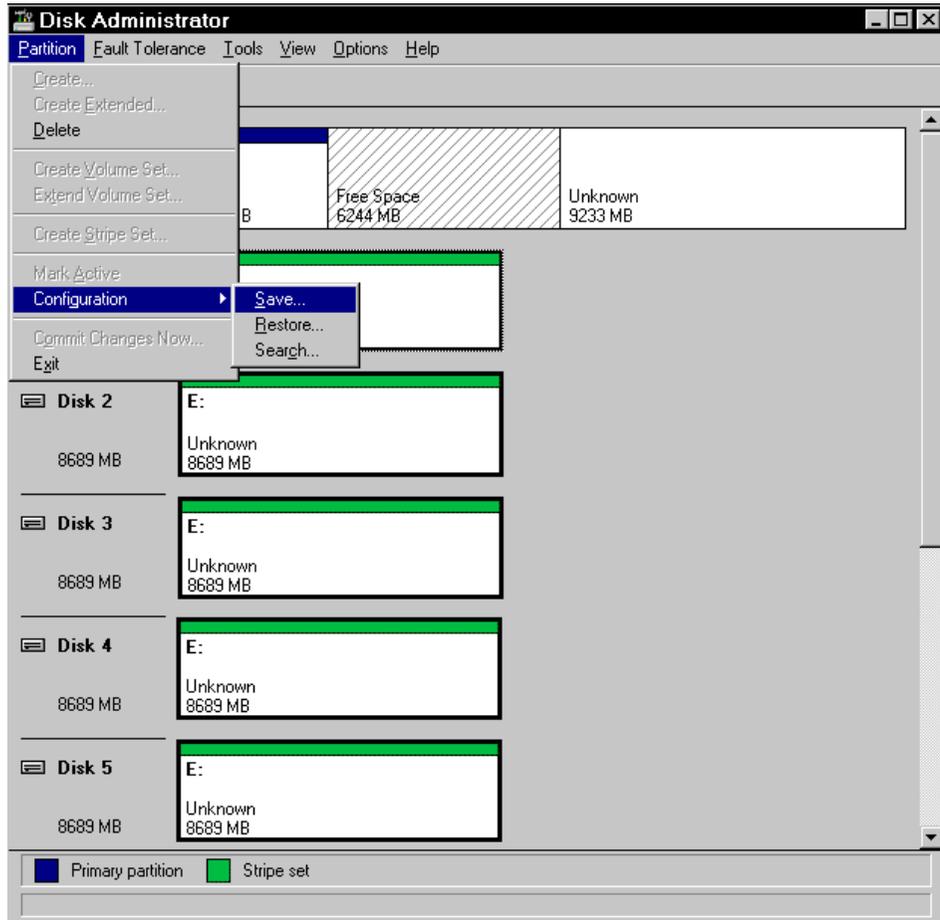


Figure 99. Saving the MDC's disk configuration

#### 5.2.1.4 Create a CIFS share

Define a CIFS share for the Stripe Set, using normal techniques. See Figure 100 for the share we defined on our testing configuration's MDC.

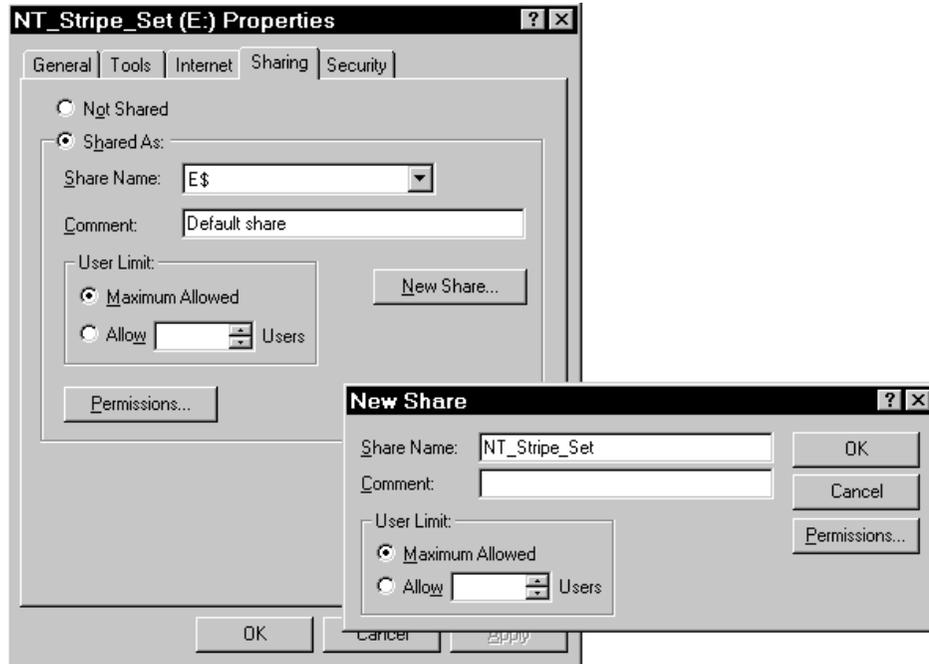


Figure 100. Defined a CIFS share for the Stripe Set

### 5.2.1.5 Install and configure SANergy

Install and configure SANergy and any current updates on the MDC (see Chapter 2, “SANergy with a Windows MDC” on page 19, for more information). When configuring SANergy, give the MDC ownership of all devices in the Stripe Set (see Figure 101) as well as the Stripe Set volume itself (see Figure 102).

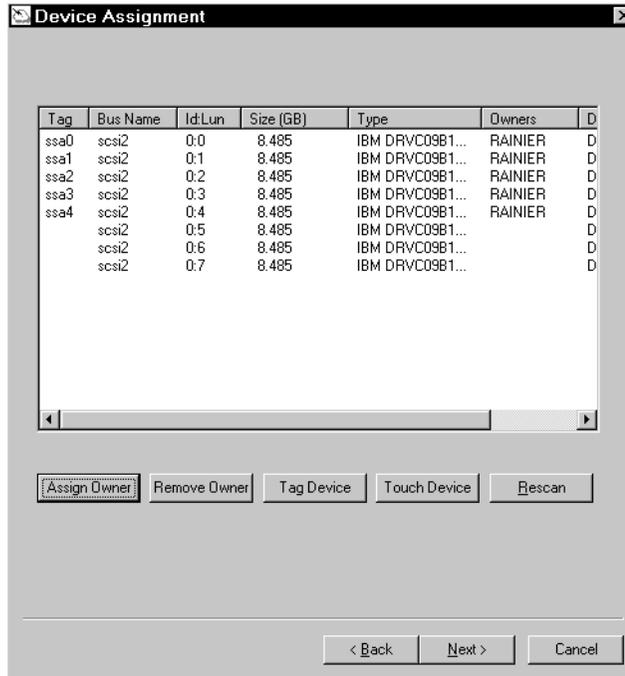


Figure 101. Set ownership of devices in Stripe Set to the MDC

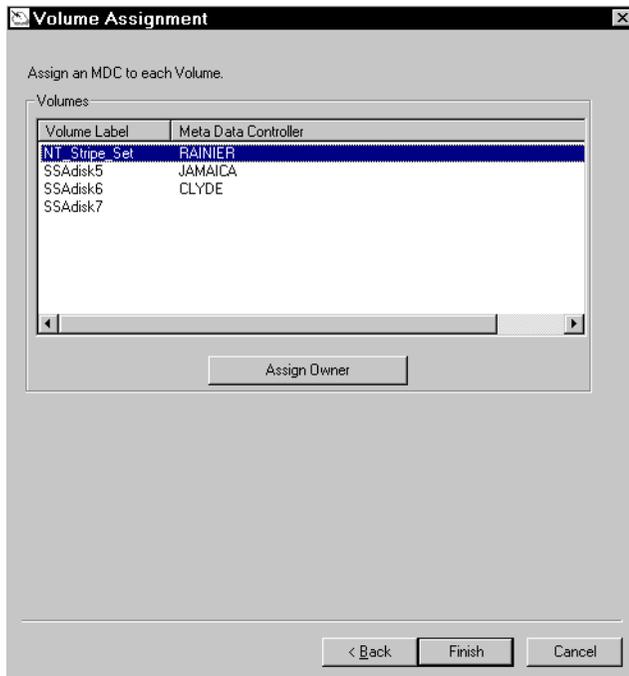


Figure 102. Assign the volume to the MDC

#### 5.2.1.6 Give SANergy hosts access to the SAN

Validate that the hosts which will share the data have access to the disks that comprise the Stripe Set. Use Disk Administrator to display the accessible disks (see Figure 103). You may need to reboot in order for the machine to detect the disks.

#### 5.2.1.7 Document the host's current disk configuration

Use Disk Administrator to view the current disk configuration on the SANergy host. Note what partitions are on each disk and what drive letters are assigned to them, if any (see Figure 103). For example, on the host displayed in Figure 103, partition 2 on disk 0 is assigned C:, partition 5 is assigned D: and partition 6 is assigned E:.

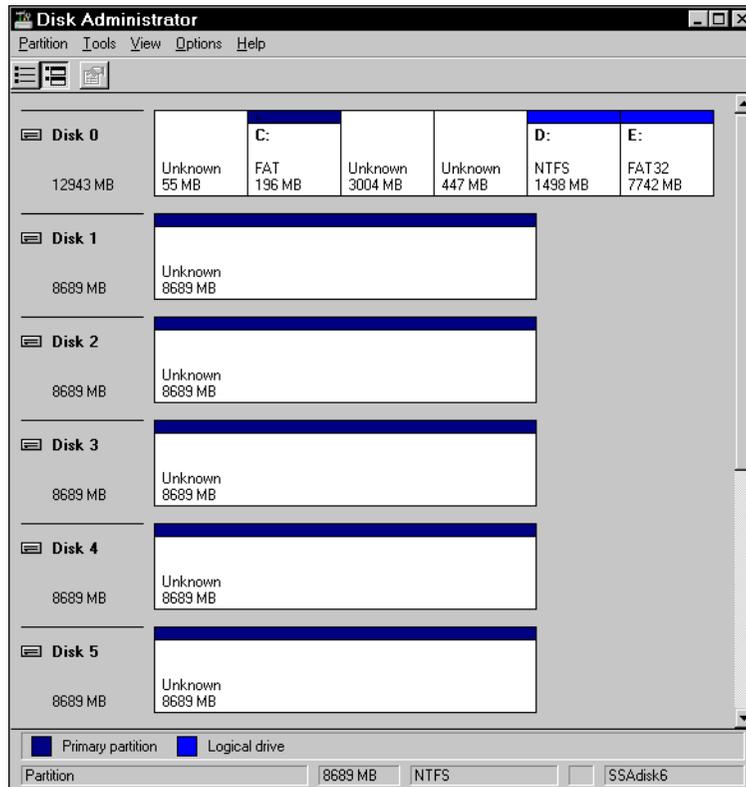


Figure 103. View and document your disk configuration

The next step is to restore the disk configuration from the MDC. This may jumble the disk configuration on the hosts, except for the new Stripe Set. Therefore, you may see various problems until you fix the configuration (we have documented those steps next).

However, this also means that you cannot restore the disk configuration from another machine, if the existing configuration contains a Stripe Set or RAID5 volume. That volumes's information will be overwritten! As an added precaution, it is always recommended that you back up your disk configuration *and* prepare a set of recovery disks before performing the next step.

### 5.2.1.8 Restore the MDC's disk configuration to the hosts

Restore the MDC's disk configuration to the host machine. Do this by starting Disk Administrator and selecting the menu item **Partition -> Configuration -> Restore** (see Figure 104).

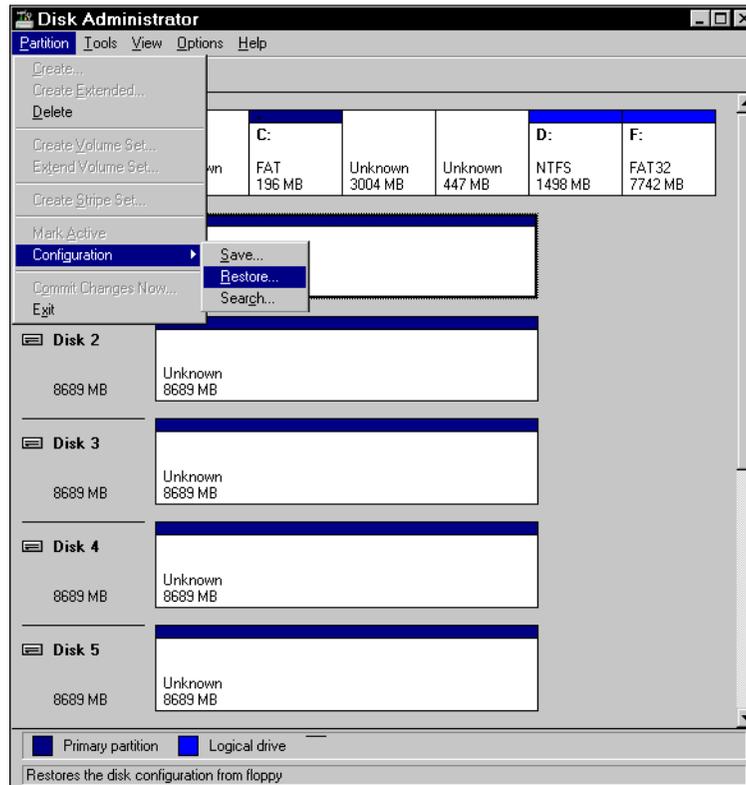


Figure 104. Restore the disk configuration

You will be prompted to insert the recovery disk generated on the MDC. After the restore, you will be prompted to reboot the machine.

### 5.2.1.9 Fix any discrepancies in host's disk configuration

Once the host restarts, start Disk Administrator. Look at the current configuration and, if necessary, modify the configuration to match what existed before. For example, Figure 105 displays the disk configuration on our host after the reboot.

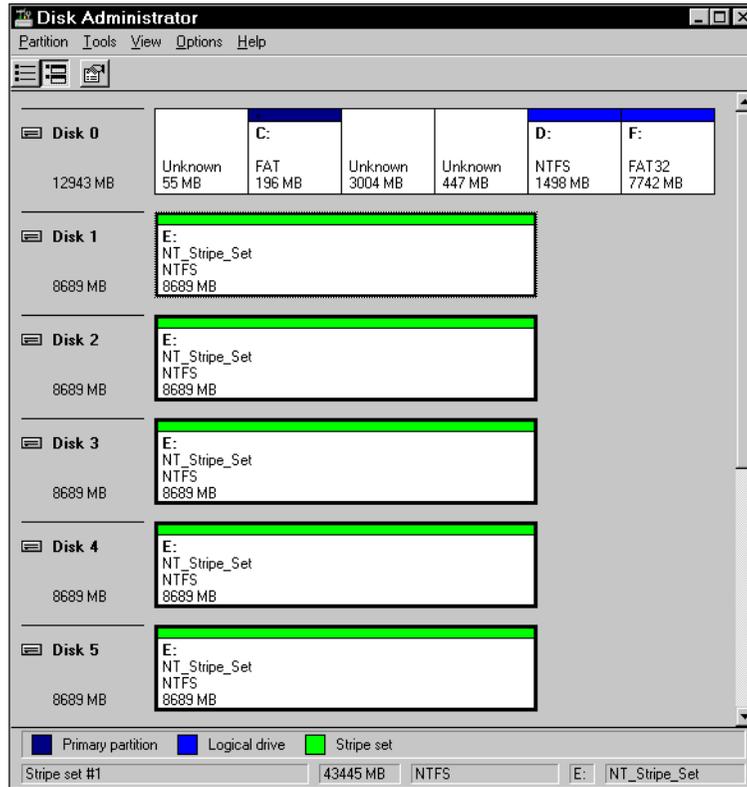


Figure 105. Disk configuration after restore; Disk 0 definition is now wrong

Note that partition 6 on disk 0 now has the wrong drive letter. Any applications or shares directed to that drive letter are now broken. We manually changed that drive letter to return disk 0 to its original state (see Figure 106). To change or remove a drive letter, right-click the volume and select **Assign Volume** from the pop-up menu.

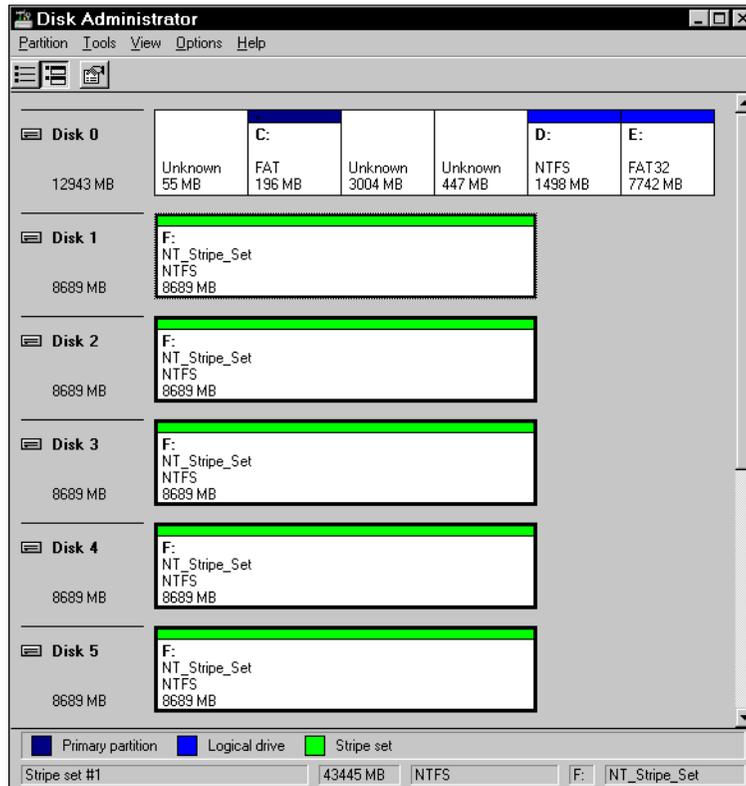


Figure 106. Disk configuration after restore and manual update

Do not assign a drive letter to the Stripe Set. You do not want hosts attempting to access the volume directly, but rather to use SANergy. If you have multiple NT machines accessing the same NTFS volume natively you will get data corruption.

### 5.2.1.10 Mount the CIFS share

After the disk configuration is restored and, if necessary, any problems corrected, you can now mount the CIFS share from the SANergy MDC. You mount the Stripe Set share the same was as you would a normal volume (see Figure 107).

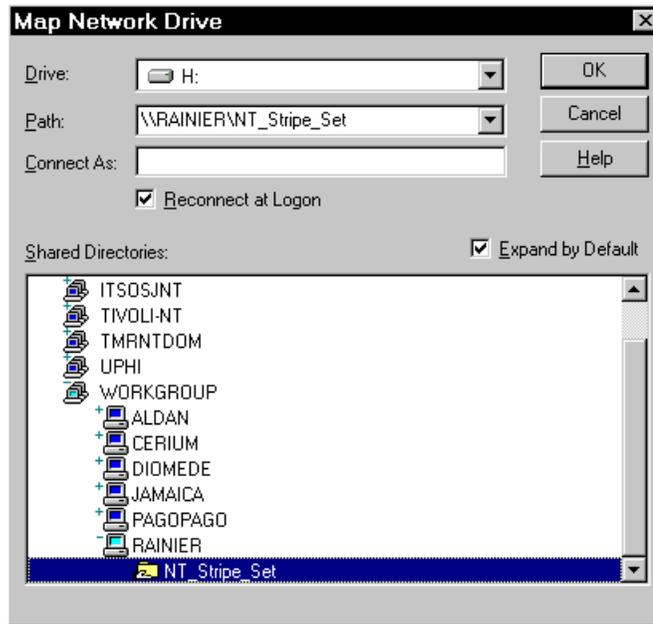


Figure 107. Mount the CIFS share

#### **5.2.1.11 Install and configure SANergy**

Install and configure SANergy and any current updates on the host machines. From this point forward, there is nothing unique to this implementation. As in the previous steps, the MDC should have ownership of both the disk devices and Stripe Set volume (see Figure 101 on page 142 and Figure 102 on page 143).

#### **5.2.1.12 Validate installation**

As we always recommend, use the SANergy Setup tool to test fusing to the mapped drive (see Figure 90 on page 130).

#### **Note**

Since Windows 2000 can import Windows NT Stripe Set definitions, it is possible to perform a procedure equivalent to the one documented in this section to allow a Windows 2000 SANergy host to share an NT Stripe Set.

## 5.2.2 SANergy and Windows 2000 Dynamic Disks

This procedure details how to use SANergy to share a Windows 2000 Dynamic Disk configured as a striped volume. Figure 94 shows a basic diagram of the configuration we used in testing Dynamic Disks and SANergy. We used one Windows 2000 Advanced Server MDC (bonnie), which owned the devices and Dynamic Disk volume. We tested two Windows 2000 hosts (elbrus and clyde), also running Advanced Server. Each of the machines had access to five disks, which we configured into a Dynamic Disk.

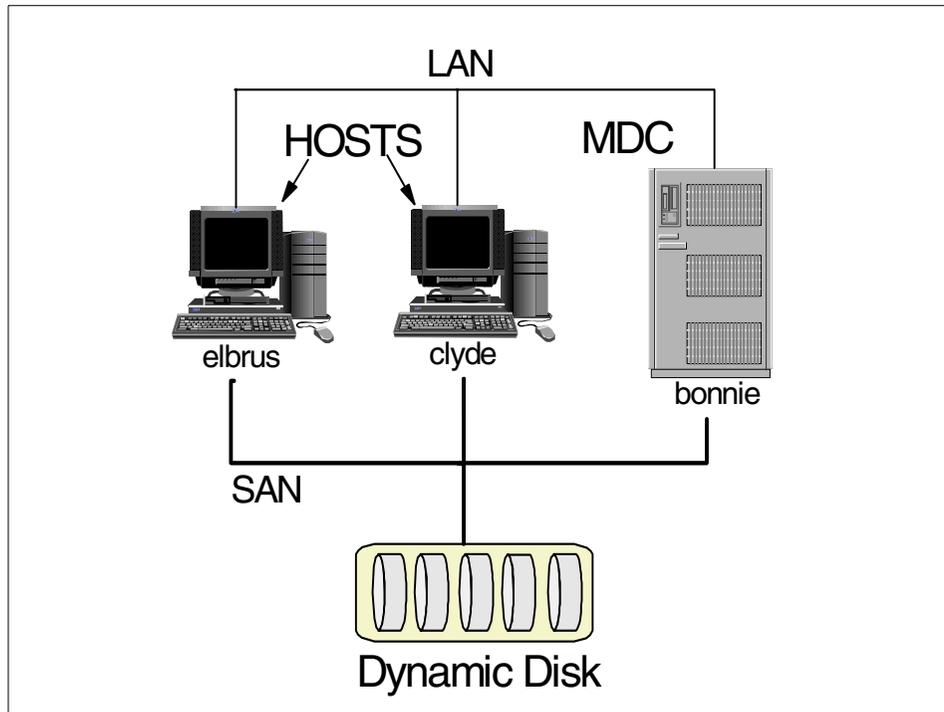


Figure 108. Dynamic disk testing configuration

#### **Note**

Since some readers will already be familiar with the processes described in this section, a concise outline of those steps is presented here. The steps are then presented in greater detail in the next sections.

1. Gain access to the SAN disk from SANergy MDC and hosts.
2. Create a Dynamic Disk formatted as a striped volume on the MDC.
3. Import Dynamic Disk information into each SANergy host. Do not assign a drive letter.
4. Create a CIFS share on the MDC and map on each host.
5. Install and configure SANergy on the MDC. The MDC should own all disks that make up the Dynamic Disk volume as well as the volume itself.
6. Install and configure SANergy on all hosts.
7. Validate installation by testing fused performance.

#### **5.2.2.1 Gain access to the SAN from hosts and MDC**

Configure your SAN so that the MDC has access to each of the disks to be used in the Dynamic Disk. Figure 109 shows the volumes on our MDC, before creating or formatting a Dynamic Disk.

This figure is of the Disk Management applet in the Computer Management utility. You can start Computer Management by selecting **Start -> Programs -> Administrative Tools -> Computer Management**.

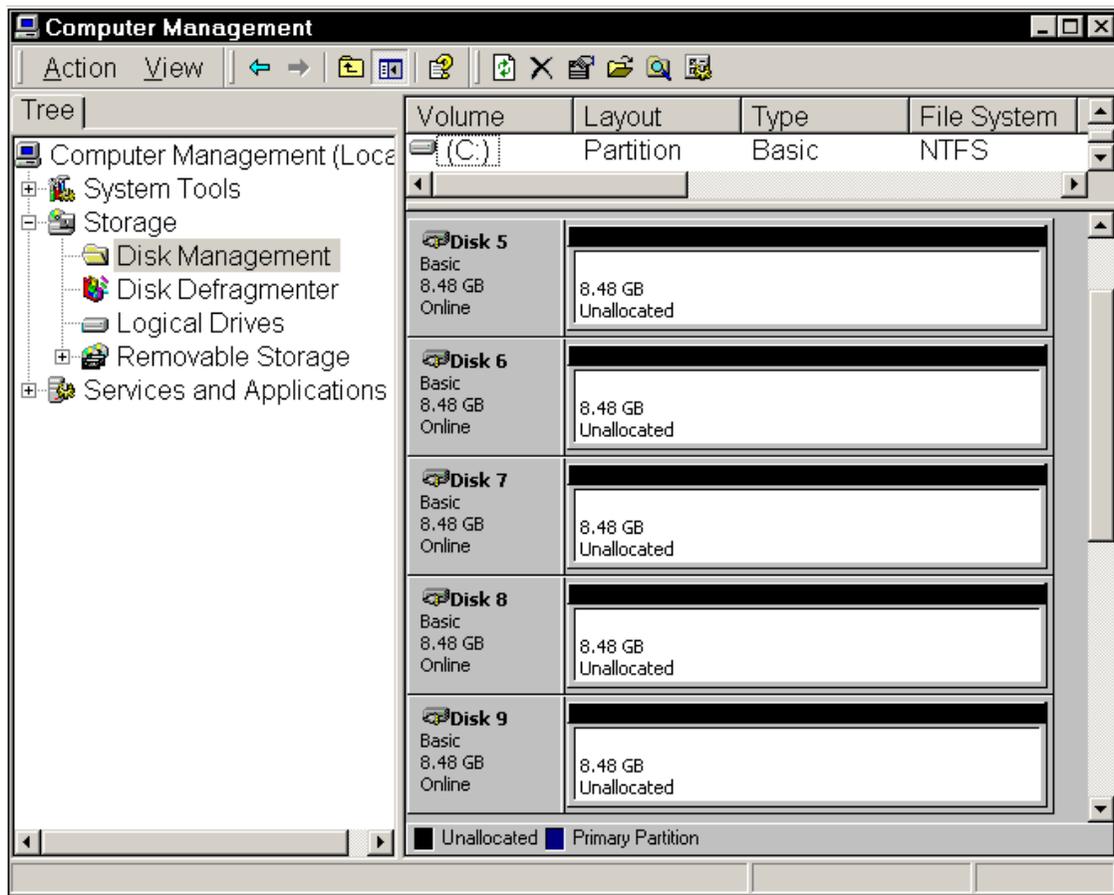


Figure 109. Disk Management applet before creating Dynamic Disk

### 5.2.2.2 Create a Dynamic Disk volume on MDC

Create a Dynamic Disk by right-clicking on one of the disks to be converted. Select **Upgrade to Dynamic Disk** on the pop-up menu (see Figure 110).



Figure 110. Begin conversion of basic disks to Dynamic Disk

A menu is presented to allow you to select all of the disks to be part of the Dynamic Disk. Select those SAN disks you wish to share and click **OK** to proceed (see Figure 111).

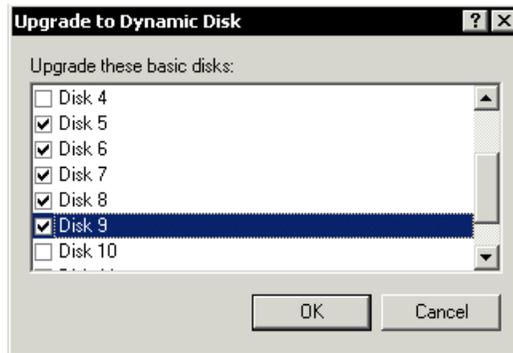


Figure 111. Select disks to upgrade

After the disks have been converted to a Dynamic Disk, create a volume on it by right-clicking on one of the volumes and selecting **Create Volume** from the pop-up menu (see Figure 112).



Figure 112. Initiate the creation of a volume on the Dynamic Disks

This will start the Create Volume wizard. The first panel of this wizard will allow you to validate which disks you wish to be included in the new volume. Add the appropriate disks and select **Next** to proceed (see Figure 113).

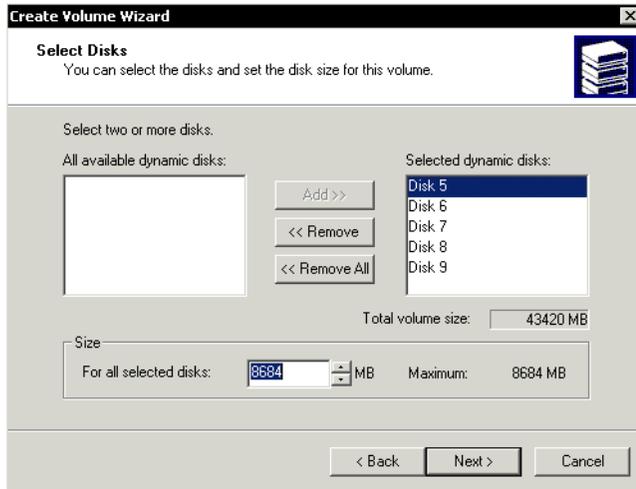


Figure 113. Identify which disks to use in creating the new volume

You can then identify which format to use for this volume. Select **Striped Volume** (see Figure 114).

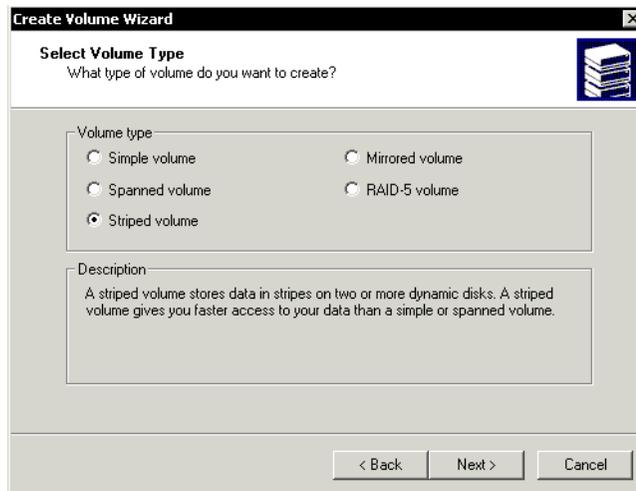


Figure 114. Identify Striped Volume as the format

The next panel allows you to specify the formatting parameters. Select **NTFS** and any other parameters per your standards (see Figure 115). The final panel allows you to validate all of the settings. If they are correct, click **Finish** to start creating and formatting the volume (see Figure 116).

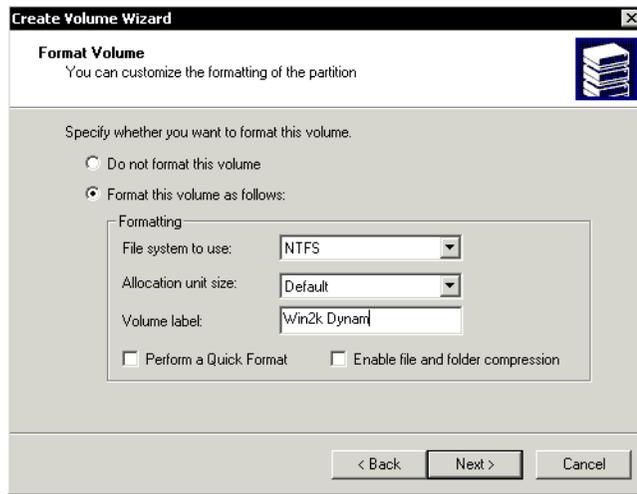


Figure 115. Format volume



Figure 116. Validate the parameters and then proceed

Computer Management shows the progress of the format process on the volume icon itself (see Figure 117).

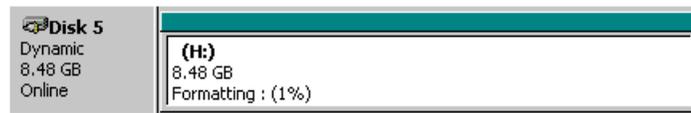


Figure 117. Progress of a format process

### 5.2.2.3 Import Dynamic Disk to SANergy hosts

Start Computer Management on the SANergy host and display the current disks accessible from this machine. If the host can see the same disk that comprises the Dynamic Disk on the MDC, you will see a foreign Dynamic Disk. You can import the Dynamic Disk configuration by clicking your right mouse button on the disk and selecting **Import Foreign Disks** from the pop-up menu (see Figure 118).

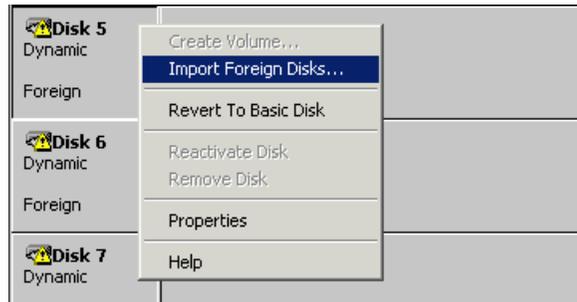


Figure 118. Importing a foreign Dynamic Disk to the host machines

This selection starts the wizard to assist in importing the disks. Select the disk set to be imported and press the **OK** button (see Figure 119). A validation screen will be displayed. Select **OK** if the volume to be imported is the one which you intended (see Figure 120). After the volume information is imported correctly, the status of the disk will change to *Dynamic Online*.

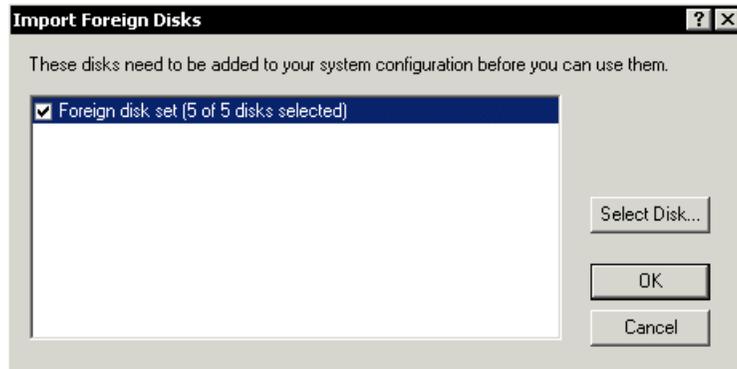


Figure 119. Select the set of disks to be imported



Figure 120. Validate that the correct volume will be imported

After the volume is imported, remove the drive letter so that the volume cannot be accessed directly. Attempting to access a volume from more than one Windows 2000 host simultaneously will result in data corruption (see Figure 121 and Figure 122).

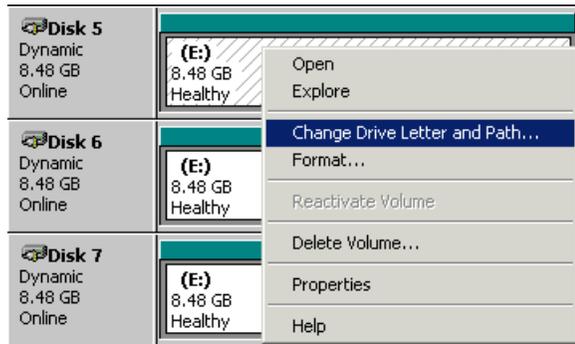


Figure 121. Changing the drive letter of imported volume

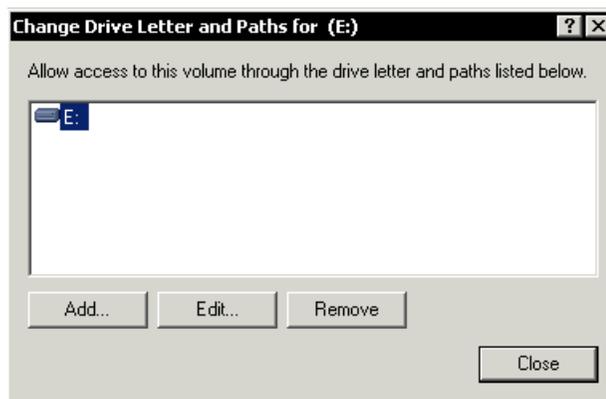


Figure 122. Removing the drive letter to prevent data corruption

#### 5.2.2.4 Create a CIFS share on MDC and map on SANergy hosts

Using standard Windows Networking techniques, create a CIFS share on the SANergy MDC and map that share on all hosts. Figure 89 on page 129 shows an example of mapping a CIFS share.

#### 5.2.2.5 Install and configure SANergy

Install and configure SANergy on the MDC and all hosts (see Chapter 2, “SANergy with a Windows MDC” on page 19, for more information). When configuring SANergy, the devices that comprise the Dynamic Disk must be owned by the MDC (see Figure 123). Likewise, the volume that is on the Dynamic Disk must belong to the MDC (see Figure 124).

| Tag  | Bus Name | Id:Lun | Size (GB) | Type           | Owners | Disk Number |
|------|----------|--------|-----------|----------------|--------|-------------|
| ssa0 | scsi3    | 3:0    | 8.480     | IBM DRVC09B... | BONNIE | Disk 5      |
| ssa1 | scsi3    | 3:1    | 8.480     | IBM DRVC09B... | BONNIE | Disk 6      |
| ssa2 | scsi3    | 3:2    | 8.480     | IBM DRVC09B... | BONNIE | Disk 7      |
| ssa3 | scsi3    | 3:3    | 8.480     | IBM DRVC09B... | BONNIE | Disk 8      |
| ssa4 | scsi3    | 3:4    | 8.480     | IBM DRVC09B... | BONNIE | Disk 9      |
|      | scsi3    | 3:5    | 8.480     | IBM DRVC09B... |        | Disk 10     |
|      | scsi3    | 3:6    | 8.480     | IBM DRVC09B... |        | Disk 11     |
|      | scsi3    | 3:7    | 8.480     | IBM DRVC09B... |        | Disk 12     |

Figure 123. Device ownership settings

| Volume Label  | Meta Data Controller |
|---------------|----------------------|
| SSAdisk5      | JAMAICA              |
| SSAdisk6      |                      |
| SSAdisk7      |                      |
| Sanergy Dynam | BONNIE               |

Figure 124. Volume assignment settings

#### 5.2.2.6 Validate installation

Validate the installation by using the SANergy Setup application to do a performance test on the mapped volume.

**Note**

Windows NT is not able to import or access Windows 2000 Dynamic Disks. Although Windows 2000 is backward compatible and can access an NT Stripe Set, this compatibility does not work in the reverse direction.

### **5.2.3 SANergy Performance on Stripe Sets and Dynamic Disks**

This section reports on the performance found on the test configurations discussed in this section. SANergy performance is discussed, in detail, in Chapter 4, "Performance" on page 89. This information is only included to give an example of some very basic performance tests done when using SANergy and striped volumes together. These steps can also be used to do a simple validation of any SANergy configuration.

Note that the performance rates achieved are specific to our isolated configuration, and should not be taken as indicative of real results which could be attained in other environments. The relative performance achieved on the different tests is what we are emphasizing.

The performance of Windows NT 4.0 and Windows 2000 Advanced Server in these tests were very similar. The tests that are shown in this section are:

- Performance test on Windows MDC to shared Dynamic Disk/Stripe Set
- Performance test on Windows Host to shared Dynamic Disk/Stripe Set
- Simultaneous performance tests on Windows MDC and 1 Windows Host to shared Dynamic Disk/Stripe Set

#### **5.2.3.1 Performance on SANergy MDC to striped volume**

Figure 125 shows a performance test done using SANergy Setup on an MDC. The disk being shared was selected and a write test was done. This I/O is done directly to the disks, and therefore the Fused Writes and Fused Read statistics do not increment. Note the results of the test, which yielded a 39.55 MB/sec throughput. This specific test was done from a Windows 2000 MDC to a Dynamic Disk formatted as a striped volume.

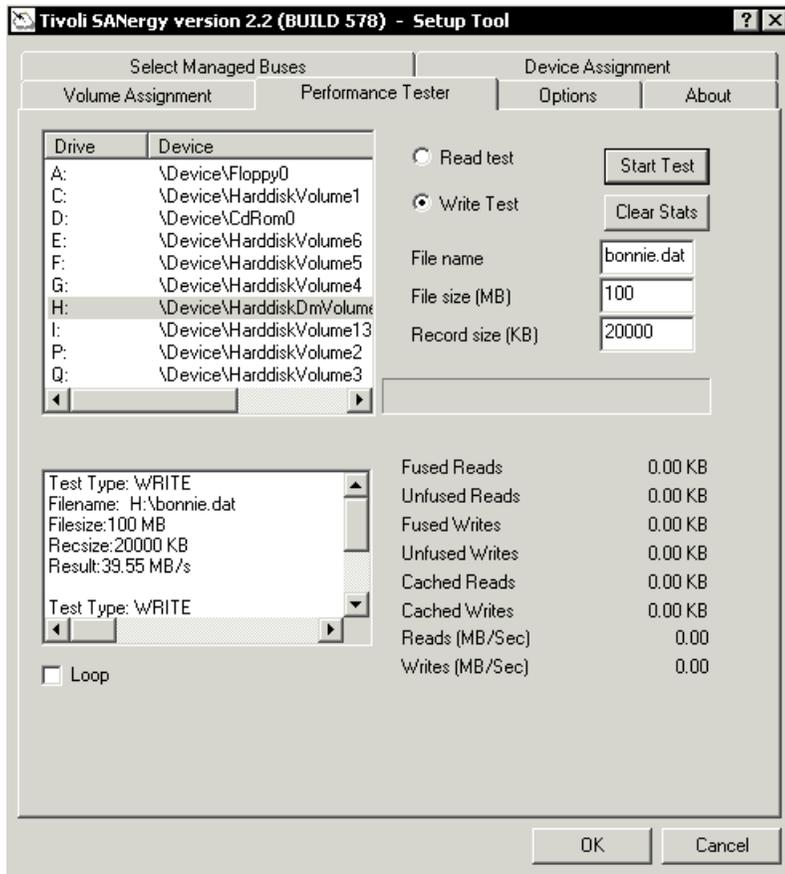


Figure 125. Performance Test of SANergy MDC

### 5.2.3.2 Performance of SANergy host to striped volume

A performance test done on a SANergy host to a striped volume is shown in Figure 126. Note that the throughput (39.30 MB/sec) is consistent with that from the SANergy MDC to the same volume (see Figure 125). Also, the Fused Write statistics increment to show that the I/O is being performed on a fused volume.

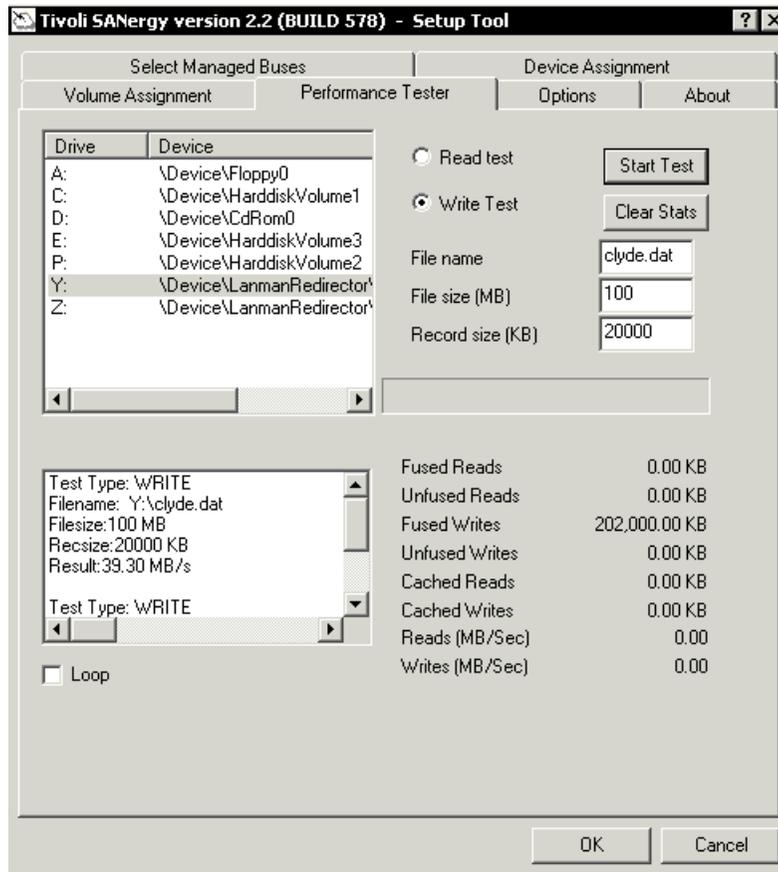


Figure 126. SANergy performance test from host

### 5.2.3.3 Performance on MDC and host with simultaneous I/O

Figure 127 and Figure 128 show the same two performance tests done above, but with the tests running simultaneously. As before, the MDC and host both show almost identical throughput rates. Also, the throughput rates, when combined, equal the rate of the previous tests. This fact indicates that no additional bottleneck is introduced by SANergy when running simultaneous I/O on a shared volume.

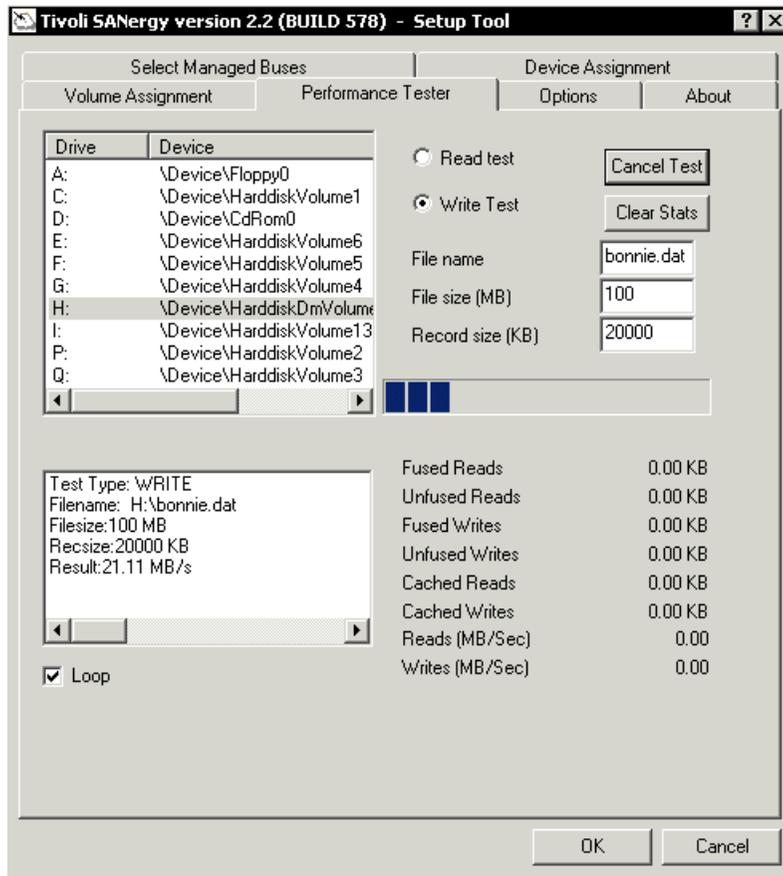


Figure 127. Simultaneous performance test — view of MDC

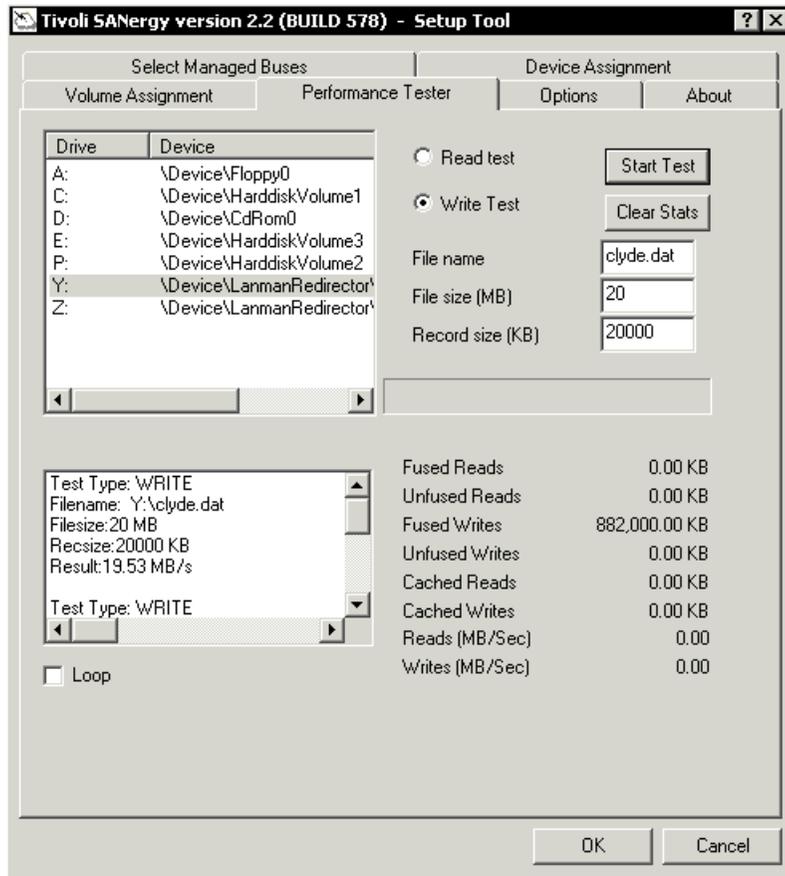


Figure 128. Simultaneous performance test — view of SANergy host

#### 5.2.3.4 Other performance considerations

The next three figures show Performance Monitor reports generated while running SANergy performance tests. The report shows CPU% Utilization and Page Swapping. At no time was any memory swapping detected. The CPU utilization was moderate on tests done while running a single I/O stream (approximately 39 MB/sec) and lower while running simultaneous tests (approximately 18 MB/sec) on multiple machines.

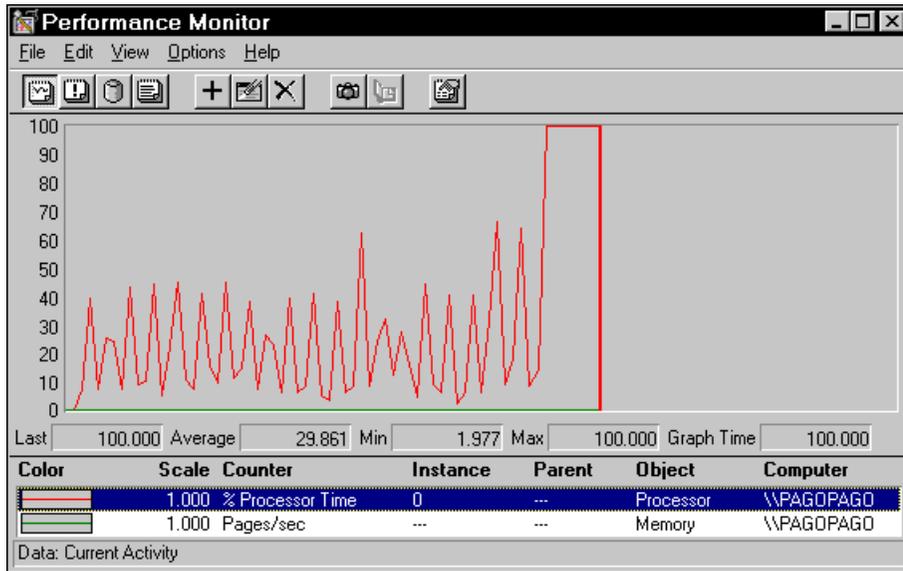


Figure 129. Perfmon report of single 40MB/sec data stream

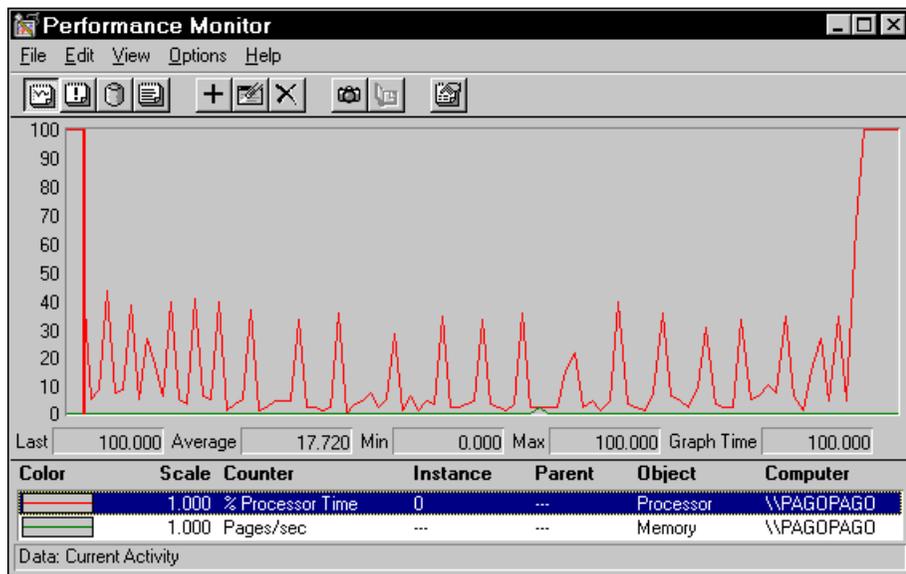


Figure 130. Perfmon report of multiple 20MB/sec data streams — MDC view

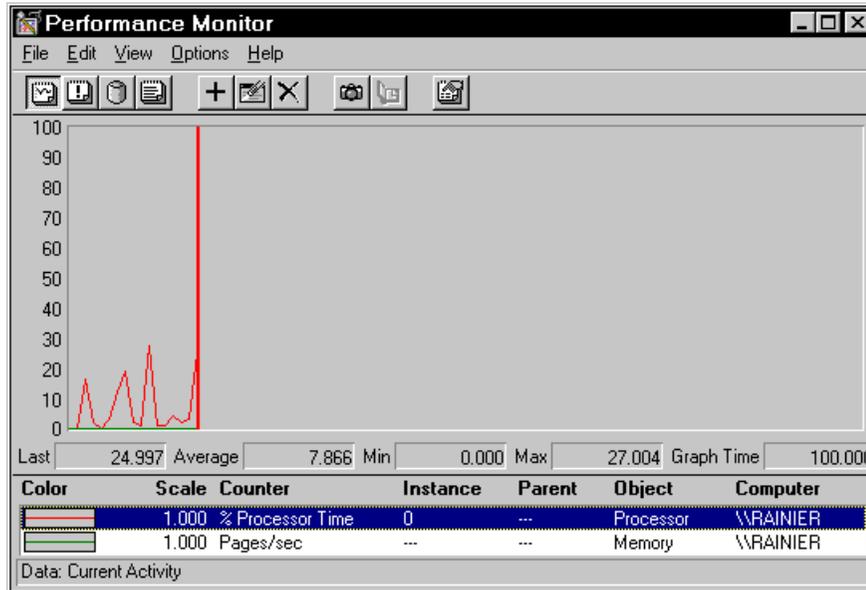


Figure 131. Perfmon report of multiple 20MB/sec data streams — SANergy host

### 5.2.3.5 Performance testing conclusions

This performance testing was done simply to validate the SANergy installation outlined in this section as well as show the potential performance benefits of using SANergy with striped volumes. This test was not run using any specific workload profile. We did not include figures from all testing since the results were almost identical when comparing the performance of the NT Stripe Set and the Windows 2000 Dynamic Disk formatted as a striped volume.

The tests validated that SANergy did function normally with striped volumes. They show that SANergy did not add any additional overhead to I/O on SAN-based disks in tests using large files and large records. The native speed to the Stripe Set and Dynamic Disk were both approximately 40 megabytes per second.

Running the same tests on a SANergy host which utilized SANergy to access the disk gave the same results. Tests with two and three machines writing to the striped volume showed no additional performance degradation due to SANergy. When running two machines (1 MDC and 1 Host) to the striped volume, speeds were approximately 20 MB/sec. When using three machines, the speed was approximately 13 MB/sec.

The CPU utilization was moderate when running only one full-speed, performance test. When running tests from multiple hosts, the throughput for any given host was reduced and the CPU utilization dropped accordingly. No tests showed any memory page swapping. These statistics should be dramatically lower using the SAN/SCSI based I/O versus attempting the same workload using network-based IP traffic on Gigabit ethernet.



---

## Chapter 6. Putting it all together

In the previous chapters we have first covered basic SANergy installations and discussed how to modify parameters for performance tuning. We then discussed how to configure SANergy in particular disk configurations: for high availability and striped performance.

In this chapter, we draw on all of these topics to describe how to build a “super data server” environment that takes advantage of the main benefits of SANergy. This super data server will be a single system owning the entire storage capacity. It makes this storage available to its clients: to file servers, application servers and even to database servers.

The Fibre Channel technology has introduced new possibilities for computing — and the storage discipline has received the most impact. Storage virtualization concepts are being developed on the basis of this new technology and will be increasingly available in the future. We also have many possibilities already available today to realize some of those ideas in a SANergy environment.

As a basis for the configurations in this chapter, we set up a SAN installation with a Microsoft Cluster Server as the MDC named musala (Figure 132). We gave details on setting up an MDC on a Microsoft Cluster Server in 5.1, “MSCS clustering for the MDC” on page 105.

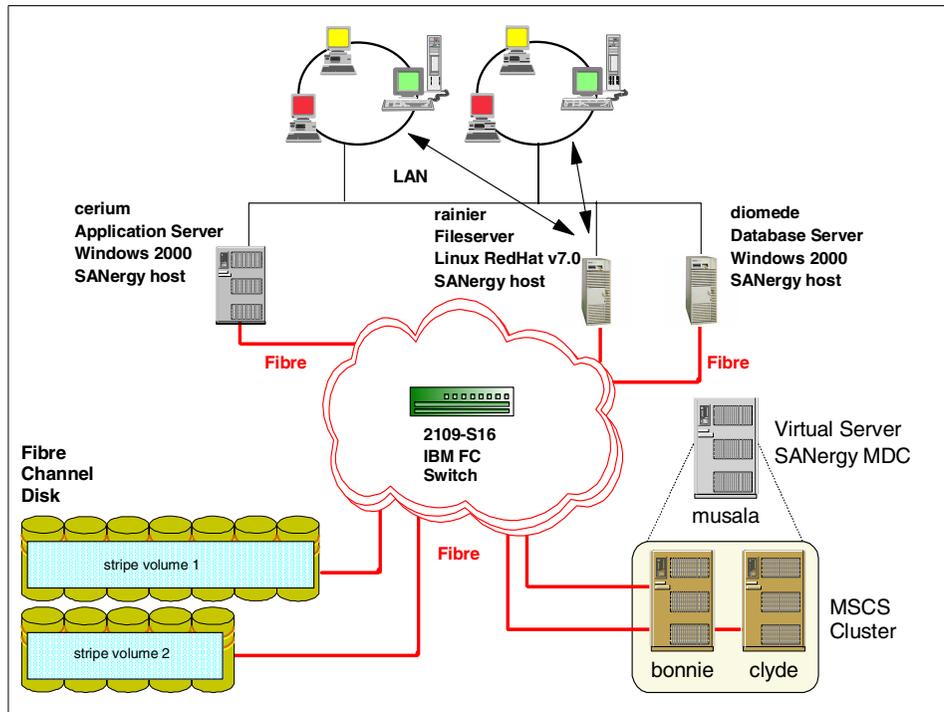


Figure 132. Super data server

The two nodes in the cluster are servers running Windows 2000 Advanced Server and are named bonnie and clyde. The disks being shared are configured on the disk subsystem as two separate striped non-parity volumes (RAID0). Another possibility to achieve a more fault tolerant solution would be to use RAID5 configured disks, that is, striping with parity.

The two configured volumes appear to the Windows servers as separate LUNs. We configured these LUNs into a Windows 2000 Dynamic Disk. We then created a partition on this Dynamic Disk and formatted it as a striped volume. This will be the volume we will share with SANergy. Please see the section on using Windows 2000 striped volumes with SANergy, for more information (5.2, “Sharing Windows Dynamic Disks and Stripe Sets” on page 132).

In order to share the file system to a heterogeneous group of different computers, we installed the “Services for UNIX” component on the cluster and exported the file system both as a CIFS and as an NFS share. We have

previously discussed installation and configuration of SFU in “Microsoft SFU NFS server” on page 30.

Other computers with SAN attachment to this storage system can now share the file system from the MDC with standard SANergy features, as we described in Chapter 2, “SANergy with a Windows MDC” on page 19. Our example configuration demonstrates many of the topics we have already covered. This “super data server” has a large and high-performance shared volume. This volume takes advantage of Windows 2000 Dynamic Disks so that it can be enlarged when needed, on the fly. This data on this volume is being accessed and served in a variety of ways.

In Chapter 1, “Introduction to SANergy” on page 3, we described how SANergy's volume sharing capabilities can benefit Web servers, file servers, and database servers. Our configuration includes all three types of data servers, sharing the same SANergy volume. The servers have workstation clients which they themselves serve the data. While an organization may elect not to have all three types of data servers accessing the same volume, SANergy is fully capable of this, and we have done it here for demonstration purposes.

In the following sections we describe the configuration shown in Figure 132, and explain how the features work together in this environment.

---

## 6.1 High availability

Since the owner of the filesystem in this SANergy environment is the MDC, all the SANergy hosts and their associated workstation clients depend on this machine, to get to their data. This MDC is thus a single point of failure. We recommend that you safeguard the SANergy process by implementing high availability using Microsoft Cluster Server on the MDC as described in 5.1, “MSCS clustering for the MDC” on page 105. We did this on the cluster server musala using the two machines bonnie and clyde as cluster nodes.

Achieving high availability for the SANergy process on the MDC is one concern. But the availability of the physical data on the storage devices is equally as important. Sharing a large pool of storage with SANergy enables you to protect against physical failures much more effectively.

Assume that we have two scenarios, one with SANergy and one without. In the first scenario (Figure 133) we have three computers accessing their storage directly over a SCSI connection. Each single disk has a capacity of 10 MB.

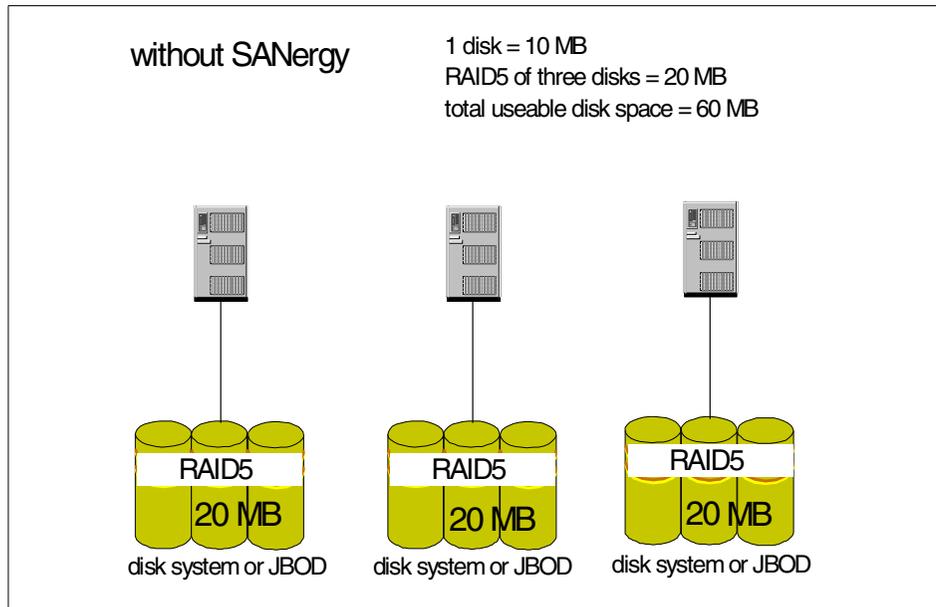


Figure 133. RAID5 configuration without SANergy

For disk failure protection, we could have implemented RAID5 on all three computers (either in hardware on the disk system itself or through software RAID) to prevent data loss in case of a physical disk failure. The available storage capacity for each computer is 20 MB (one disk in a RAID5 configuration is needed for parity data and is not available to applications).

With SANergy we can consolidate the storage and share the large volume among the same computers. We could build a new large RAID5 set with the same nine disks available (Figure 134).

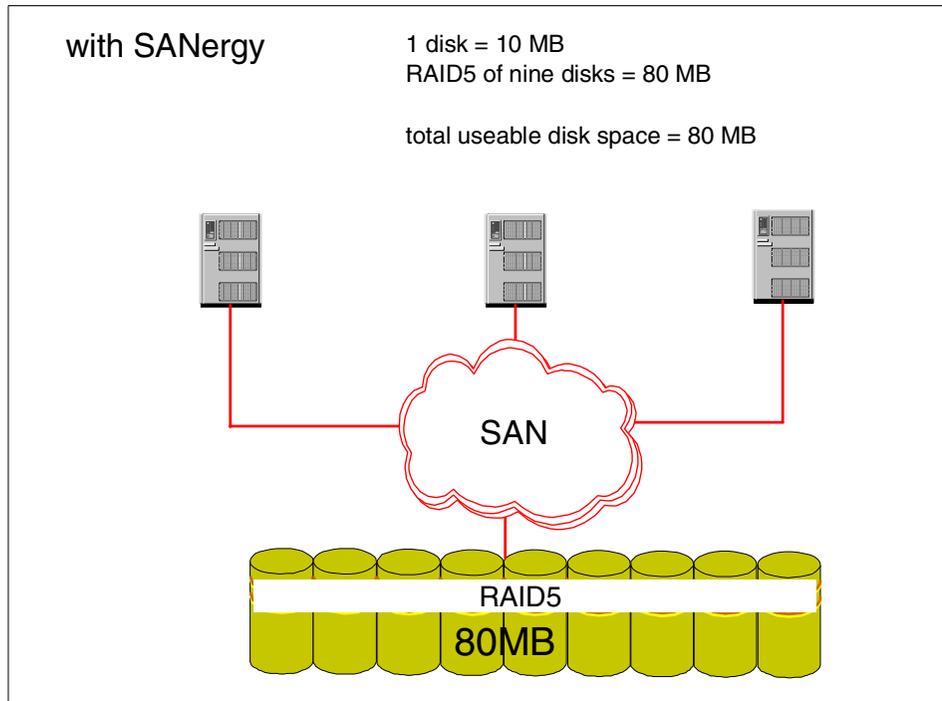


Figure 134. RAID5 with SANergy

The new RAID5 system has a total capacity of 80 MB. This is 20 MB more, compared to the amount of the three systems (3x20 MB) in the previous configuration. The level of protection remains unchanged either way. In both scenarios you can recover from a single disk failure.

## 6.2 High performance

In our test environment we had access to an IBM MSS (Modular Storage Server) SAN storage device. The MSS has two controller units that manage eight disks each. Due to other testing activities we only had seven disks on the first controller and five disks on the second controller available for our use. Depending on your priority (high performance or high availability) you can configure the storage device to your needs (using striping, mirroring, or RAID5) with the hardware specific tools, usually provided by the vendor. A redbook specifically on the MSS is available — *IBM Modular Storage Server An Introduction Guide*, SG24-6103.

Since we wanted to emphasize high data throughput in our implementation, we created two striped volumes, one on each controller (as shown in Figure 135) because striped disk configurations achieve the highest performance per controller (compared to other RAID5 or mirror sets).

All the HBAs on the SAN have visibility to the two LUNs, corresponding to the two striped volumes — one on each controller.

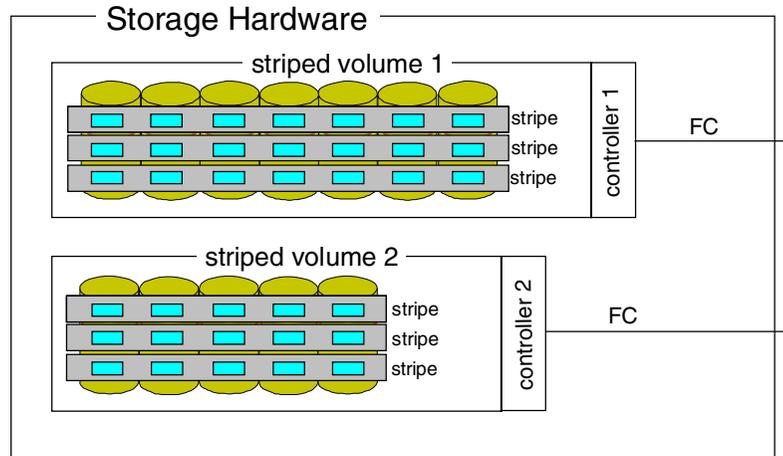


Figure 135. Storage Hardware (striped volumes)

The vendor configuration tools that are shipped with the storage subsystem are normally specific to the particular hardware model. Therefore, in order to further consolidate the storage in terms of creating a large filesystem out of the available disk capacities we used the Windows 2000 Dynamic Disk feature. Dynamic Disk allows us to create a new software stripe out of the two LUNs (Figure 136). The result is that the stripes on each stripe volume are now combined to large stripes within the new volume. This is not actually the way that the Dynamic Disk feature creates the new volume since it has no knowledge about the internal hardware stripes but this description will account for the results we see.

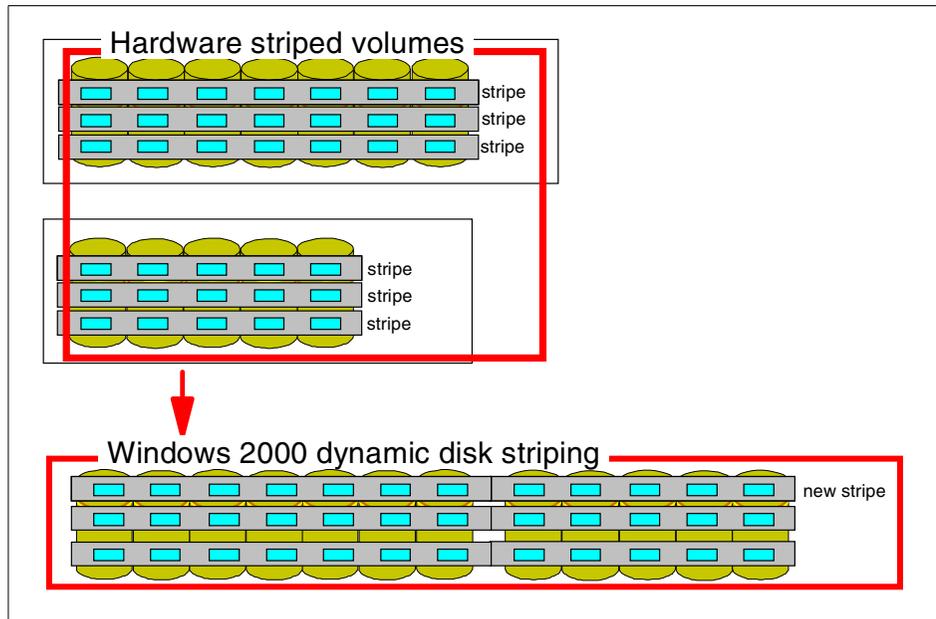


Figure 136. Stripe configuration with Dynamic Disk

The two hardware striped LUNs are now treated as one single volume by the operating system. We formatted an NTFS filesystem on the new volume. This has two benefits for our SANergy implementation.

First, we have a higher storage capacity on our single filesystem for the SANergy MDC. Without the software striping it would have been necessary to create at least two filesystems, one on each hardware stripe volume in order to share all the disks with SANergy.

Second, we improved performance on the new filesystem compared to a filesystem that is stored on one of the hardware stripe volumes. Measurements showed that data throughput on the new filesystem is the summation of the throughput from both hardware stripe volumes. The overall throughput of this configuration depends on the number of hard drives involved in both striped volumes, as we would normally expect.

By setting up in this way, we were able to create a large, high-performing filesystem ready to be shared to the SANergy hosts. Without further performance tuning on the storage hardware (for example, modifying cache settings, chunk size of the disk, and so on) we achieved almost the nominal rated speed of a single SAN connection (100 MB/s) with this configuration on the new filesystem (Figure 137).

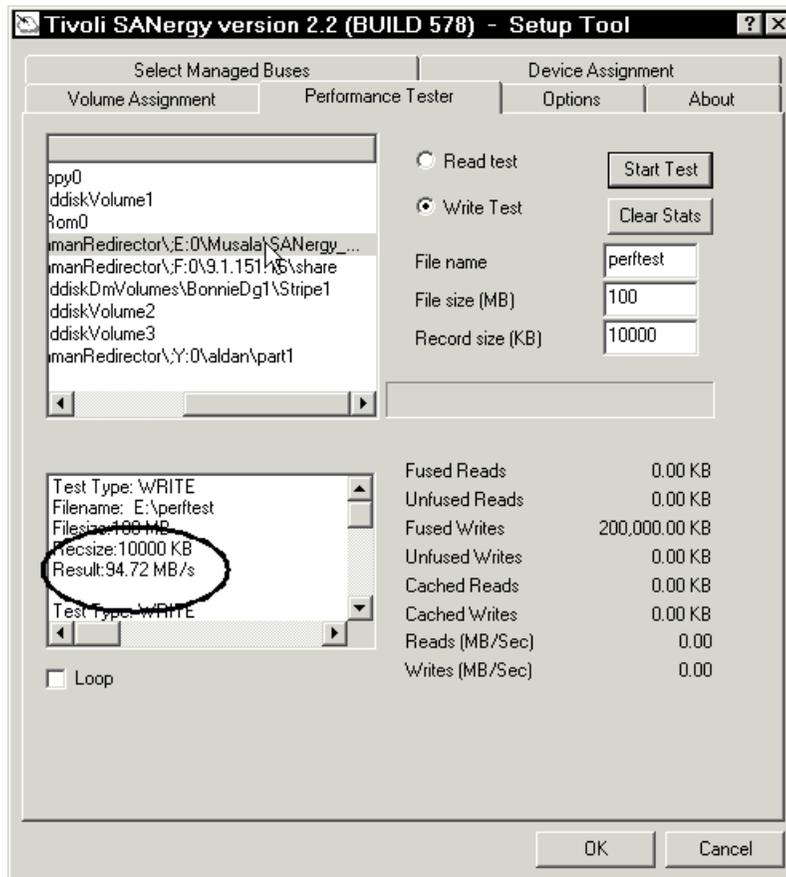


Figure 137. Performance of a large filesystem

### 6.3 Scalable file server farm

In our configuration for this chapter, we have leveraged the benefits offered by SANs and SANergy to create a large, high performance, fault tolerant volume which can be shared by multiple hosts. This volume can be used by file servers as the repository for the data being served.

As we discussed in Chapter 1, “Introduction to SANergy” on page 3, there are some limitations in the way that software is currently exploiting new storage technology. One prime example of this is the limitation of both the common implementations of NFS and the Microsoft implementation of CIFS on *chained exports* or *shares* — that is, they cannot export or share data that itself already resides on a mapped drive. For example, a Windows 2000

server cannot share a drive that it has mapped from another server, via CIFS. NFS, likewise, cannot export a mount point that the host itself has imported from another machine.

The reason that these limitations were implemented is obvious. In a traditional, LAN-based file serving implementation, you would not want an administrator accidentally double-sharing data. Not only would this perform very badly, you would risk a situation where a share would end up referring back to itself through multiple hosts.

However, the new faster storage technology makes this double-sharing a viable and advantageous configuration. For example, having file servers share a SANergy volume or NAS appliance which is then served to their clients has the many benefits we have discussed several times in this Redbook.

So, if most NFS implementations and the Microsoft CIFS implementation do not allow double-sharing, how can we accomplish the benefits of SANergy on file servers? Luckily, Samba can accomplish CIFS sharing, without the built-in limitation on double-sharing.

Samba, as discussed in 3.1.1.2, “Samba CIFS server setup” on page 60 is a powerful and feature-rich implementation of CIFS that can operate on many platforms. For example, many organizations are discovering the benefits of running a UNIX CIFS server. This type of implementation can range from making a very powerful and robust CIFS server running on an AIX host (see also the redbook, *Samba Installation, Configuration, and Sizing Guide*, SG24-6004) to implementing Samba on Linux servers and achieving the benefits of stability and cost-effectiveness.

In our implementation, we utilized Linux to fill the need of CIFS file serving to an imaginary user community. Using such a design, an organization could add as many Linux servers as needed to fill the requirements for file serving horsepower. Our powerful Windows 2000 MDC (running on MSCS) shares its striped volume to the Linux fileserver. Housekeeping tasks, such as backup and recovery, can be performed from the MDC.

The MDC also provides the sharing of the data via NFS. NFS is configured on the MDC using Microsoft's Services for UNIX. The NFS server is controlled by MSCS, so that it is fault tolerant, as well. For more information on using NFS on a Windows server, see 2.1.1.2, “Sharing the storage via NFS” on page 25.

Our Linux hosts mount the NFS share exported from the MDC and share the striped volume at SAN-level speeds via SANergy. The example shows mounting the share H from the MDC musala onto the Linux host rainier.

```
[cbres3@rainier cbres3]$ mount
/dev/hda5 on / type ext2 (rw)
none on /proc type proc (rw)
usbdevfs on /proc/bus/usb type usbdevfs (rw)
/dev/hda1 on /boot type ext2 (rw)
none on /dev/pts type devpts (rw,gid=5,mode=620)
musala:H on /H type nfs (rw,addr=193.1.1.3)
/dev/SANergyCDev/sdc1 on /partition4 type ext2 (rw)
```

Once this share is mounted on rainier, we use Samba to serve the data out to workstation clients using CIFS. We simply define a CIFS share (see Figure 138), as shown before in 3.2.5.1, “Samba usage setup” on page 72. The workstation clients access this share as usual.

The screenshot shows the 'Share Parameters' window for a share named 'H'. At the top, there are buttons for 'Choose Share', 'Delete Share', 'Create Share', 'Commit Changes', 'Reset Values', and 'Advanced View'. The share name 'H' is displayed in a box. Below this, there are sections for 'Base Options' and 'Security Options'. Each option has a 'Help' link, a text input field, and a 'Set Default' button. The 'Base Options' section includes 'comment' (empty), 'path' (set to '/H'), 'guest account' (set to 'nobody'), 'writeable' (set to 'Yes'), 'guest ok' (set to 'Yes'), and 'hosts allow' (set to '9.1. 193.1. cerium').

Figure 138. Re-sharing an imported NFS mount via Samba

Using the combination of NFS on Windows and CIFS sharing from both the MDC and Linux hosts, you can design and implement a flexible, scalable, and universal file-serving infrastructure.

---

## 6.4 Web serving

The Web serving, in this configuration, is done using our Linux hosts running the Apache Web server. This Web server could have as easily been implemented using Windows and Microsoft's Web server, IIS. Likewise, it could have been designed to use enterprise-class UNIX platforms or clusters.

However, using Linux and Apache, in this example, demonstrates how a lower cost implementation can be created without losing any reliability or functionality. When Apache accesses the data on the shared volume, it does so at SAN-level speeds via SANergy.

In today's world, a Web serving infrastructure is a requirement in almost every organization. In addition to the traditional HTML and small graphics files that make up a typical Web page, you may also wish to make large streaming media files available. As the sheer size and complexity of a Web site grows, it becomes critical to minimize the administrative overhead of the site's data, as much as possible.

Many organizations find that a single Web server is insufficient. This may be caused by the Web-serving horsepower of a single server becoming insufficient. Or, the amount of data storage accessible by a single server may be insufficient to house the whole Web site.

As discussed in Chapter 1, "Introduction to SANergy" on page 3, once these situations occur, you have several options, all of which have advantages and disadvantages. By utilizing a shared volume to house the site's data, you can avoid many of the potential problems while still being able to run as many Web server hosts as needed to meet your needs.

Having Apache access the data on the shared volume is no more difficult than accessing it on local storage. Specifically, tell Apache where its site is stored by changing the DocumentRoot parameter in its configuration file.

```
# DocumentRoot: The directory out of which you will serve your
# documents. By default, all requests are taken from this directory, but
# symbolic links and aliases may be used to point to other locations.
#
DocumentRoot "/H/www/html"
```

Once the Web servers are accessing the shared volume (at SAN speeds), you can serve the shared data up to the remainder of the organization (Figure 139).



Figure 139. Web site served by the Linux SANergy host rainier

---

## 6.5 Databases on SANergy hosts

The storage virtualization concepts based on Fibre Channel technology demand new kinds of applications that can exploit with storage when it is not directly attached. Direct attached storage has been standard in the past, and most applications still act as if this is the case (Figure 140).

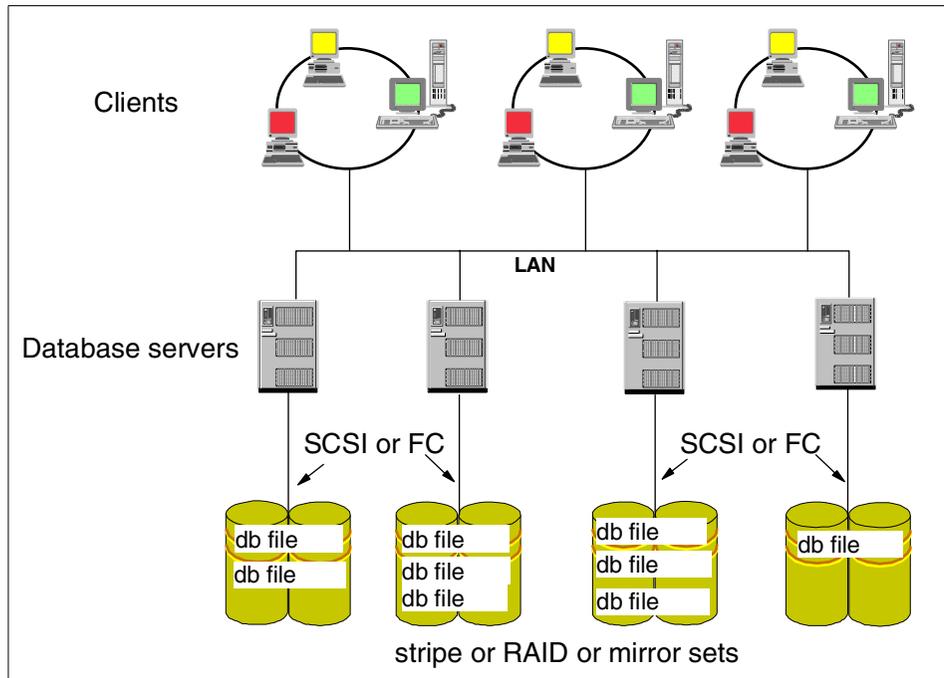


Figure 140. Database servers running in a conventional manner

The change of topology from direct attached (one-to-one) to network attached storage (many-to-many) provided by SANs has the result that applications have to be able to access data on shared storage. Even technologies that are not based on Fibre Channel, such as Network Attached Storage (NAS) systems, ask for this new capability in their applications.

NAS devices are accessible over a LAN path and provide disk storage capacity to computers via file sharing. SANergy, as we know, also provides access to disk and data via file sharing over the SAN path. Applications that want to use this storage must be able to run with data on mapped network drives (Figure 141).

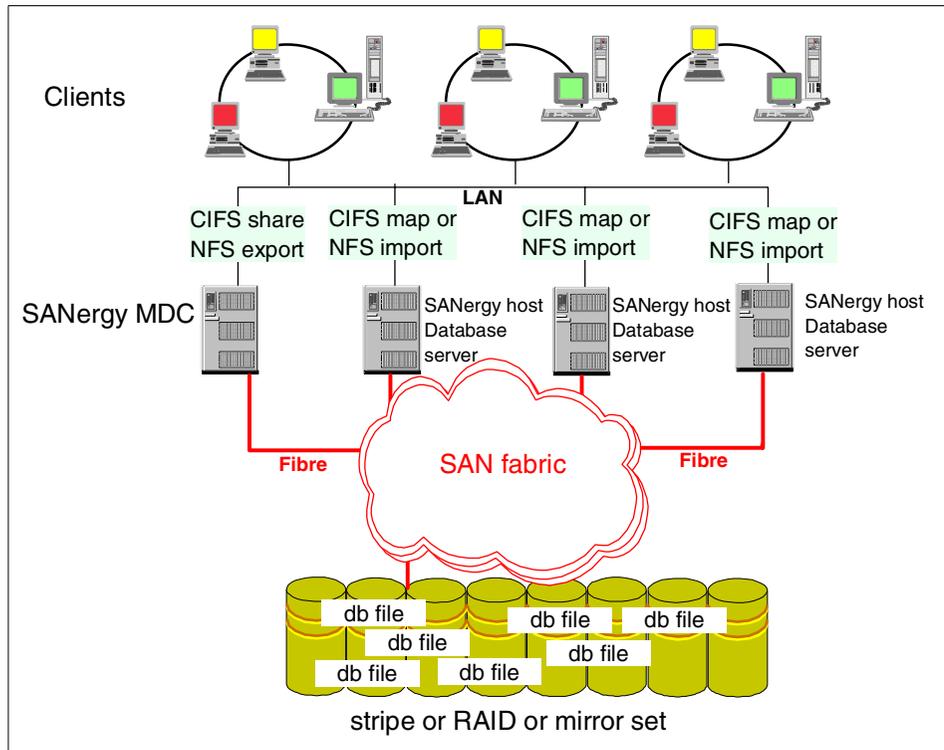


Figure 141. Database servers in a SANergy environment

SANergy delivers the storage capacity of a SAN volume to hosts over network shares just as NAS does. In either case, the applications see the same storage, but with SANergy, the real file I/O traffic goes over the SAN connections. The requirements that have to be met of the applications are similar in both worlds, NAS and SAN.

At the present time, not all applications have this functionality — many of them restrict the use of network drives. But the increasing popularity of SAN and NAS technologies will likely encourage most applications to be able to run on and exploit shared volumes in the future.

There are certainly many possibilities today to run applications with data on network drives. As an example, we have installed Microsoft SQL Server v7.0 on a SANergy host. The installation of the SQL Server software was done on the local disk. The aim is then to create databases on shared volumes. Installing SQL Server was straightforward — more information is available in the vendor's documentation.

We assume a basic familiarity with SQL Server administration for the rest of this section. The procedure we will focus on in this section is the configuration of SQL server to run databases on the shared volume on the SAN. We also assume that SANergy is properly installed on your SANergy MDC and on the host and that you have fused access to the shared volume from your host. For further information on installing SANergy, please see Chapter 2, “SANergy with a Windows MDC” on page 19 and Chapter 3, “SANergy with a UNIX MDC” on page 57.

### 6.5.1 Creating a database on a mapped network drive

The default SQL installation process creates six databases: master, model, msdb, pubs, Northwind, and tempdb. Four of these databases are system databases (master, model, msdb, tempdb). These six databases are automatically installed on a local disk.

In order to create a new database on a shared network drive, we need to have the DBCC trace flag **1807** set. This flag allows any command issued to the SQL server to contain UNC values. To activate this trace, go to **Start -> Programs -> Microsoft SQL Server -> Enterprise Manager** to start the manager console. In the tree view of the console, go to the name of the current machine, in our case, diomedes (Figure 142).

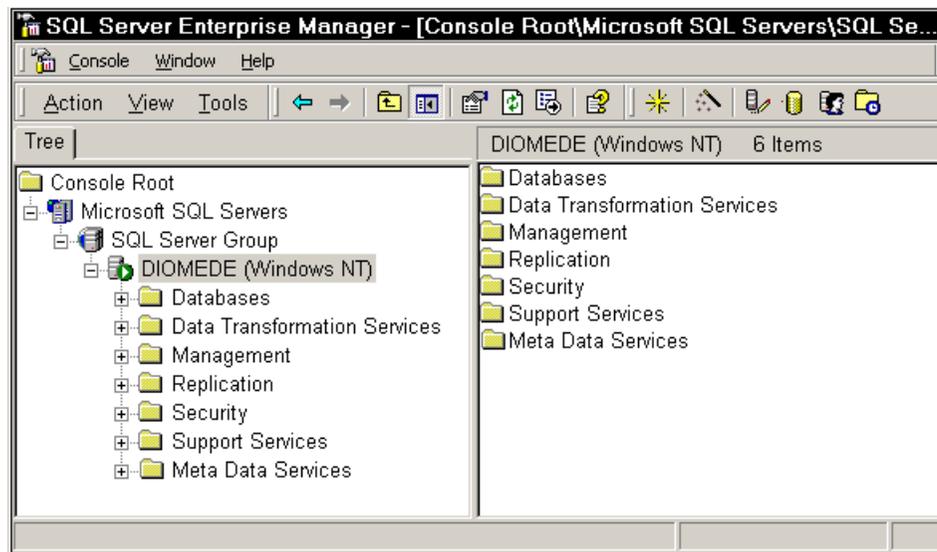


Figure 142. SQL Server Enterprise Manager

Right-click the server and select **Properties** to get to the **SQL Server Properties** configuration screen (Figure 143).

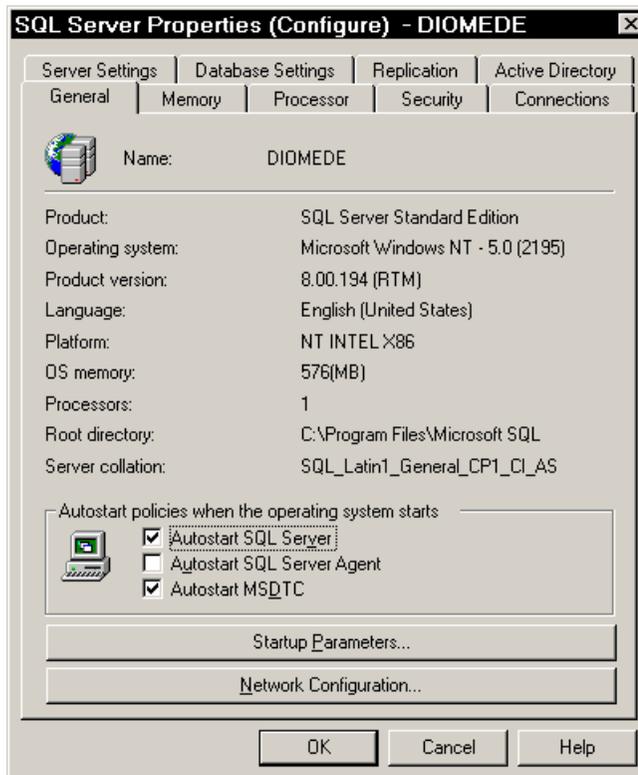


Figure 143. SQL Server Properties

Select the **General** tab on the top of the screen and then click the **Startup Parameters** button. The following window will appear (Figure 144).

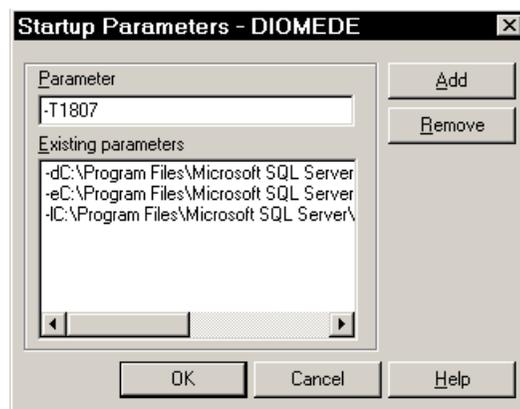


Figure 144. Startup parameters

In the **Parameter** field you can now specify the trace flag. Type in the field the value `-T1807` and click the **Add** button.

You have to restart the SQL server service in order to activate the trace flag before attempting to proceed with the database creation. We used a standard SQL script shipped with the SQL server to create a database. We adapted the script to create the sample database SANERGY\_DB1 and executed it in the **SQL Query Analyzer** tool. To do this, we selected **Tools** from the SQL Enterprise Manager screen in Figure 142 on page 183 and chose **SQL Query Analyzer**. Next we opened the file that contains this script and executed the query with F5.

```
USE master
GO
CREATE DATABASE SANergy_DB1
ON PRIMARY
(NAME = SANergy_DB1,
FILENAME = '\\musala\SANergy_Share1\sqldb\SANergy_DB1.mdf',
SIZE = 200MB,
MAXSIZE = 5000MB,
FILEGROWTH = 10)
LOG ON
(NAME = SANergy_log1,
FILENAME = '\\musala\SANergy_Share1\sqldb\SANergy_DB1.ldf',
SIZE = 10MB,
MAXSIZE = 20MB,
FILEGROWTH = 20)
GO
```

The filename for the database files contains the UNC name `\\musala\SANergy_Share1` of the mapped drive which has fused access over the SAN and is owned by the clustered MDC musala. For creating the tablespaces and other objects in the database, we again used the standard scripts provided with the SQL server software. You do not need to change anything specific in these scripts to use SANergy.

When looking at the directory of databases on the SQL Server Manager you can verify that the database was created in the specified directory (Figure 145) by right-clicking on the database icon in the tree view and selecting **Properties**. A window named **SANergy\_Share1 Properties** appears where you can select the **Data Files** tab to verify the created files.

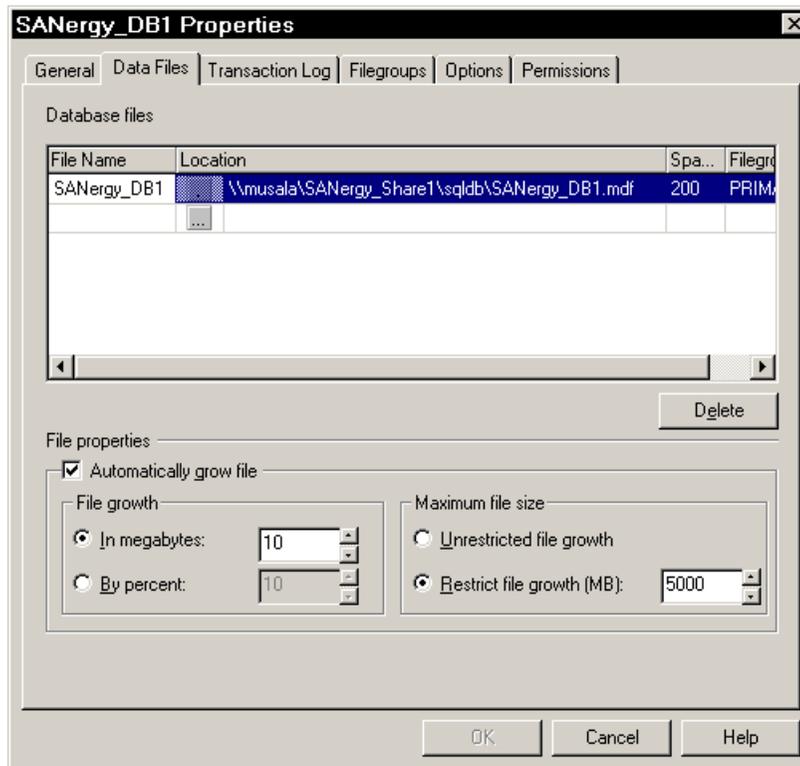


Figure 145. Properties of created database

## 6.5.2 Migrating an existing database from local to shared disk

The master database as created by the installation process must always reside on a local disk. For the other standard databases created during the install, we have not found any particular restrictions, but it is probably best practice to confine these also onto local disk. Note that by *local disk*, we mean disk which is formatted on this server and appears on a local drive, as opposed to a network shared disk. Local disk, of course, could still be SAN attached, as opposed to disk physically installed inside the database server.

We want to show a way to migrate databases from a local disk to a shared volume managed by SANergy. Therefore, we first created a database called **SANergy\_DB10** on local disk. To move the database files to the mapped share, we detached the database from the server. You have to make sure that no user is connected to the database before doing this. Issue the `checkpoint` command to the query analyzer to ensure consistency of all data at this time.

We detached the database by executing the command:

```
sp_detach_db 'SANergy_DB10'
```

The command completed successfully. Next we copied all the database files (.mdf, .ndf, .ldf) that belong to the database **SANergy\_DB10** to the directory **sqldb** on the shared volume — in this case, that is, the data file SANergy\_DB10.mdf and the log file SANergy\_DB10.ldf. Now we could re-attach the files to the database by issuing the command `sp_attach_db` with the appropriate parameters.

```
sp_attach_db 'SANergy_DB10', '\\musala\SANergy_Share1\sqldb\SANergy_DB10.mdf',  
'\\musala\SANergy_Share1\sqldb\SANergy_DB10.ldf'.
```

The database can now be accessed from the shared volume, and the old database files on the local disk can be removed.

---

## 6.6 Conclusion

The intent of this chapter was to describe a configuration that demonstrated many of the benefits of a “super data server” configuration with SANergy. As you can see, SANergy enables you to combine many products that are very beneficial, on their own, such as Microsoft Cluster Server, NFS, CIFS (Microsoft and Samba), a Web server (Apache), an RDBMS (Microsoft SQL Server, in our example) — with the advantages of SAN-based data storage.

Most organizations will implement many if not all of these types of applications. When they do so using SANergy, they will achieve the benefits of shared access to large, SAN-based data storage with high performance and fault tolerance. Using SANergy, therefore, increases the performance of these applications and greatly reduces the administrative overhead.



---

## Part 3. SANergy and other Tivoli applications



---

## Chapter 7. SANergy and Tivoli Storage Manager

This chapter describes how to use SANergy with Tivoli Storage Manager. Tivoli Storage Manager is Tivoli's solution for enterprise-wide data protection, and is described in the redbook *Tivoli Storage Management Concepts*, SG24-4877. Tivoli Storage Manager operates in a client-server architecture where a server system (with various storage media attached) provides backup services to the clients.

The trivial case of using Tivoli Storage Manager with SANergy is where the Tivoli Storage Manager client, resides on the MDC and performs normal Backup/Archive client operations for the MDC data. Setup and considerations here are the same as for a Tivoli Storage Manager client without SANergy. Beyond this, various scenarios are presented in the following sections, including application server-free backup and restore, implementing Tivoli Storage Manager backup sets on SANergy volumes, backing up a commercial database from a SANergy host, and using Tivoli Storage Manager where there are different operating systems for the MDC and SANergy host.

In this chapter, unless otherwise indicated, *client* refers to the Tivoli Storage Manager backup/archive client, and *server* refers to the Tivoli Storage Manager server.

---

### 7.1 Application server-free backup and restore

The purpose of this scenario is to show how to successfully perform LAN-free backup and restore operations where the Tivoli Storage Manager client does not run on the system which owns the data, that is, it is not installed on the SANergy MDC, but rather on one or more of the SANergy hosts. This configuration is sometimes referred to as *application server-free*, since there is no Tivoli Storage Manager code installed and no backup processing takes place on the MDC/application server itself. The operating system or platform of the SANergy host is the same as that of the SANergy MDC. The reason we chose to run the client on a SANergy host whose platform is the same as that of the SANergy MDC is explained in section 7.4, "Heterogeneous SANergy backup configurations" on page 225.

Our environment is the Tivoli Storage Manager Version 4.1 server and client on Windows 2000 system jamaica. Note that jamaica is also a SANergy host. The SANergy MDC is aldan, and it is an application server. Figure 146 illustrates this configuration. The data owned by MDC aldan is on a partition of the Fibre Channel disk storage device.

SANergy for this configuration is set up as described in 2.3.1, “Installing and configuring SANergy hosts on Windows” on page 45, Windows SANergy MDC and Windows and UNIX hosts. The other systems shown are SANergy hosts and application servers not directly involved in this setup. The client on jamaica is sending data through shared memory to the server.

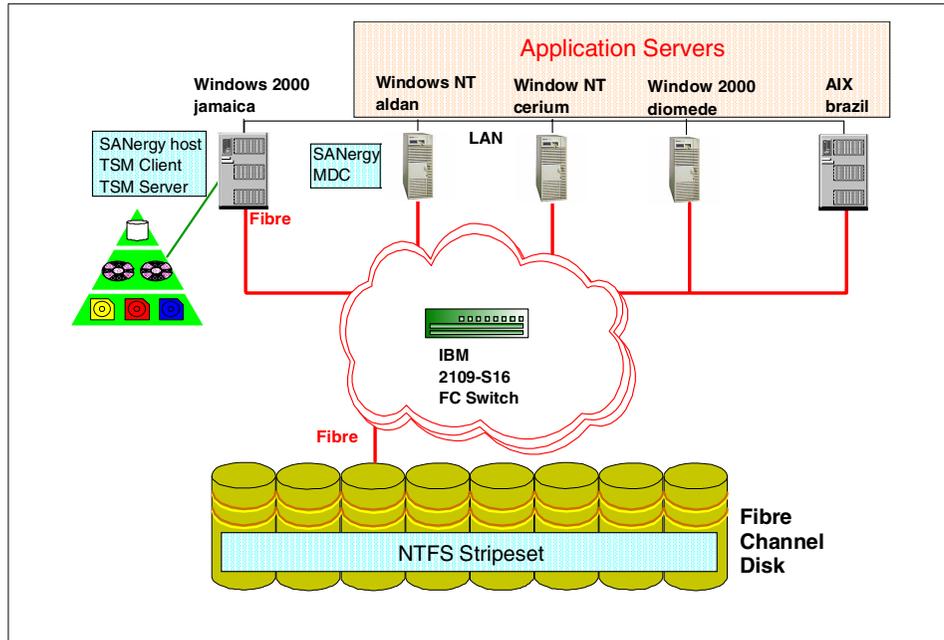


Figure 146. Application server-free backup configuration

## 7.1.1 Client implementation and testing without SANergy

Tivoli Storage Manager and SANergy should be implemented and tested independently of each other prior to using them together. We implemented the Tivoli Storage Manager server and client on jamaica before installing SANergy, but the order is not significant. This section covers Tivoli Storage Manager setup considerations and testing without SANergy.

### 7.1.1.1 Implementation

The Tivoli Storage Manager server and client should be installed, set up, and tested. When running the client on the same Windows system as the server, the recommended communications method is *namedpipe* for the best performance. The relevant statements from our `dsm.opt` are shown below. The `namedpipename` value must match with the same option in the server's `dsmserv.opt` file.

```
*=====
* named pipes
*=====

commethod namedpipe
namedpipename \\.\pipe\Server1
```

Additional information on the implementation of a Tivoli Storage Manager server and client can be found in the redbook *Getting Started with Tivoli Storage Manager: Implementation Guide*, SG24-5416.

There are no special considerations for the server to support a SANergy host as a client. Regarding performance, the speed of the devices used for the storage pool of the client's data, whether disk or tape, will definitely affect backup and restore performance.

For example, if the storage pool used consists of SCSI disk volumes that have a maximum throughput rate of 5 MB/sec, the Tivoli Storage Manager throughput rates will be limited by that limit, even if the SANergy data being backed up or restored is on a SAN attached disk subsystem that can handle much higher data rates. For our testing, we used a disk storage pool with a volume on a SAN attached disk in our Fibre Channel disk subsystem (an IBM MSS) which was not managed by SANergy.

### 7.1.1.2 Testing

Testing can be done first with data on a local drive that is not SANergy managed to prove that there is basic connectivity and functionality between the Tivoli Storage Manager server and client. We do not show this simple test here. Then the Tivoli Storage Manager backup or archive functions should be tested using a network mapped drive before it is fused using SANergy (in our case, using a simple CIFS mapped volume from the server aldan). We backed up the folder TSMtest which contained 10 files and approximately 200 MB of data. The Tivoli Storage Manager GUI immediately before starting the backup is shown in Figure 147.

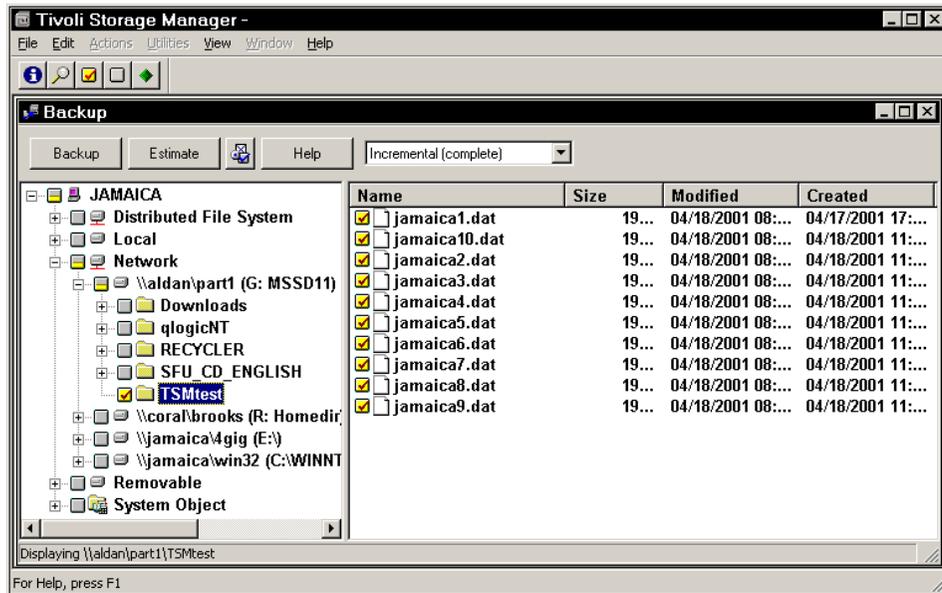


Figure 147. Tivoli Storage Manager client GUI for backup

Please note that mapped drives, which is how Windows applications view SANergy volumes, will show up under the **Network** portion of the GUI file structure tree on the left side of the panel.

We selected the folder to be backed up on the left side of the display; this selects all of the files within that folder. The backup operation was then initiated. Because of the policy settings on the server, the backup data was sent to a disk storage pool residing on a SAN attached, but not SANergy managed, disk on our server jamaica.

The data flows from the SAN attached disk over the Fibre Channel through the CIFS server, aldan, then over the LAN to the Tivoli Storage Manager client, jamaica, which has mapped the drive. The data transfers through shared memory between the Tivoli Storage Manager client and server (both physically located on jamaica), and finally to the Tivoli Storage Manager server disk storage pool volume. That volume is itself SAN attached, but is shown as part of the server storage hierarchy icon. The data flow from aldan to jamaica is over the LAN because we are not using SANergy in this test. This is illustrated with the dark arrows in Figure 148.

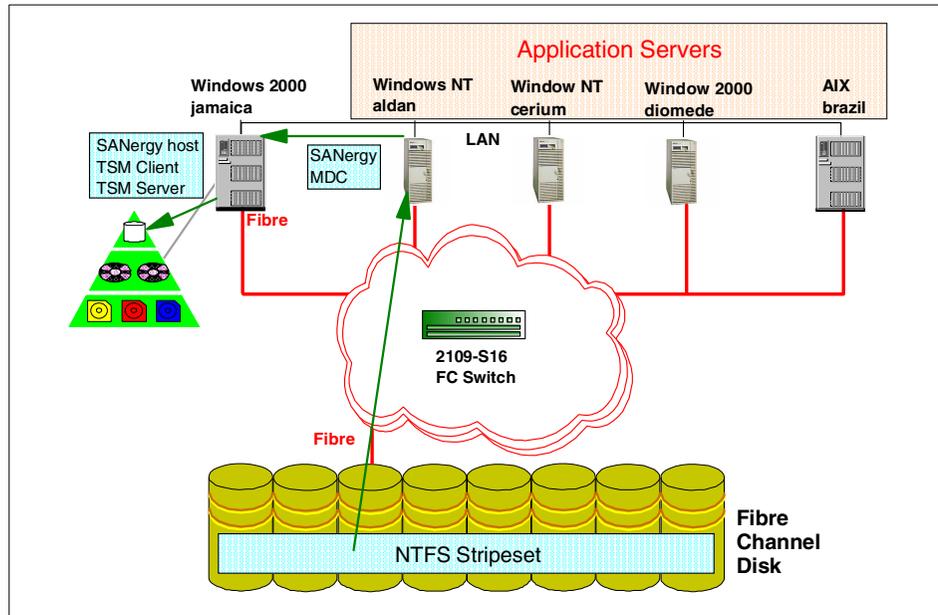


Figure 148. Backup data flow without SANergy

The results of the backup operation are shown in Figure 149. The throughput of the backup operation was 1.5 MB/sec, as shown by the Aggregate Transfer Rate on the screen in the figure. This will be used as the baseline for the backup performance test with SANergy later.

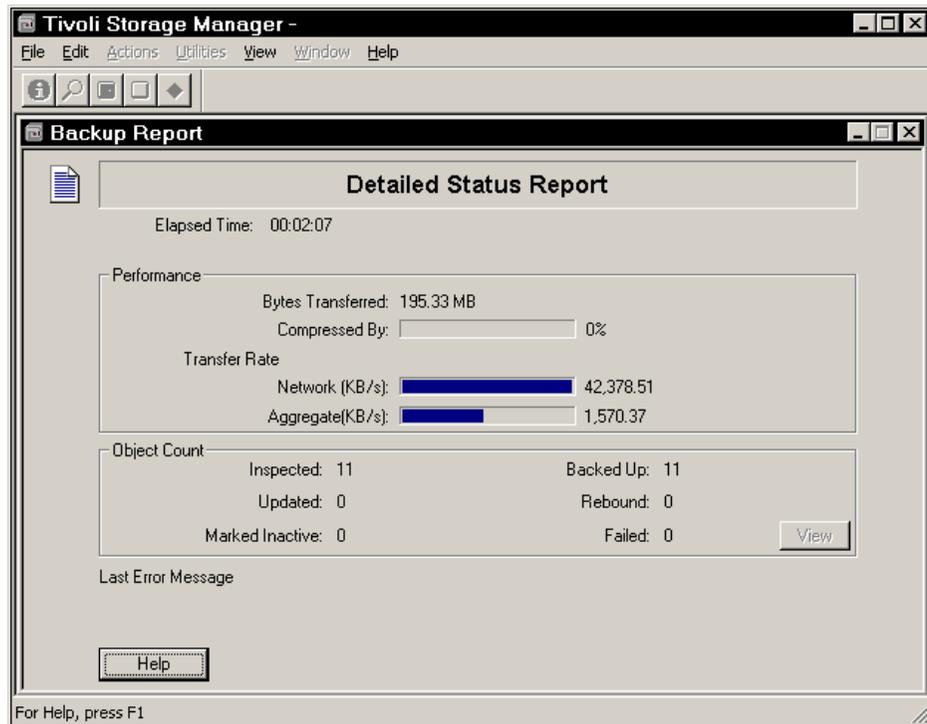


Figure 149. Backup results without SANergy

Restore testing is also highly recommended. The purpose of our restore test was twofold:

1. Verify that when backing up and restoring mapped drive data, the client properly restores not just the directory and files, but also the permission data for both.
2. Establish a baseline for our simple performance test with SANergy.

Before running the restore test, we checked the Windows security properties, that is, the permissions, for the directory and one of the files. The Windows properties of a directory or file can be displayed by clicking the right mouse button on the object whose properties are desired, then selecting **Properties**, then clicking the **Security** tab. The permissions prior to the restore are shown in Figure 150 and Figure 151.

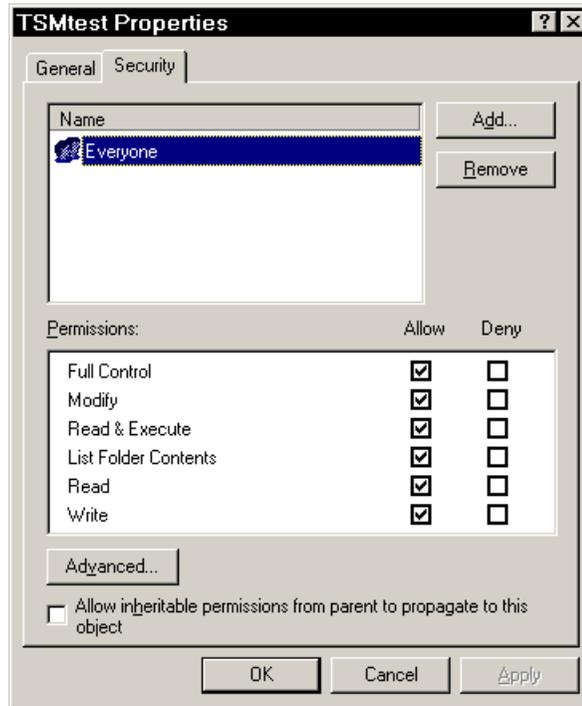


Figure 150. TSMtest directory properties

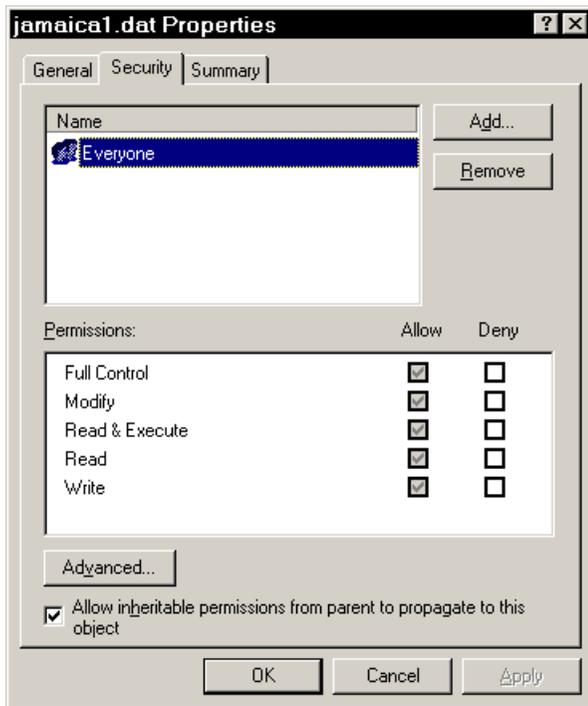


Figure 151. TSMtest file properties

We then deleted the folder TSMtest and then used the client GUI to run the restore. This is shown in Figure 152.



Figure 152. Client GUI for restore

The results of the restore operation are shown in Figure 153. The directory and files were indeed restored, and the resulting Windows permissions for both were the same as before the directory was deleted. The throughput of the restore was 1.1 MB/sec, as shown by the Aggregate Transfer Rate on the screen in the figure. Again, this will be used as the baseline for the restore performance test with SANergy later.

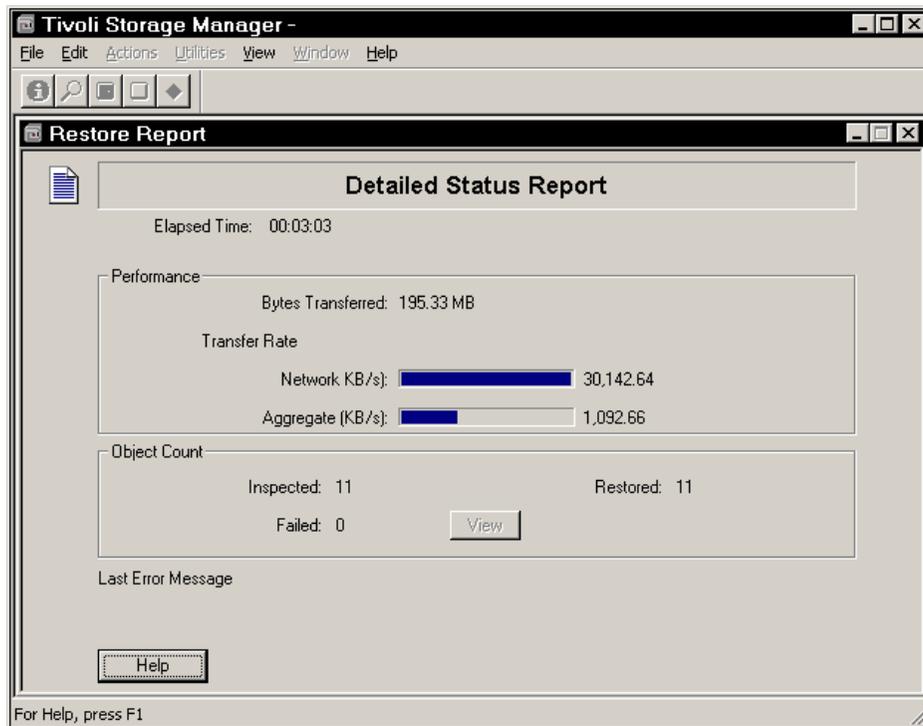


Figure 153. Restore results without SANergy

## 7.1.2 Tivoli Storage Manager client with SANergy

In this section we describe implementation and testing of Tivoli Storage Manager on a SANergy host.

### 7.1.2.1 Implementation

The system performing the backups needs to be set up as a SANergy host. SANergy should be implemented as described in Chapter 2, "SANergy with a Windows MDC" on page 19. The SANergy Setup Tool should be used to fuse to the required volume and to verify that I/O to the volume is indeed handled by SANergy.

The SANergy setup screen for jamaica is shown in Figure 154 with the SANergy disk which we used for Tivoli Storage Manager testing selected, **part1** on aldan. SANergy performance tests verified that our reads and writes to the drive were fused; this is shown by the **Fused Reads** and **Fused Writes** statistics on the screen.

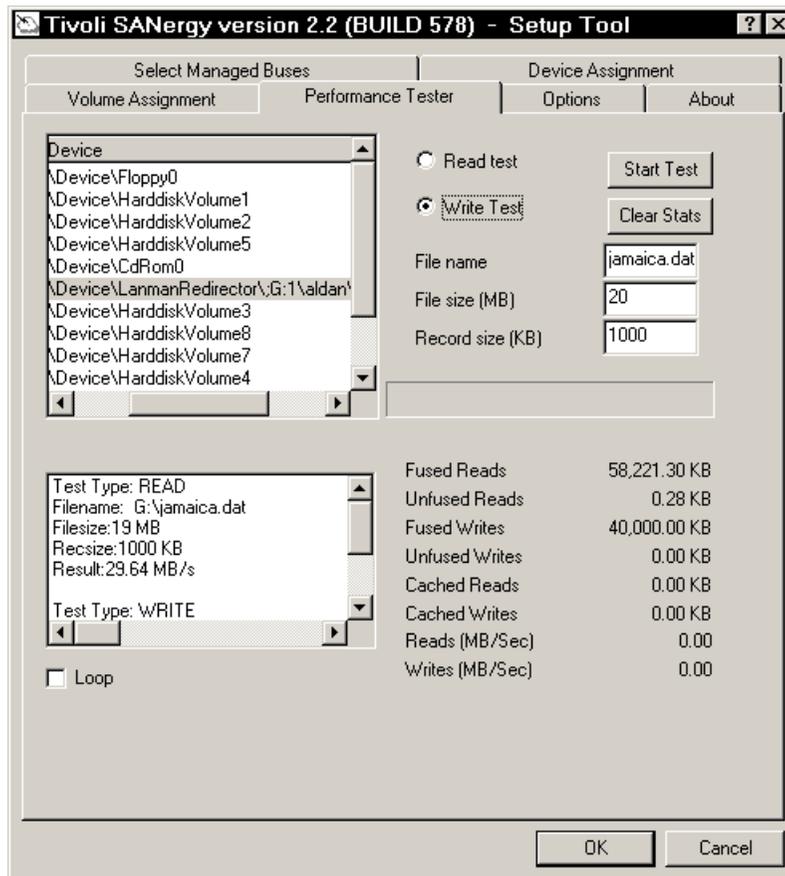


Figure 154. SANergy setup on the client

### 7.1.2.2 Testing

The use of the Tivoli Storage Manager client with SANergy is functionally the same as without SANergy. The client is not aware that the path to its data is different than without SANergy (which is exactly how SANergy is designed to work; that is, to be transparent to applications). The observed differences are:

- Now the data does not flow through aldan, the CIFS server, and MDC.
- The data does not flow over the LAN, thus reducing the load on the LAN.
- The performance of backup and restore operations has improved.

The data flow with SANergy is illustrated with the dark arrows in Figure 155. The server storage pool configuration is the same as before.

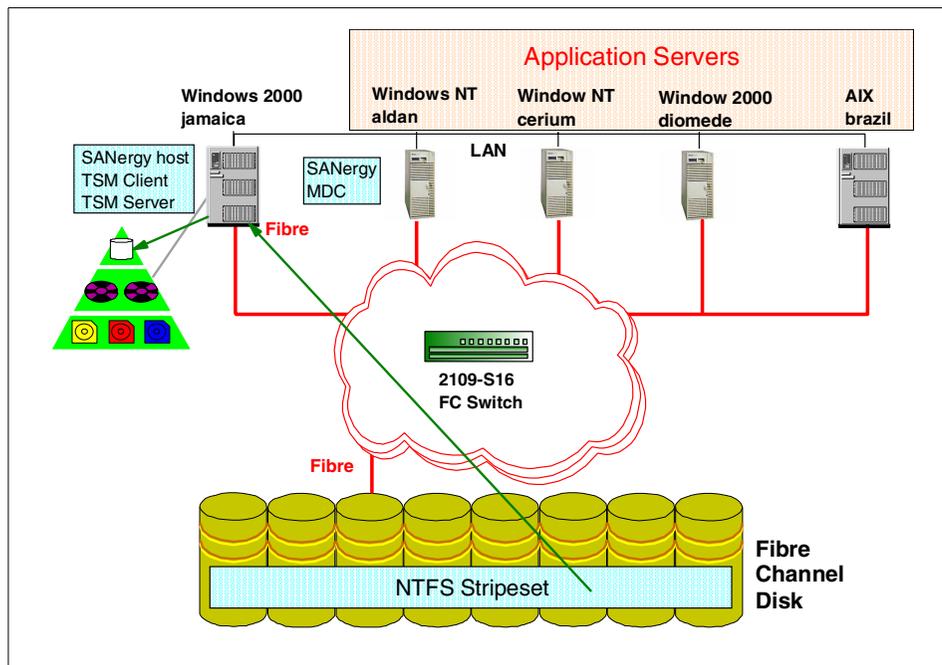


Figure 155. Backup data flow with SANergy

We ran a backup of the same directory and files we did before, described in 7.1.1, “Client implementation and testing without SANergy” on page 193. The client GUI displays are the same so they will not be given again. The Detail Status Report from the backup test is shown in Figure 156. The report shows a throughput of over 6.9 MB/sec versus 1.5 MB/sec without SANergy.

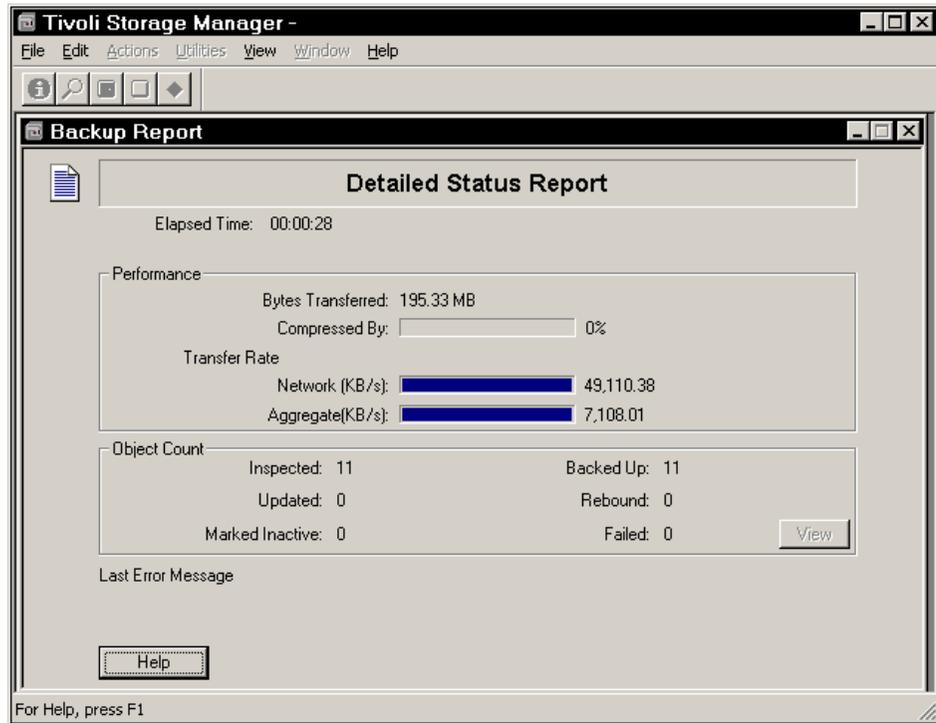


Figure 156. Backup results with SANergy

Next we deleted the directory and performed the restore test again. The results are shown in Figure 157. The Aggregate Transfer Rate shows a restore rate of 4.1 MB/sec compared to 1.1 MB/sec when we restored without SANergy.

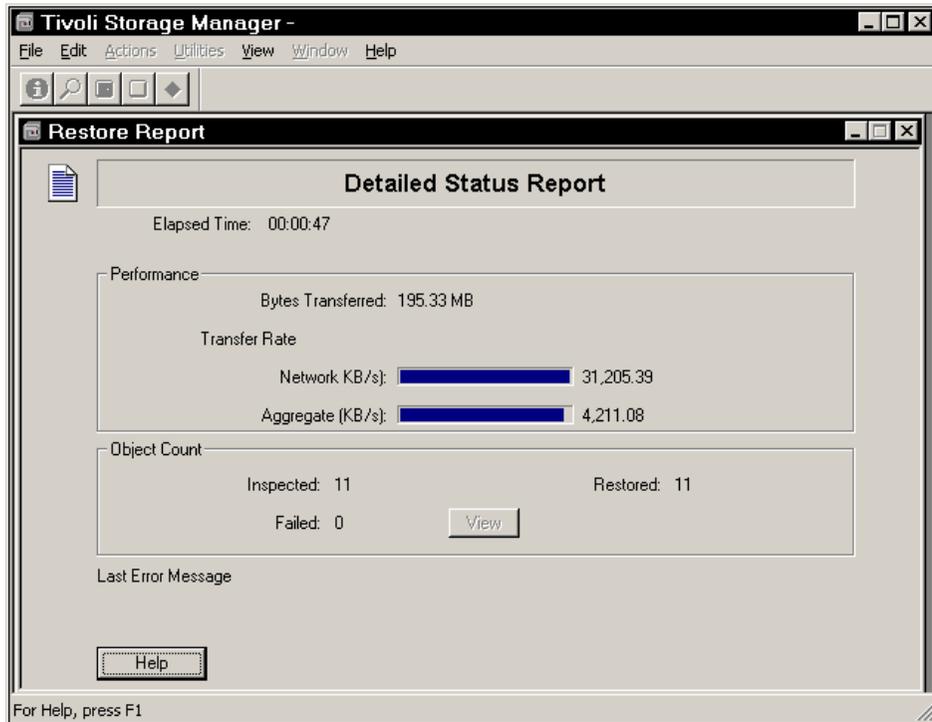


Figure 157. Restore results with SANergy

We also checked the Windows permissions of the directory and a file. They were the same as before.

This testing demonstrates successful application server-free backup and restore, in other words, backup and restore that does not put a resource load on applications servers that are using SANergy managed data. Along with that, it also shows how to perform backup and restore without sending the data over the LAN. Finally, it shows a potential for significant performance improvements using SANergy versus backing up and restoring the same data with a network mapped client.

Keep in mind that both Tivoli Storage Manager and SANergy perform better with large files and that we were using files of 20 MB each for our test. Also, the performance figures we obtained were specific to our environment, and should be used to gain an idea of likely relative improvements only.

---

## 7.2 SANergy managed backup sets

In this section we will see how SANergy can be used to store Tivoli Storage Manager *backup sets* and make them accessible, with high performance, to the clients that need to restore from them.

First, what is a Tivoli Storage Manager backup set and how is it used? A backup set is a collection of backed up files from one client, stored and managed as a single object in the server's managed storage. The most recent version of a client node's active, backed up files are consolidated onto media that may be directly readable by the client. In our example, the media will be a disk volume that is SAN attached to both the Tivoli Storage Manager server and clients and is managed by SANergy.

Backup sets are an efficient way to create long-term storage of periodic backups, without requiring the data to be sent over the network again. Client nodes can restore their backup sets in either of two ways:

- Directly from the server
- Using a device, for example disk or tape, attached to the client's machine that will read the media on which the backup set is stored.

More detailed information on backup sets can be found in these books:

- *Tivoli Storage Manager for Windows Administrator's Guide*, GC35-0410
- *Tivoli Storage Manager for Windows Administrator's Reference*, GC35-0411
- *Tivoli Storage Manager for Windows Using the Backup-Archive Client*, SG26-4117
- *Tivoli Storage Manager Version 3.7: Technical Guide*, SG24-5477

We will generate backup sets for two clients using our Tivoli Storage Manager server jamaica. The backup sets will be stored by jamaica on a SANergy managed volume accessible to the clients cerium and sol-e. The clients and server are all SANergy hosts of the same MDC, aldan. The illustration in Figure 158 shows our configuration for this scenario.

The benefits of this type of combined use of Tivoli Storage Manager and SANergy include an efficient use of common disk space as well as the ability to achieve SAN access speeds when clients restore from backup sets without needing to access the Tivoli Storage Manager server.

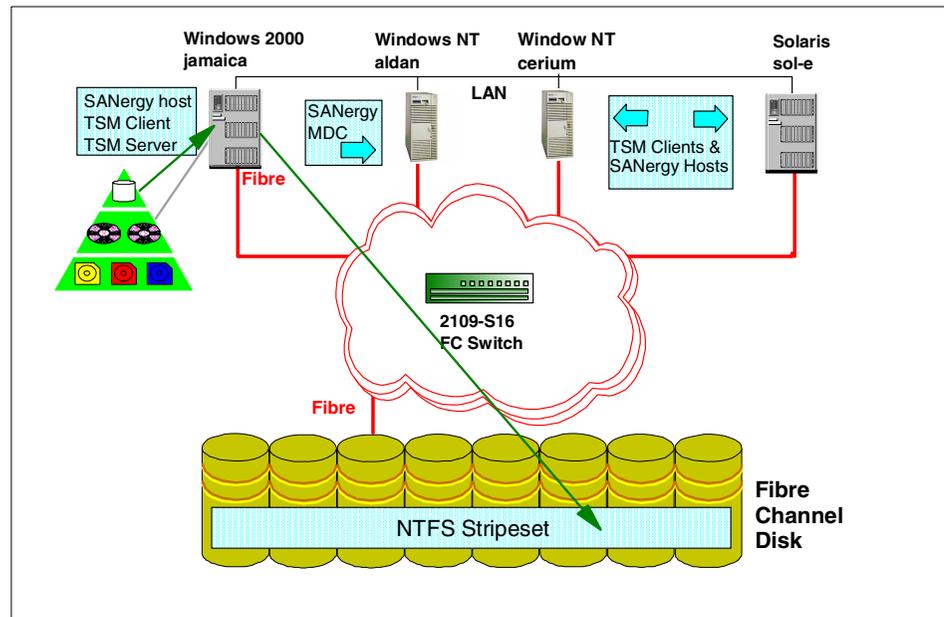


Figure 158. Backup set configuration

### 7.2.1 Tivoli Storage Manager server setup

The Tivoli Storage Manager server should already be implemented and operational for client backup and restore. In our environment, the server was at Version 4.1.3.0 and running on jamaica. To create backup sets as files on a disk volume, a device class with device type file is used. A more detailed description of this is given below. The volume which we used as the target for our backup sets was mapped as drive G: on jamaica and was fused through SANergy. A Windows Explorer view of our mapped drive can be seen in Figure 159.

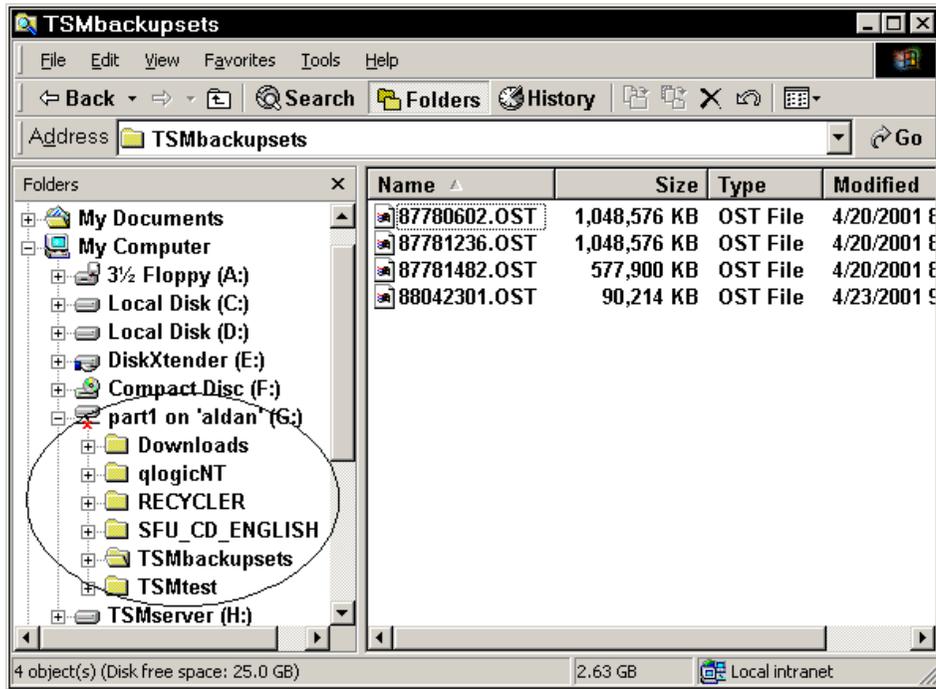


Figure 159. Tivoli Storage Manager server mapped drive

### ***Tivoli Storage Manager on Windows — considerations***

It is important to keep in mind that the disk volume which will be used to write backup sets must be mapped and fused so that it is accessible by the Windows Tivoli Storage Manager server service and available before any server activity that needs to access the volume. This is not normally the case, as the Tivoli Storage Manager server starts at boot time, and network drives are mapped only when a user logs in.

We overcame these limitations by combining a user exit with a scheduled backup/archive client operation to make sure that the required drive would be mapped and available before Tivoli Storage Manager required it. Details of the user exit and our procedure can be found in Appendix B, “Windows server drive mapping and SANergy” on page 271. A future release of Tivoli Storage Manager may provide a facility eliminating the need for this procedure. Tivoli Storage Manager servers on UNIX will not have this problem, because filesystems are mounted before starting the Tivoli Storage Manager process.

## 7.2.2 Defining the backup set device class

A device class using the file device class must be defined, and its directory parameter must point to the desired directory on the SANergy managed volume. The root directory of the drive can be used. Our device class definition is shown in the following screen.

```
Session established with server JAMAICA_SERVER1: Windows NT
  Server Version 4, Release 1, Level 3.0
  Server date/time: 04/23/2001 14:37:02  Last access: 04/23/2001 13:07:09

tsm: JAMAICA_SERVER1>define devclass sanbackupset devtype=file maxcap=1g
dir="g:\TSMbackupsets"
ANR2203I Device class SANBACKUPSET defined.

tsm: JAMAICA_SERVER1>
```

The server is now ready to generate backup sets.

## 7.2.3 Backup set generation

Any UNIX filesystem or Windows drive for which a backup set is to be generated must have been previously backed up completely at least once with the incremental backup function. In the two scenarios below, one for Windows and the other for UNIX, we first run a full incremental backup of the drive or filesystem, then we generate a backup set.

### 7.2.3.1 Windows backup set generation

The filespace we used for our Windows example was \\cerium\e\$, the E drive on system cerium. The GUI screen to run the incremental backup of this drive is shown in Figure 160.

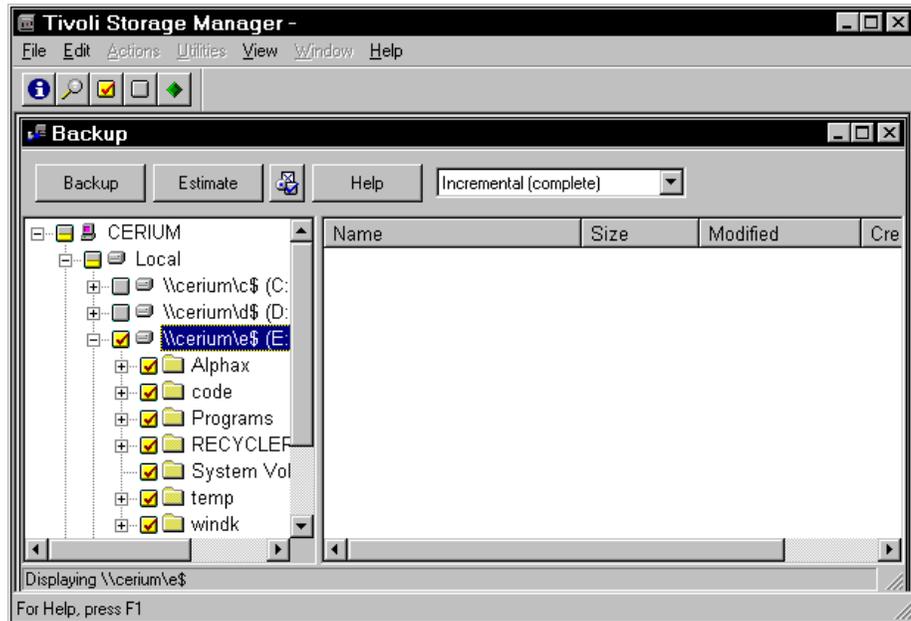


Figure 160. Windows GUI for backup set generation

The results of the incremental backup are shown in Figure 161.

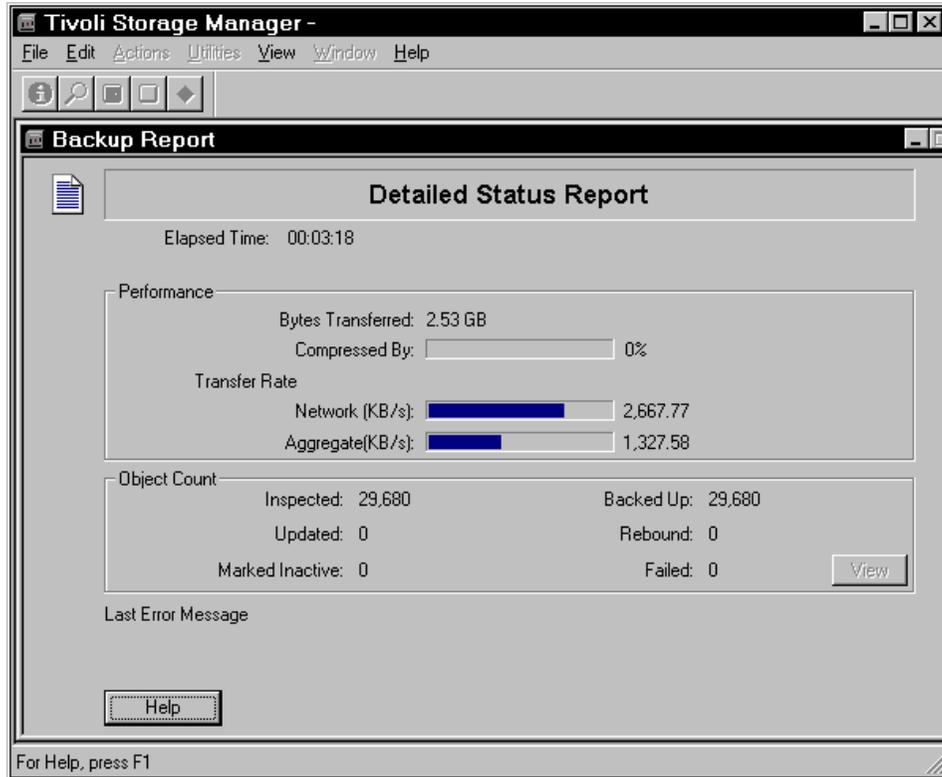


Figure 161. Windows file system backup results

We now had the data from which to generate a backup set for cerium. It should be noted that this backup occurred over the LAN — that is, the data flowed from cerium’s local E drive, over the network to the server jamaica, and was then written to its storage pool.

After the successful backup, we initiated a backup set of \\cerium\e\$ from the server, specifying to use our newly defined device class, *sanbackupset*. The `generate backupset` and `query backupset` commands and their output are shown here:

```
tsm: JAMAICA_SERVER1>generate backupset cerium edrive \\cerium\e$ ret=5
desc="Cerium E$ backupset" devc=sanbackupset
ANS8003I Process number 6 started.

tsm: JAMAICA_SERVER1>q backupset cerium
Session established with server JAMAICA_SERVER1: Windows NT
Server Version 4, Release 1, Level 3.0
Server date/time: 04/20/2001 09:05:36 Last access: 04/20/2001 09:02:47

Node Name: CERIUM
Backup Set Name: EDRIVE.86021
Date/Time: 04/20/2001 08:30:02
Retention Period: 5
Device Class Name: SANBACKUPSET
Description: Cerium E$ backupset
```

We also wanted to verify that the server I/O for writing the backup set was fused, that is handled by SANergy, so we took a snapshot of the SANergy Setup Tool/Performance Tester screen *before* starting the backup set generation and *after* it was completed. The tool keeps track of fused I/O even if it is not initiated by the performance tester itself.



The statistics *after* the backup set generation are given in Figure 163. The data amount of fused writes (2.6GB) corresponds exactly with the size of the backup set.

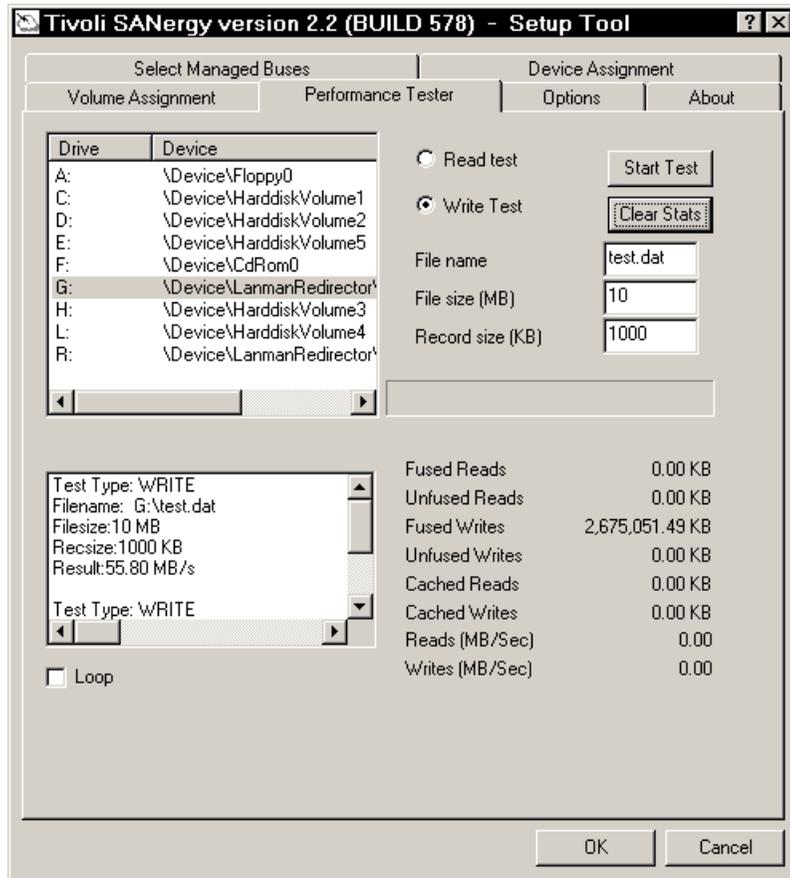


Figure 163. SANergy statistics after generate backup set

### 7.2.3.2 UNIX backup set generation

The file system we used for our UNIX example was /tmp on our Solaris system sol-e. Again, we show below the initial incremental backup of the file system.

```

# dsmc incr /tmp
.
.
.
Successful incremental backup of '/tmp'

Total number of objects inspected:    1,781
Total number of objects backed up:    1,777
Total number of objects updated:      0
Total number of objects rebound:     0
Total number of objects deleted:      0
Total number of objects expired:      0
Total number of objects failed:       0
Total number of bytes transferred:    87.24 MB
Data transfer time:                   94.44 sec
Network data transfer rate:           945.87 KB/sec
Aggregate data transfer rate:         806.97 KB/sec
Objects compressed by:                0%
Elapsed processing time:               00:01:50
#

```

After the successful backup, we generated a backup set of /tmp using the *sanbackupset* device class. The generate backupset and query backupset commands and their output are shown here:

```

Session established with server JAMAICA_SERVER1: Windows NT
  Server Version 4, Release 1, Level 3.0
  Server date/time: 04/23/2001 08:57:49  Last access: 04/23/2001 08:51:50

tsm: JAMAICA_SERVER1>generate backupset sol-e solTmp /tmp ret=5
desc="Solaris tmp backup set" devc=sanbackupset
ANS8003I Process number 20 started.

tsm: JAMAICA_SERVER1>q backupset sol-e

      Node Name: SOL-E
      Backup Set Name: SOLTMP.89591
      Date/Time: 04/23/2001 09:11:41
      Retention Period: 5
      Device Class Name: SANBACKUPSET
      Description: Solaris tmp backup set

tsm: JAMAICA_SERVER1>

```

Monitoring the SANergy Setup Tool/Performance Tester screen on jamaica, as we did with the Windows test, we saw the fused write value increase. When the `generate backupset` I/O was being handled by SANergy, it took 40 seconds to create the backup set to a SANergy volume. This compared with 2 minutes 43 seconds to create a traditional LAN-based backup set.

## 7.2.4 Backup sets as seen by the server

The `generate backupset` command creates files in the directory specified with the device class definition. It also creates objects in the Tivoli Storage Manager server database which can be queried to help an administrator manage and use the backup sets. The output from the `query backupset` command is shown below and shows our backup sets from sol-e and cerium. This output is identical to how it would appear in a non-SANergy environment.

```
tsm: JAMAICA_SERVER1>q backupset

      Node Name: CERIUM
      Backup Set Name: EDRIVE.86021
      Date/Time: 04/20/2001 08:30:02
      Retention Period: 5
      Device Class Name: SANBACKUPSET
      Description: Cerium E$ backupset

      Node Name: SOL-E
      Backup Set Name: SOLTMP.89591
      Date/Time: 04/23/2001 09:11:41
      Retention Period: 5
      Device Class Name: SANBACKUPSET
      Description: Solaris tmp backup set
```

Also, there is generated volume history information for each backup set. Those records can be displayed with the `query volhist type=backupset` command. This information can be used to tie the backup sets to the actual disk files created.

The screen below shows the volume history data for our backup sets. It shows the command we issued to generate each backup set in the *Command* field for the first volume for each backup set. We can see that the backup set for cerium (containing 2.5 GB of data) used three volumes, while the set for sol-e (using 87MB of data) used only one volume. Remember that the device class was defined in 7.2.2, “Defining the backup set device class” on page 208 to use a volume size of 1GB. Each record also shows the fully qualified file name for each volume of the backup set. The file names are what we expect to see on Windows in the target directory.

```

tsm: JAMAICA_SERVER1>q volh type=backupset

    Date/Time: 04/20/2001 08:30:02
    Volume Type: BACKUPSET
    Backup Series:
    Backup Operation:
    Volume Seq: 1
    Device Class: SANBACKUPSET
    Volume Name: G:\TSMBACKUPSETS\87780602.OST
    Volume Location:
    Command: generate backupset cerium edrive \\cerium\e$ ret=5
            desc="Cerium E$ backupset" devc=sanbackupset

    Date/Time: 04/20/2001 08:30:02
    Volume Type: BACKUPSET
    Backup Series:
    Backup Operation:
    Volume Seq: 2
    Device Class: SANBACKUPSET
    Volume Name: G:\TSMBACKUPSETS\87781236.OST
    Volume Location:
    Command:

    Date/Time: 04/20/2001 08:30:02
    Volume Type: BACKUPSET
    Backup Series:
    Backup Operation:
    Volume Seq: 3
    Device Class: SANBACKUPSET
    Volume Name: G:\TSMBACKUPSETS\87781482.OST
    Volume Location:
    Command:

    Date/Time: 04/23/2001 09:11:41
    Volume Type: BACKUPSET
    Backup Series:
    Backup Operation:
    Volume Seq: 1
    Device Class: SANBACKUPSET
    Volume Name: G:\TSMBACKUPSETS\88042301.OST
    Volume Location:
    Command: generate backupset sol-e solTmp /tmp ret=5 desc="Solaris tmp
            backup set" devc=sanbackupset

```

We can see the actual files created and their respective sizes in a directory listing of the device class directory, as shown in the next screen.

```
G:\TSMbackupsets>dir
Volume in drive G is MSSD11
Volume Serial Number is FC60-C63C

Directory of G:\TSMbackupsets

04/23/2001  09:17a    <DIR>          .
04/23/2001  09:17a    <DIR>          ..
04/20/2001  08:35a      1,073,741,824  87780602.OST
04/20/2001  08:46a      1,073,741,824  87781236.OST
04/20/2001  08:50a           591,769,092  87781482.OST
04/23/2001  09:17a           92,378,231  88042301.OST
           4 File(s)  2,831,630,971 bytes
           2 Dir(s)  26,937,036,800 bytes free

G:\TSMbackupsets>
```

## 7.2.5 Restoring from a backup set

We will now discuss considerations for restoring backup sets in a SANergy environment. The backup sets we created in the prior steps will be used for the restore examples. We will look at backup set restore on both Windows and UNIX.

There are three sources or locations which can be used to write and restore backup sets: the Tivoli Storage Manager server storage pools, a tape, or a file. If you use server storage pool space to store backup sets, the server itself must be available and communicating with the client in order to be able to restore. Also, the data will be transferred over the LAN. Restoring from a tape generated backup set requires that the client have a suitable locally attached tape drive and device driver to be able to read the set. Using a SANergy-managed volume eliminates these requirements and allows the restore to take place, over the SAN, without requiring any interaction with the Tivoli Storage Manager server. Therefore, we will only look at restoring a backup set from a file.

The bold arrows in Figure 164 show the path of the data for our Windows and UNIX client backup set restores.

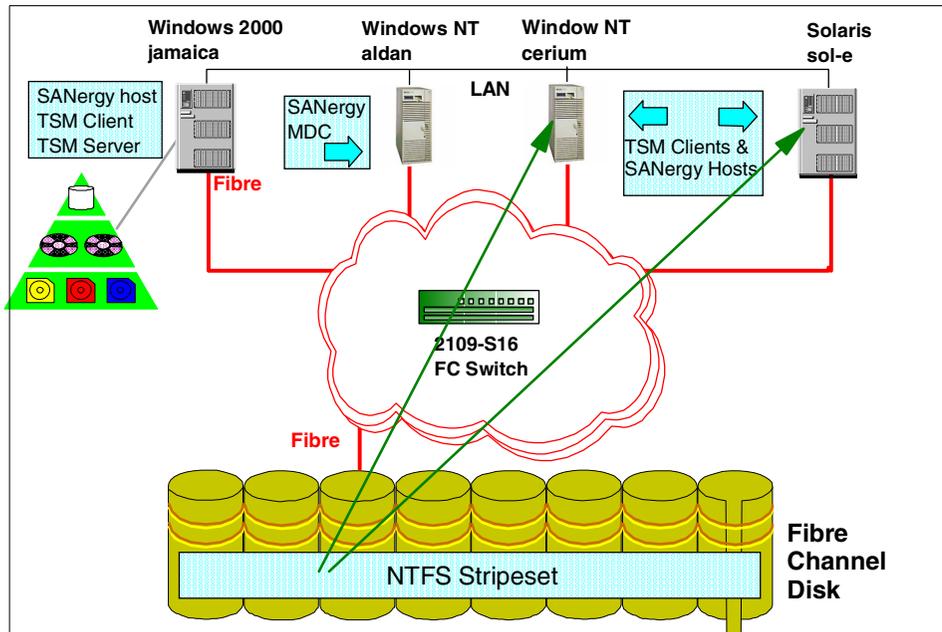


Figure 164. Backup set restore data flow

### 7.2.5.1 Windows backup set restore

The data in the backup set is recovered to the client either with the `restore backupset` command or using the Backup/Archive client GUI. We use the GUI for the Windows example.

The GUI is initiated using the normal procedure: **Start->Programs->Tivoli Storage Manager->Backup Archive GUI.**

When the Tivoli Storage Manager client GUI is displayed, select **Restore**, then from the Restore panel, select **Backup Sets** and then click the "+" before **Local**. You will be presented with a screen allowing you to specify the fully qualified identifier of the backup set file. You can type it in or use the Browse function. In our case, the client cerium has accessed the SANergy share as drive L. This is shown in Figure 165.

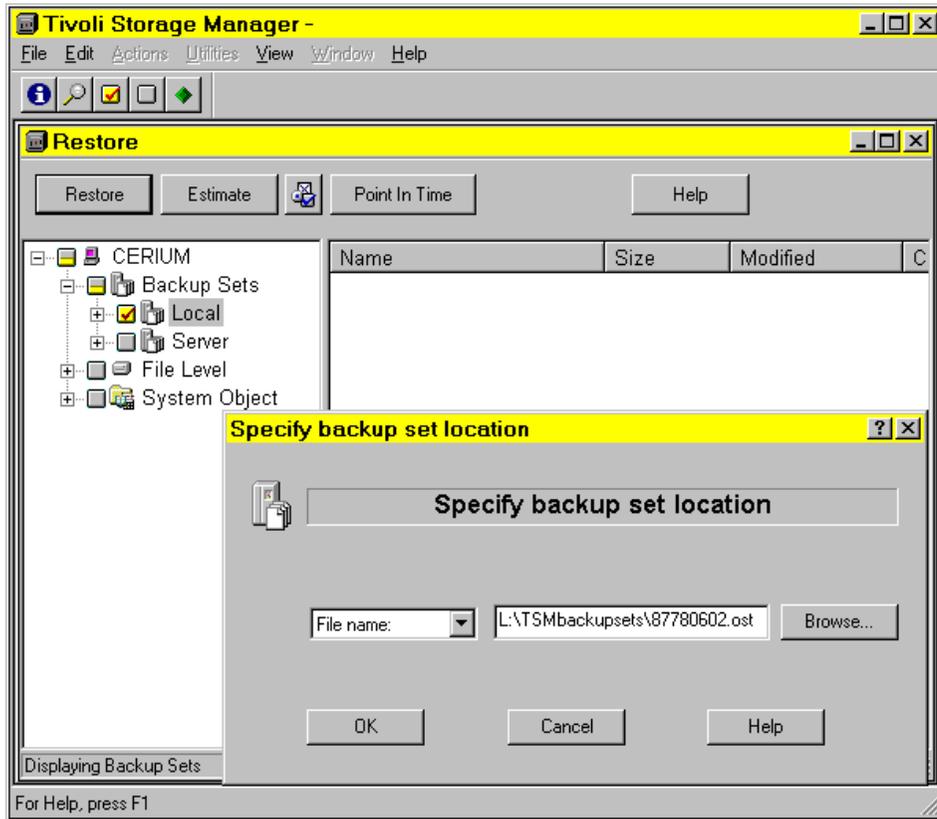


Figure 165. Windows backup set file selection

After clicking the **OK** button, the GUI displays the backup set name and displays the file spaces within it that can be restored. In our case the only filespace is the E drive. We selected that and clicked the Restore button as shown in Figure 166.

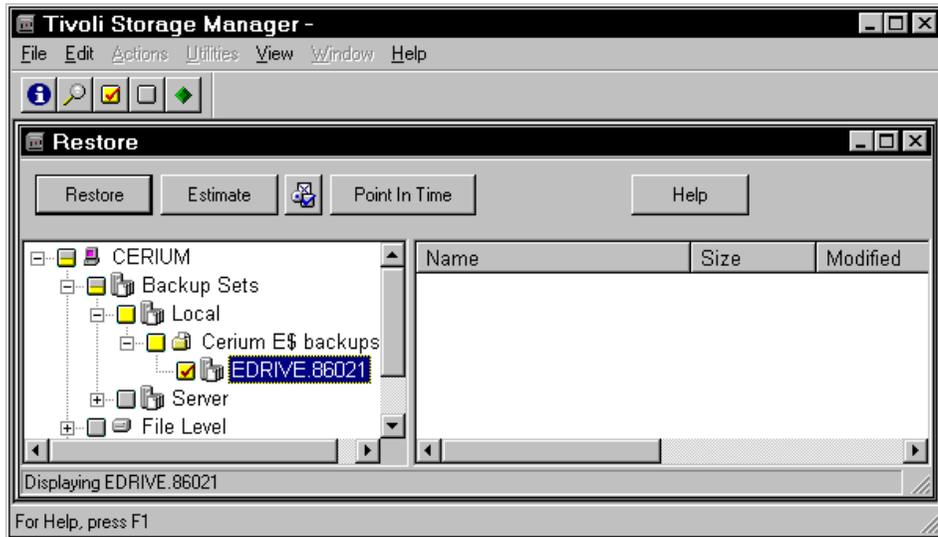


Figure 166. Windows backup set restore

The client then requests the restore location and restore options. We selected to restore the data to another drive, H:. The restore then proceeds until it reaches the end of the first volume of the backup set at which point it prompts for the next volume. The prompt for our third volume is shown in Figure 167.

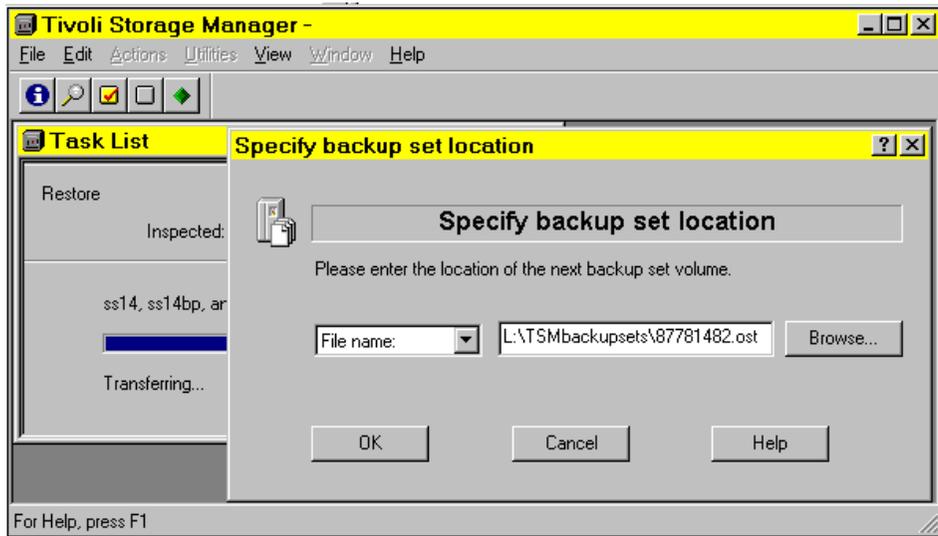


Figure 167. Windows next backup set volume

When the data from all volumes of the backup set had been restored, we received a status report indicating successful completion as shown in Figure 168. We also checked the statistics on the SANergy Setup Tool/Performance Tester screen and found all of the backup set data read was handled by SANergy.

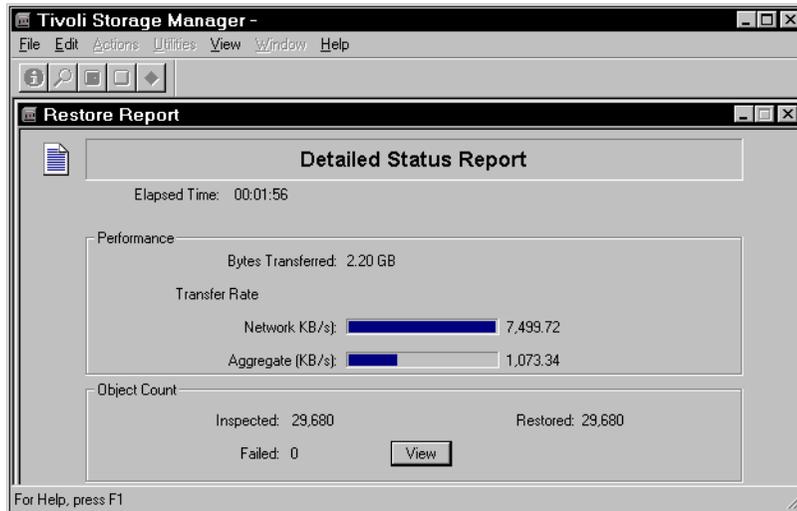


Figure 168. Windows backup set restore statistics

### 7.2.5.2 UNIX backup set restore

The data in the backup set is recovered to the client using the `restore backupset` command. From our Tivoli Storage Manager client `sol-e`, we first query the backup set to verify we have access to it as a file. We do this through the `backup/archive` client command line. This is shown in the next screen view.

For the query we must know the location and name of the file, not the backup set name. This information we obtained from the volume history query described previously in 7.2.4, “Backup sets as seen by the server” on page 215. Note how we convert the Windows type file names to UNIX convention for the UNIX client — using forward instead of back slashes.

```

# dsmc
Tivoli Storage Manager
*** Fixtest, Please see README file for more information ***
Command Line Backup Client Interface - Version 4, Release 1, Level 2.12
(C) Copyright IBM Corporation, 1990, 2000, All Rights Reserved.

tsm> q backupset -loc=file /G/TSMbackupsets/88042301.OST
Node Name: SOL-E

  Backup Set Name          Generation Date  Retention Description
-----
1 SOLTMP.89591            04/23/01  09:11:41 5          Solaris tmp backup set
tsm>

```

Now that we have verified our access to the backup set file, we are ready to perform the restore. We are going to restore to a subdirectory on our Solaris system, called /newTmp. The `restore backupset` command and results summary are shown in the screen below.

#### Attention

SANergy will only manage the I/O's of a UNIX application, such as the Tivoli Storage Manager client, if the proper environment variables are set in advance. This is done by running the setup shell script appropriate to your platform and shell. For our Solaris environment, we used the script `SANergyshsetup`. We discussed using this script in 2.3.2.3, "Using SANergy with a process" on page 55.

```

# . /opt/SANergy/SANergyshsetup
# dsmc
Tivoli Storage Manager
*** Fixtest, Please see README file for more information ***
Command Line Backup Client Interface - Version 4, Release 1, Level 2.12
(C) Copyright IBM Corporation, 1990, 2000, All Rights Reserved.

tsm> restore backupset -loc=file -subdir=yes
/G/TSMbackupsets/88042301.OST /tmp/* /d102/newTmp/
.
.
.
Restore processing finished.

Total number of objects restored:      1,776
Total number of objects failed:        0
Total number of bytes transferred:     87.42 MB
Data transfer time:                    4.13 sec
Network data transfer rate:            21,663.70 KB/sec
Aggregate data transfer rate:          6,718.12 KB/sec
Elapsed processing time:                00:00:13
tsm>

```

The backup set was successfully restored to the new directory. The Performance Test screen of Solaris SANergy GUI showed the backup set data read was fused, and the restore time was 13 seconds with SANergy versus over 2 minutes using a normal LAN-based restore.

It should be noted that when the location of a backup set is a file, the Backup/Archive client does not establish a session with the Tivoli Storage Manager server for queries or restores — therefore the server is not required to be available.

---

### 7.3 Commercial database backup from SANergy host

Implementing databases on SANergy managed volumes means running the database with data on a network drive. Depending on the particular database, there may be some limitations or restrictions from the vendor for this type of implementation. Certain flags or options may be required for each database to enable storing database files on a network drive. An example of MS SQL Server is shown in 6.5, “Databases on SANergy hosts” on page 180.

But even in a SANergy environment, there is one configuration using databases that has no restrictions — installing the database on the MDC. Running on the MDC, the database sees the shared volume as a locally attached drive and is not aware of the SANergy service at all.

You might ask, now, what is the benefit of using SANergy in such a configuration. Its advantage is the possibility to back up the database “application-server-free” with SANergy. That means the database files are backed up without having the database server involved in the backup process. The backup will be done by a SANergy host that gets fused access to the storage volume on the SAN where the database files are located. The configuration may look like this (Figure 169).

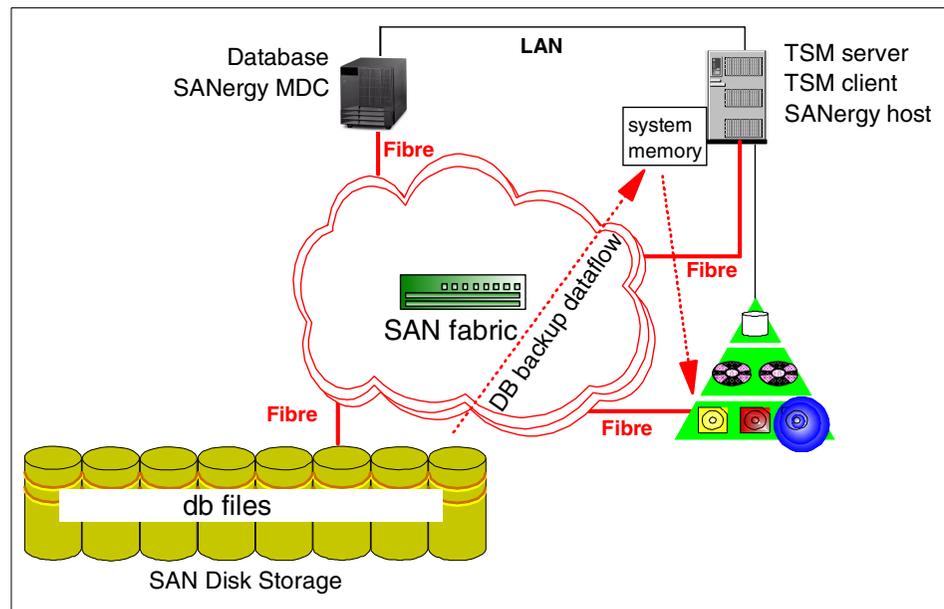


Figure 169. Backing up databases with SANergy

A Tivoli Storage Manager client and server are installed on a SANergy host which has mapped the volumes where the database is stored. The Backup/Archive client can now back up the database files that it can access over the network drive. The data flow is from the SAN disks to the system memory of the Backup/Archive client (SANergy host), then to the Tivoli Storage Manager server over shared memory and finally direct to disk or tape storage pools. There is no LAN traffic, besides the metadata between SANergy MDC and SANergy host, involved in this backup process.

However, this technique presumes that the database is in a consistent state when the files are being backed up. Therefore the database must be put into the backup mode on the MDC prior to copying the files. An example of this is using the `ALTER TABLESPACE BEGIN BACKUP` command for Oracle. More information on the correct way to do this for a particular database platform is available in the vendor's documentation. A further restriction is that this procedure applies only for offline backups of the database, that is, no transactions are allowed during the backup session and also no incremental backup is possible.

---

## 7.4 Heterogeneous SANergy backup configurations

Our Tivoli Storage Manager backup configurations in this chapter have used a homogeneous environment — that is, the MDC and the SANergy host (which is also the Tivoli Storage Manager client/server) use the same operating system. However, what if the SANergy host and MDC are on different operating system platforms — can we perform application server-free backup and restore with the Tivoli Storage Manager client on the SANergy host?

For example, suppose the SANergy host is AIX with a Windows 2000 MDC. Is it possible to install the AIX backup/archive client on the SANergy AIX host and successfully backup/restore data owned by the Windows MDC? The answer is no. Although the Tivoli Storage Manager client on the AIX host can back up individual files, it will not correctly back up and restore the file and folder or directory attribute information, that is, the metadata. This is because the AIX file system does not understand Windows 2000 file attributes, and is a restriction outside the control of both Tivoli Storage Manager and SANergy. Therefore, when performing backup of MDC data with a Tivoli Storage Manager client on a SANergy host, it is recommended that the platform type of the host be identical to that of the MDC. This is described above in 7.1, “Application server-free backup and restore” on page 191.



## Chapter 8. SANergy and OTG DiskXtender 2000

As we have seen in the previous chapters, SANergy provides the ability to consolidate storage devices and facilitate storage administration by minimizing the number of file systems needed in an IT environment. This consolidation also means that we can configure larger file systems that have to be managed on the MDC.

A proven technology for maintaining sufficient space on file servers is space management. In order to provide the available disk space for fast access to important and frequently needed data — and, on the other hand, not wasting expensive disk space for inactive or less frequently used data — the space management software migrates files (selected by defined rules) from disk to a less expensive media (for example, tape or optical media). If any migrated data is required for access, the management software will automatically recall it from its secondary location to the original disk. This procedure is transparent to the end user, so that to them, it appears that the amount of available disk space is virtually unlimited.

Tivoli offers the OTG DiskXtender 2000 software for managing volumes on Windows NT and Windows 2000 platforms. DiskXtender 2000 migrates data to a Media Service, the software managing the tape devices. It is possible to define a Tivoli Storage Manager server as the media manager, so that space managed files from DiskXtender are stored as backup objects in the storage pool hierarchy of the Tivoli Storage Manager server. More information on OTG DiskXtender 2000 is available from the Web site:

<http://www.otg.com/tivoli>

We will refer to this product simply as DiskXtender for the remainder of this chapter.

In the following sections we will show how to configure a SANergy environment where the volumes owned by the MDC are also managed by DiskXtender in order to virtually expand the amount of available disk space accessible to the SANergy hosts. We want the file migration process to be LAN-free — that is, that the data transfer from DiskXtender to the Tivoli Storage Manager server will flow over the SAN. Therefore we will also utilize the LAN-free feature provided for the Tivoli Storage Manager client API, which will be called by the DiskXtender procedure when migrating files.

At this point we assume that the SANergy environment is properly installed and running. For further information, see the Chapter 2, “SANergy with a Windows MDC” on page 19. We also assume that there is a Tivoli Storage Manager server running with a SAN-attached tape drive. For implementation guidance on Tivoli Storage Manager server and clients, see the redbooks *Using Tivoli Storage Manager in a SAN Environment*, SG24-6132, and *Getting Started with Tivoli Storage Manager: Implementation Guide*, SG24-5416, as well as the product documentation referred to in the Appendix F.3, “Other resources” on page 296.

In our implementation, the machine diomedes is the MDC running on Windows 2000 Advanced Server (Figure 170). It provides the SANergy host cerium with fused access to the shared storage. On the MDC this shared volume is also managed by DiskXtender 2000. The Tivoli Storage Manager server is installed on the machine named jamaica which is running on Windows 2000 Server.

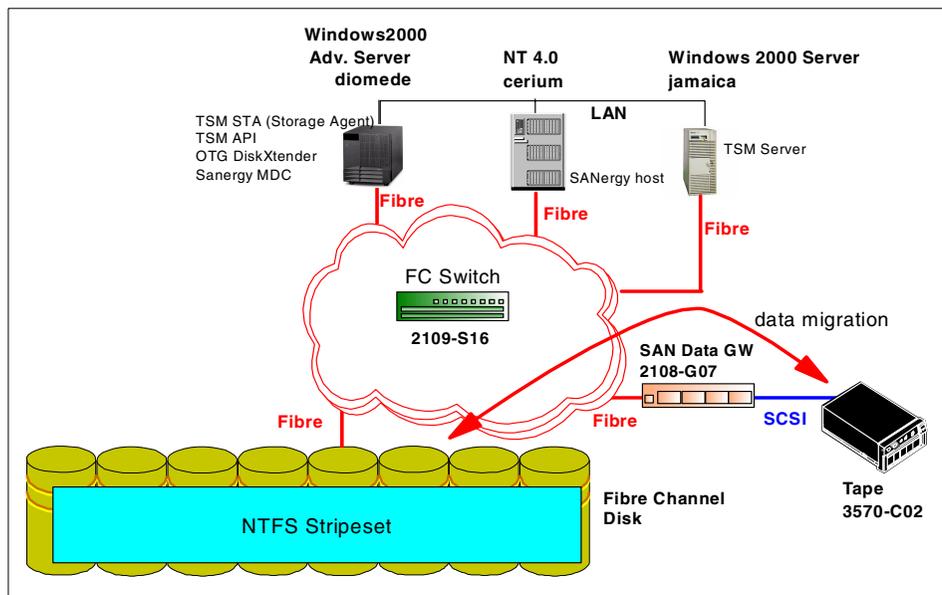


Figure 170. Space management in a SANergy environment

## 8.1 Configuring the Tivoli Storage Manager server

The Tivoli Storage Manager server (in our case, jamaica) should be configured with a supported tape drive. This may or may not be attached via the SAN, although in our configuration we used an IBM 3570 Magstar MP tape library attached via the IBM SAN Data Gateway. For implementation guidance on Tivoli Storage Manager server and clients, see the redbook *Getting Started with Tivoli Storage Manager: Implementation Guide*, SG24-5416 as well as the product documentation referred to in the Appendix F.3, “Other resources” on page 296.

First we need to define a client on the Tivoli Storage Manager server that is designated for use by the DiskXtender client. We added the node *otg* to the server using the `REGISTER NODE` administrative command. We also created a new policy domain for this node on the server with a new management class named *DiskXtender* and associated the `copy destination` option in the corresponding backup copy group of this management class to a tape storage pool using the SAN-attached library. The following screens show how our server definitions were set up.

```
tsm: JAMAICA_SERVER1>q policy
```

| Policy Domain Name | Policy Set Name | Default Mgmt Class Name | Description                   |
|--------------------|-----------------|-------------------------|-------------------------------|
| STANDARD           | ACTIVE          | STANDARD                | Installed default policy set. |
| STANDARD           | STANDARD        | STANDARD                | Installed default policy set. |
| OTG                | ACTIVE          | DISKXTENDER             | Policy set for OTG domain     |
| OTG                | STANDARD        | DISKXTENDER             | Policy set for OTG domain     |

```
tsm: JAMAICA_SERVER1>q node otg
```

| Node Name | Platform    | Policy Domain Name | Days Since Last Access | Days Since Password Set | Locked? |
|-----------|-------------|--------------------|------------------------|-------------------------|---------|
| OTG       | DiskXtender | OTG                | <1                     | 11                      | No      |

```
tsm: JAMAICA_SERVER1>q copygroup otg standard diskxtender type=backup f=d
```

```
Policy Domain Name: OTG
Policy Set Name: STANDARD
Mgmt Class Name: DISKXTENDER
Copy Group Name: STANDARD
Copy Group Type: Backup
Versions Data Exists: 1
Versions Data Deleted: 0
Retain Extra Versions: 30
Retain Only Version: 0
Copy Mode: Modified
Copy Serialization: Shared Static
Copy Frequency: 0
Copy Destination: OTGTAPE
Last Update by (administrator): ADMIN
Last Update Date/Time: 04/25/2001 08:25:06
Managing profile:
```

```
tsm: JAMAICA_SERVER1>q devclass 3570dev f=d
```

```
Device Class Name: 3570DEV
Device Access Strategy: Sequential
Storage Pool Count: 2
Device Type: 3570
Format: DRIVE
Est/Max Capacity (MB): 10,240.0
Mount Limit: DRIVES
Mount Wait (min): 10
Mount Retention (min): 1
Label Prefix: ADSM
Drive Letter:
Library: 3570LIB
```

```
tsm: JAMAICA_SERVER1>q library 3570lib
```

```
Library Name: 3570LIB
Library Type: SCSI
Device: LB1.3.0.3
```

```
tsm: JAMAICA_SERVER1>q drive
```

| Library Name | Drive Name | Device Type | Device    | ON LINE |
|--------------|------------|-------------|-----------|---------|
| 3570LIB      | DRIVE0     | 3570        | MT1.4.0.3 | Yes     |
| 3570LIB      | DRIVE1     | 3570        | MT1.2.0.3 | Yes     |

---

## 8.2 Installing the client and API on the SANergy MDC

At the time this book was written, the latest available Tivoli Storage Manager client code was version 4.1.2.12, which corresponds to package IP22151\_12. We installed the code by executing the file IP22151\_12.exe which can be downloaded from:

[www.tivoli.com/support/storage\\_mgr/clients.html](http://www.tivoli.com/support/storage_mgr/clients.html)

We used the default installation path c:\Program Files\Tivoli\TSM\. When asked for the installation type, we chose **Custom**, and selected the Tivoli Storage Manager client, the API, and the administrative client to be installed. The rest of the installation was straightforward.

At this point we had to create a option file for the Tivoli Storage Manager client API that will later be used by DiskXtender to connect to the Tivoli Storage Manager server.

We edited the sample option file dsm.smp that the installation process creates in the directory c:\Program Files\Tivoli\TSM\config and renamed it to dsm.opt. Here are the minimum settings needed for DiskXtender. Note the enablelanfree yes option.

```
*****
* Tivoli Storage Manager
*
* Sample dsm.opt for OTG DiskXtender 2000
*****
*=====
* TCP/IP
*=====
commethod      tcpip
tcpport        1500
TCPServeraddress  jamaica
nodename       otg
passwordaccess generate
enablelanfree  yes
```

---

## 8.3 Installing OTG DiskXtender 2000 on the SANergy MDC

The installation of DiskXtender was done by executing the setup.exe program on the installation CD ROM. The installation routine started with the window in Figure 171.

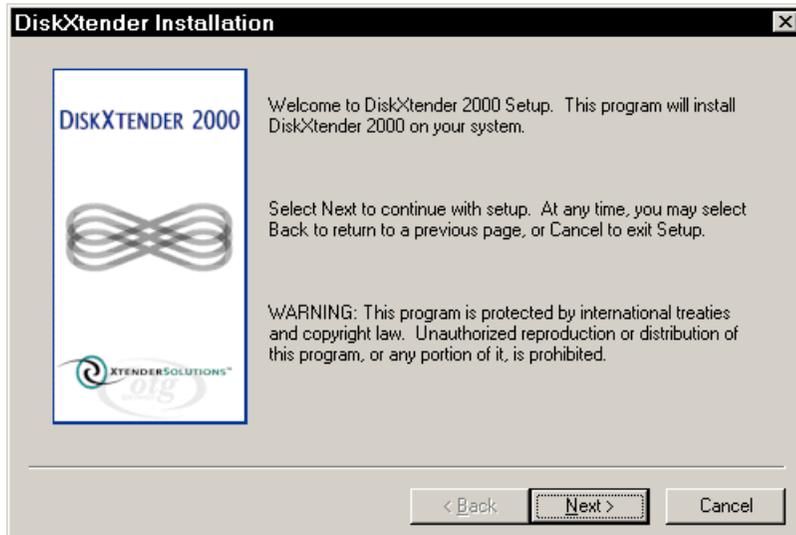


Figure 171. Installation program for DiskXtender 2000

Click **Next** to start the installation. A screen with the installation options will be displayed (Figure 172).

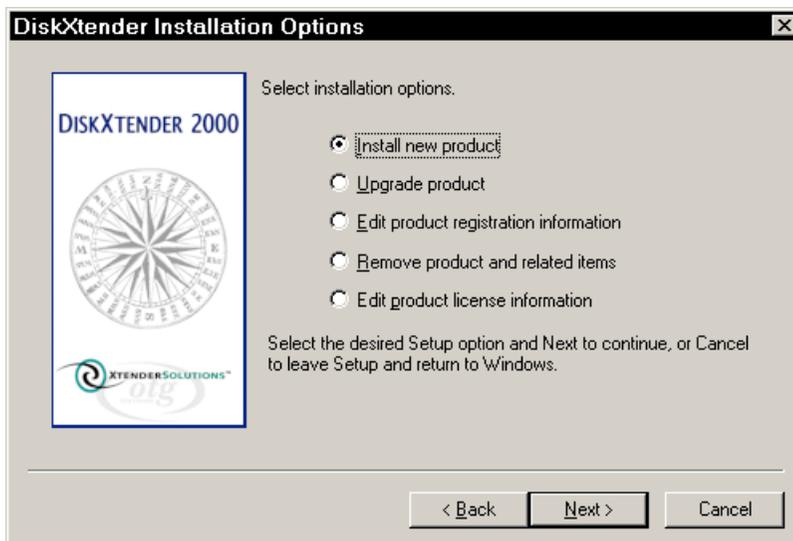


Figure 172. DiskXtender Installation Options

We selected **Install new product**. Click **Next** to get to the license agreement window. (This window is not shown here.)

Accept the license terms by checking the appropriate box and click **Next**. The registration form will appear. (This window is not shown here.)

You have to enter customer name and organization to get to the next screen by clicking **Next**. You get to the DiskXtender Service Account dialog box. (This window is not shown here.)

Enter the domain name, user name and user password for the account that you want DiskXtender to run with. Click **Next**. In the following window (Figure 173) you can choose whether you have a licensed copy of the code or just want to install a 30 day evaluation copy.

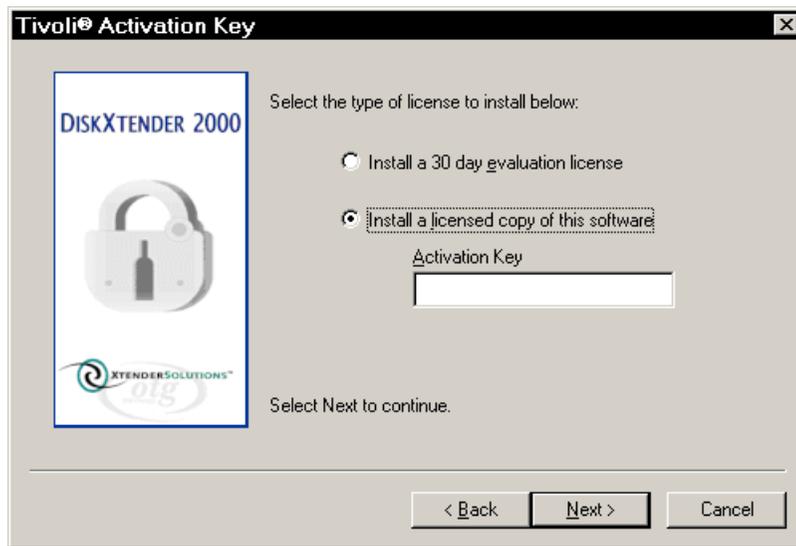


Figure 173. DiskXtender license type

Enter the Activation Key when installing a licensed copy and click **Next** to get to the screen shown in Figure 174.

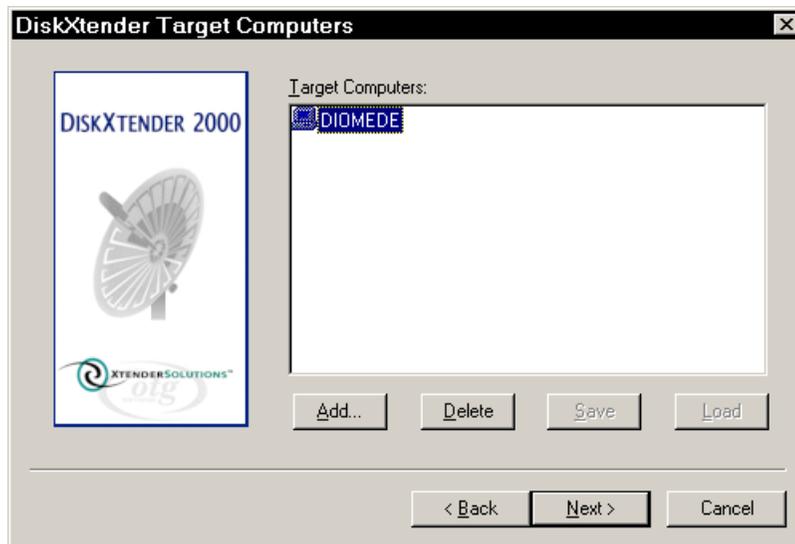


Figure 174. Choose target computers

In this dialog box you can select the computers where DiskXtender should be installed. We selected our SANergy MDC diomede here. Proceed by clicking **Next**. A setup summary will display. Review the setup information you have entered and select **Finish**. The installation ends with window shown in Figure 175.



Figure 175. Setup complete

The next step is to configure DiskXtender. Before starting the administrator tool you should copy the file `dsm.opt` file created in 8.1, “Configuring the Tivoli Storage Manager server” on page 229 to the installation directory of DiskXtender.

Copy the file in to the directory `c:\program files\OTG\DiskXtender\BIN`. This is necessary for the DiskXtender administrator to link to the Tivoli Storage Manager Media Services. After the file is copied, click **Start** in the foregoing figure to launch the administrator tool. A window will pop up, indicating that no media services are currently configured (Figure 176).



Figure 176. No media services currently configured

Select **Yes** to start the configuration. A configuration window shows that no services are currently installed. Click **Add** to get to the **Select Media Service Type** window (Figure 177).



Figure 177. Configure Media Services Type

Select Tivoli Storage Manager as the media service and click **Next**. In the following screen you are asked for the TSM option file (Figure 178).

**TSM Information**

Please enter the name of the options file and the client password for the 'TSM' media service below. The options filename should not include a path, and may be left blank if the default options file is to be used.

Options File:

Client Password:

< Back   Next >   Cancel   Help

Figure 178. TSM Information

As you have already copied the dsm.opt options file in a previous step to the DiskXtender directory (which is the default location expected by DiskXtender), you enter just the name of the option file in the first field as shown. Also, you must enter the password for the Tivoli Storage Manager nodename *otg*.

Click **Next** to get a summary of the information you entered. If the information is correct, click **Finish** to start the configuration process. The installation process will respond with the following window (Figure 179).

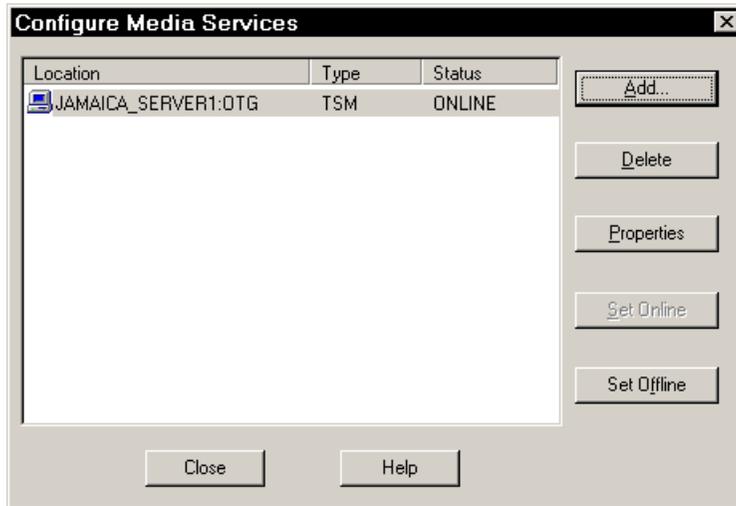


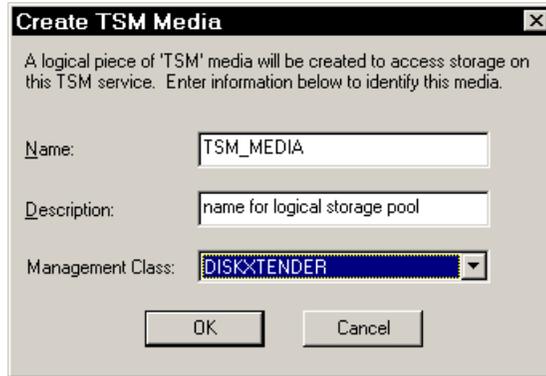
Figure 179. TSM Media Service for DiskXtender

The Status online field indicates that DiskXtender has successfully connected to the Tivoli Storage Manager Media Service which is called jamaica\_server1. Click **Properties** to add media to the media services (Figure 180).



Figure 180. Media Service Properties

There is no media yet in the media list. Click **Create** to get to Figure 181.



**Create TSM Media**

A logical piece of 'TSM' media will be created to access storage on this TSM service. Enter information below to identify this media.

Name: TSM\_MEDIA

Description: name for logical storage pool

Management Class: DISKXTENDER

OK Cancel

Figure 181. Create TSM Media

Media is a logical object on the DiskXtender side that enables the communication with the Tivoli Storage Manager server and through it, with the physical media. Enter a name for the media (which is arbitrary) for identification reasons in the field **Name** and select the management class created on the server for DiskXtender. You will be able to choose from all available management classes in the drop down menu in the field **Management Class**.

We selected the management class **DISKXTENDER** which we created in 8.1, “Configuring the Tivoli Storage Manager server” on page 229. Click **OK** to continue. Since the management class is mapped to the tape library storage pool via the copy destination parameter in the copy group, the association between DiskXtender and the Tivoli Storage Manager media is made. The following window shows the created media (Figure 182).

*Figure 182. Media list on Media Service Properties*

Click the **OK** button to return to the media services window and click **Close** to terminate media services configuration. A pop-up window will indicate that there are no drives configured to be managed by DiskXtender (Figure 183).

DiskXtender uses the term *extended drives* to refer to drives which will

Click **Yes** to start configuring the SANergy volume as an extended drive to DiskXtender. The following information is displayed (Figure 184).



Figure 184. New Extended Drive screen

These are the steps that must be performed to enable DiskXtender to manage the specified drive. We already have completed the step **1) Configure one or more media services**, so we can proceed with the step **2) Create media folders** after selecting the drive to be managed by DiskXtender. Click **Next** to get a list of all available drives to select for extending (Figure 185).

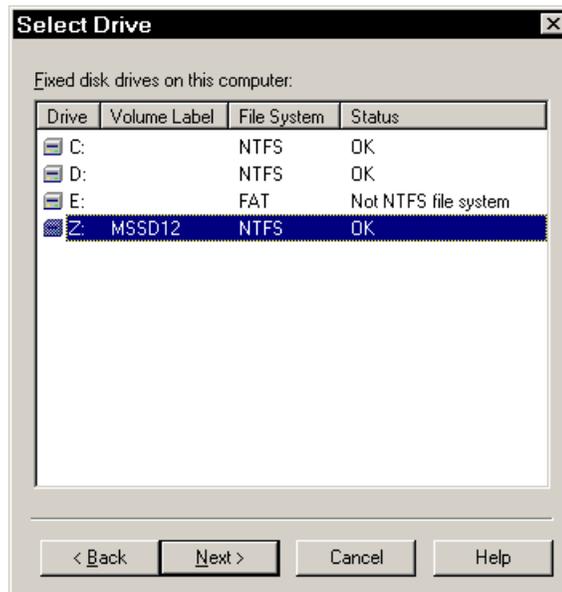


Figure 185. Select Drive for extension

We selected the shared SAN volume **Z:\** with the label **MSSD12** on the SANergy MDC for extension and proceeded with **Next**. In the following window we have to assign media to this extended drive (Figure 186).

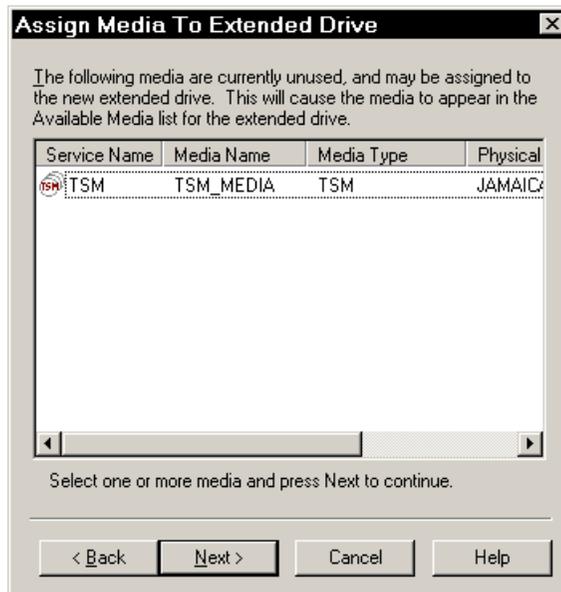


Figure 186. Assign Media To Extended Drive

A list with currently unused media is shown. We selected the previous created media **TSM\_MEDIA** for this drive. Click **Next** to continue with Figure 187.



Figure 187. Settings for DiskXtender operations

From this screen you can specify the times where DiskXtender should be active for moving files, media task processing, and media copy updates.

Media task processing includes such tasks as media compaction, formatting and labelling media and restoring files. Media copy updates will duplicate original media (containing migrated files) onto new blank media. This function is analogous to Tivoli Storage Manager's copy storage pool function. For more information, refer to the manual *OTG DiskXtender2000 Data Manager System Guide*, which is available at the following Web site:

<http://www.otg.com/tivoli/Data Manager System Guide.pdf>

Select **Schedule** to get to the window shown in Figure 188.

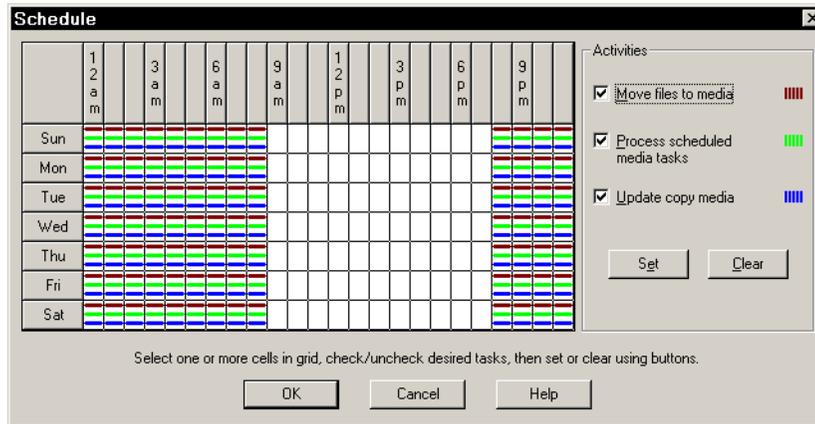


Figure 188. DiskXtender Scheduler

You can customize the service to your needs at this point and schedule the activity of all three processes independently from each other. By default the working hours from 9 am to 8 pm are inactive. Highlight the fields corresponding to the time you will activate a service, check the appropriate tasks on the **Activities** panel and click **Set**. Exit the window by clicking the **OK** button.

From the window in Figure 187, you also can configure the frequency of drive scans on the extended drive. Click **Drive Scan** to get to the window shown in Figure 189.

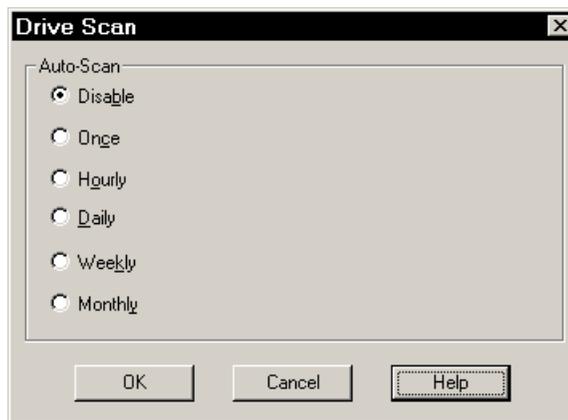


Figure 189. Drive Scan scheduling

Drive scans must be performed periodically in order to move files to media and purge file data to create space on the extended drive. During a drive scan, DiskXtender inventories all of the files on an extended drive and checks each file against the migration rules for the drive, adding eligible files to the move and purge lists.

DiskXtender uses the term *move* to describe the process of copying eligible files to media without deleting them from their original location (analogous to Tivoli Space Manager pre-migration). The term *purge* describes the process of moving or migrating the files to the tape pool and deleting them from the disk.

The Drive Scan Schedule option of the extended drive properties defines how frequently drive scans will occur. Drive scans can be scheduled to occur once, hourly, daily, weekly, or monthly.

When proceeding from the window in Figure 187 with the **Next** button, you get to the **Options** window for the defined DiskXtender extended drive (Figure 190).

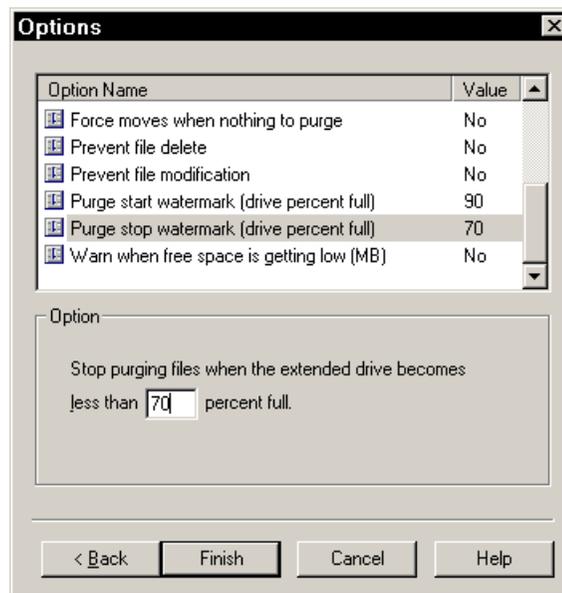


Figure 190. Options for extended drive

We accepted the default values here, except for the **Purge start watermark / (drive percent full)** and **Purge stop watermark / (drive percent full)** parameters. These two fields indicate how full the extended drive should be for when files begin to migrate from the extended disk and when the migration process stops — that is, they act as high and low water marks. DiskXtender uses the term Click **Finish** to initialize the extended drive.

Now that the SAN volume is managed by DiskXtender, you are asked to add a media folder to this drive (Figure 191).



Figure 191. No media folder

You can specify one or more folders (or directories) to be managed by DiskXtender on the extended drive. Only files in a defined media folder are eligible for move and purge operations. Click **Yes** to add a media folder (Figure 192).

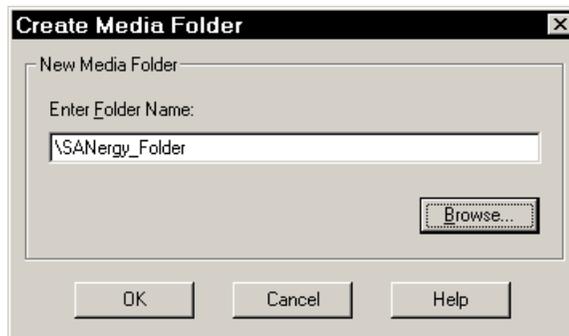


Figure 192. Create Media Folder

A media folder in DiskXtender creates an association between a physical directory on the extended drive and the media to which the files stored in that directory should be moved.

You have the following choices:

- To create a new folder on the extended drive, type the media folder name in the **Enter Folder Name** text box.
- To use a folder that already exists on the extended drive as a media folder, click **Browse**. The Select Folder dialog box appears. Select the folder and click **OK**. The folder name appears in the **Enter Folder Name** text box.

We chose the folder **\SANergy\_Folder** on the extended drive to be managed by DiskXtender. Click **OK** to get to the main DiskXtender administrator GUI (Figure 193).



Figure 193. DiskXtender Administrator main window

Next we added media to the media folder by right-clicking on **Media** and selecting **Add Media**. The list of original media available for this folder appears (Figure 194).

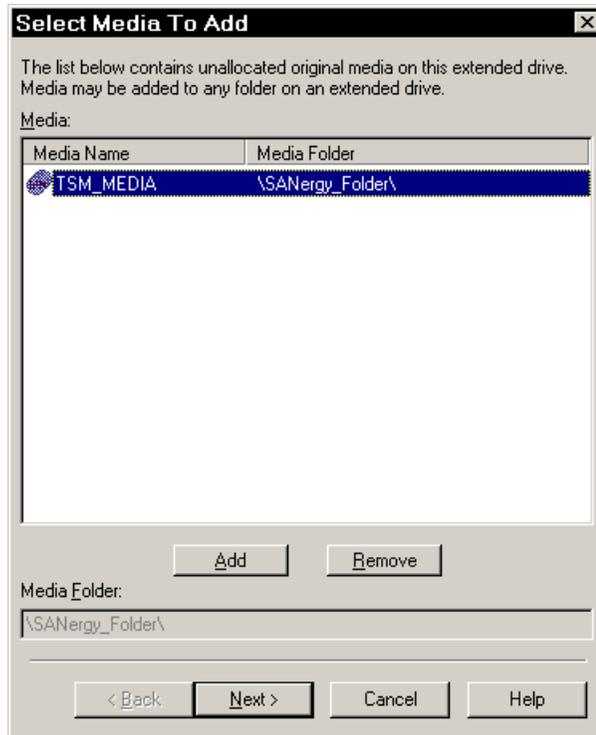


Figure 194. Select Media to Add

We selected TSM\_MEDIA and clicked on the **Add** button to associate the folder to the media. Next we created a move group by right-clicking on **Move Group** in the Figure 193 and selecting **New**. The window in Figure 195 opens.

**New Move Group**

Group Name  
A move group is identified by a unique descriptive name.  
Name:   
Enter a name that will easily identify this move group.

Media Type  
A move group's media must be of the same supported writable type.  
Media Type:   
Select a media type for this move group from the list above.

< Back   Next >   Cancel   Help

Figure 195. New Move Group

We specified the name **SANergy\_Movegroup** which is arbitrary and the **Media\_Type TSM**. Click **Next** to add media to this group in the next window. Select **Add** to see the window in Figure 196.

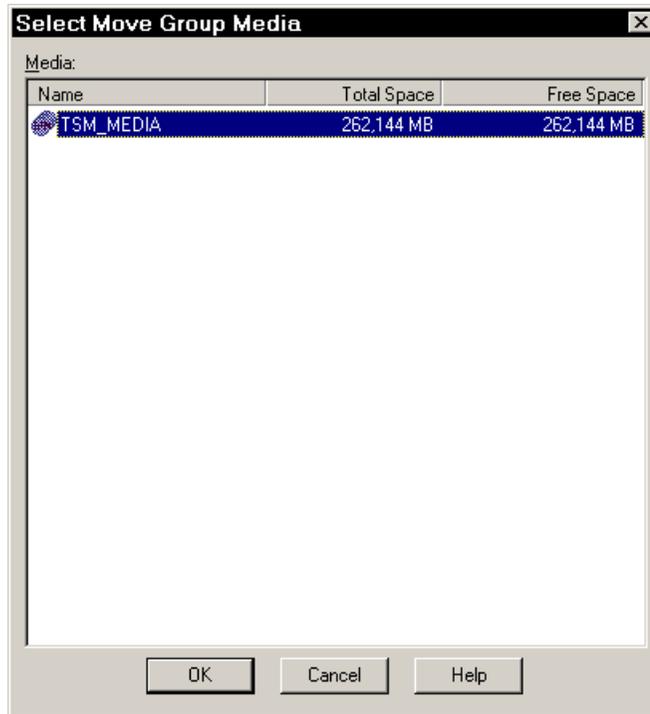


Figure 196. Select Move Group Media

Click **OK** to continue. The next window shows you the summary of this configuration. Click **Finish** to exit.

Now you have to define move rules and purge rules for the selected folder. Right-click the **Move Rules** icon in the DiskXtender main administrator GUI in Figure 193 on page 247. Select **New** to get to the window in Figure 197.

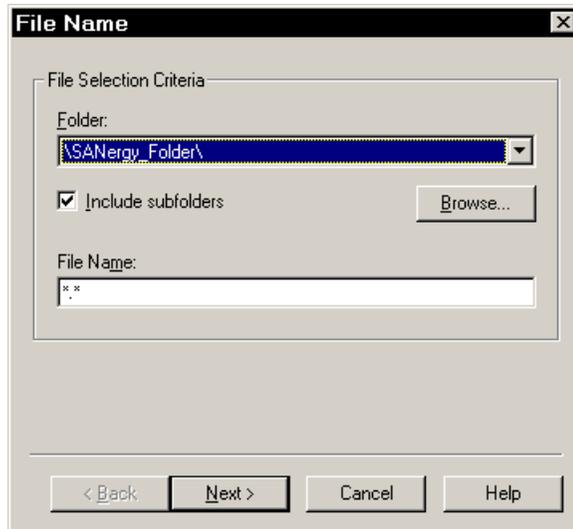


Figure 197. Move Rules definition for folder

Select the folder in the **Folder** field to be configured and click **Next**. In the following steps you can specify the attributes, the size, the age and the type of files that will be moved when disk space is needed. We accepted the default values at this point for our installation. By default, no further specification is selected, and all files are treated equally. You can define purge rules in the same manner.

To check if DiskXtender is properly installed and that communication to the Tivoli Storage Manager server is working, we copied some files into the **SANergy\_Folder** so that it would exceed the purge start watermark. This causes the move process to begin.

On the Tivoli Storage Manager server, we opened an administrative client session and queried for open sessions. We saw the two DiskXtender sessions in the screen below. One session is always open as long as the DiskXtender service is running, and the other one is for transferring data when the move process is started. You can see that the session 317 is increasing the number of bytes received, indicating that it is sending data to the server.

```
tsm: JAMAICA_SERVER1>q sess
```

| Sess Number | Comm. Method | Sess State | Wait Time | Bytes Sent | Bytes Recvd | Sess Type | Platform         | Client Name |
|-------------|--------------|------------|-----------|------------|-------------|-----------|------------------|-------------|
| 213         | Tcp/Ip       | Run        | 0 S       | 6.9 K      | 108         | Admin     | Win95            | ADMIN       |
| 316         | Tcp/Ip       | IdleW      | 50 S      | 4.7 K      | 726         | Node      | DiskXte-<br>nder | OTG         |
| <b>317</b>  | Tcp/Ip       | RecvW      | 0 S       | 64.9 K     | 11.0 M      | Node      | DiskXte-<br>nder | OTG         |
| 318         | Tcp/Ip       | Run        | 0 S       | 7.0 K      | 324         | Admin     | WinNT            | ADMIN       |

```
tsm: JAMAICA_SERVER1>q sess
```

| Sess Number | Comm. Method | Sess State | Wait Time | Bytes Sent | Bytes Recvd | Sess Type | Platform         | Client Name |
|-------------|--------------|------------|-----------|------------|-------------|-----------|------------------|-------------|
| 213         | Tcp/Ip       | Run        | 0 S       | 7.0 K      | 108         | Admin     | Win95            | ADMIN       |
| 316         | Tcp/Ip       | IdleW      | 54 S      | 4.7 K      | 726         | Node      | DiskXte-<br>nder | OTG         |
| <b>317</b>  | Tcp/Ip       | Run        | 0 S       | 67.6 K     | 13.0 M      | Node      | DiskXte-<br>nder | OTG         |
| 318         | Tcp/Ip       | Run        | 0 S       | 7.7 K      | 344         | Admin     | WinNT            | ADMIN       |

After migrating files from the managed folder on the extended drive, the view on the Explorer shows these files in the SANergy\_Folder with a special icon indicating the migrated status of the files (Figure 198).

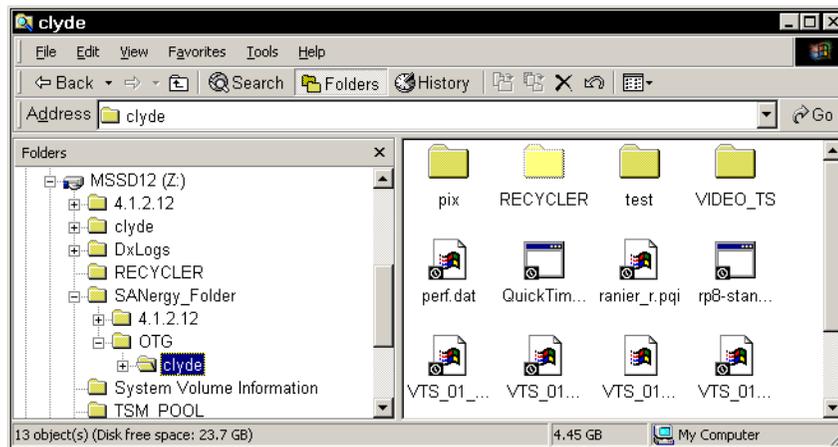


Figure 198. Migrated files in the explorer view

The status is only shown on the MDC, but not on the SANergy client cerium. The reason is that this feature is only available on Windows 2000 systems, and cerium is running Windows NT. When we tried to open one of the migrated files on cerium, there was a pause while the file was retrieved transparently from the Tivoli Storage Manager server jamaica down to the SANergy MDC diomede. From there it could be served over the LAN to the SANergy host.

The only indication we have of what is happening is to query the sessions again in the Tivoli Storage Manager administrative client on jamaica. We can see the same session as before, 317 still open on the server. This time it is indicating bytes sent — showing that data is being recalled from the server to DiskXtender to bring back the migrated file.

```

tsm: JAMAICA_SERVER1>q sess

  Sess Comm.  Sess   Wait  Bytes  Bytes Sess  Platform Client Name
  Number Method State   Time   Sent   Recvd Type                -----
    213 Tcp/Ip Run     0 S    9.5 K    108 Admin Win95    ADMIN
    316 Tcp/Ip IdleW  19 S   10.0 K    1.3 K Node DiskXte- OTG
                                nder
    317 Tcp/Ip SendW   0 S   1.9 M   13.6 M Node DiskXte- OTG
                                nder
    318 Tcp/Ip Run     0 S   32.2 K    1.0 K Admin WinNT    ADMIN

tsm: JAMAICA_SERVER1>q sess

  Sess Comm.  Sess   Wait  Bytes  Bytes Sess  Platform Client Name
  Number Method State   Time   Sent   Recvd Type                -----
    213 Tcp/Ip Run     0 S    9.6 K    108 Admin Win95    ADMIN
    316 Tcp/Ip IdleW  26 S   10.0 K    1.3 K Node DiskXte- OTG
                                nder
    317 Tcp/Ip SendW   0 S   3.1 M   13.6 M Node DiskXte- OTG
                                nder
    318 Tcp/Ip Run     0 S   32.9 K    1.0 K Admin WinNT    ADMIN

```

## 8.4 Preparing for LAN-free data transfer

In the configuration up to this point, the files are moved from disk to tape over the LAN. The DiskXtender talks to the API of the Tivoli Storage Manager client on the SANergy MDC, and the API sends the data over the LAN to the Tivoli Storage Manager server.

In a SANergy environment, all the SANergy hosts have direct connectivity (fused) to the data on the shared volume at SAN speed. To also provide SAN speed to the migration process, we have implemented the Tivoli Storage Agent on the MDC. The agent connects to the API, and whenever data is sent from the DiskXtender to the server, the agent redirects the traffic over the SAN to the tape drives. In this way, DiskXtender does LAN-free data transfer of the moved and purged files. The only TCP/IP communication is between the Tivoli Storage Manager server and the API to transfer metadata, and media request information. The data that is recalled from the tape by DiskXtender also goes over the SAN, even if the request comes from a SANergy client.

For this configuration, as opposed to our first configuration which used a LAN path for migration, it is a requirement to have a supported SAN-attached tape library on the Tivoli Storage Manager server. This is because we use a form of SAN tape library sharing between the Storage Agent on the Tivoli Storage Manager client and the server. We used an IBM 3570 tape library, attached to the SAN using the IBM SAN Data Gateway. Information on supported SAN tape libraries for Tivoli Storage Manager is available at these Web sites:

[http://www.tivoli.com/support/storage\\_mgr/san/overview.html](http://www.tivoli.com/support/storage_mgr/san/overview.html)  
[http://www.tivoli.com/support/storage\\_mgr/san/libsharing.html](http://www.tivoli.com/support/storage_mgr/san/libsharing.html)

Tape library sharing in general, as well as the operation of the Tivoli Storage Agent for LAN-free backup, is covered in detail in the redbook *Using Tivoli Storage Manager in a SAN Environment*, SG24-6132.

#### 8.4.1 Installing the Tivoli Storage Agent

The installation of the Storage Agent on the MDC diomede is straightforward. We started the installation by executing the latest install package available at this time and accepting the default settings. This code can be downloaded from this site:

<ftp://ftp.software.ibm.com/storage/tivoli-storage-management/maintenance/server/v4r1/NT/LATEST/>

The package number used was IP22268\_StorageAgent.exe.

The installation path is C:\Program Files\Tivoli\TSM. After the installation, you are asked to reboot the machine for the changes to take affect.

#### 8.4.2 Configuring the Tivoli Storage Agent

After the machine restarted, we opened the dsmsta.opt file in the C:\Program Files\Tivoli\TSM\storageagent\ directory. We entered the following options:

```
commmethod      tcpip
commmethod      namedpipe
devconfig       devconfig.txt
enableallclients yes
```

The first and second entries refer to the communication method of the Storage Agent to the Tivoli Storage Manager server and the API. As the internal communication to the API on the same machine is over shared memory, we must select in addition to `tcpip` also `namedpipes` as communication methods.

The third entry specifies the file `devconfig.txt` where the Storage Agent stores device configurations.

The fourth entry is only necessary because DiskXtender at the time of writing is not yet supported for using LAN-free data transfer. It is intended to provide this support in a future release of Tivoli Storage Manager. To allow testing of this combination, we specified the **enableallclients yes** option. This option is not documented and will only be necessary until the LAN0-free support for DiskXtender is made available.

The Tivoli Storage Manager device driver (`adsm SCSI`) should be running at this time after the reboot of the system. The service is responsible for the communication to the tape drives. To verify this we issued the `net start` command:

```
C:\Documents and Settings\Administrator>net start adsm SCSI
The requested service has already been started.
```

The system returns the information that the driver is running.

### 8.4.3 Communication between server and Storage Agent

The Tivoli Storage Manager server and Storage Agent have to be configured to recognize each other for LAN-free backup processes. The mechanism used for this is server-to-server communication. On the Tivoli Storage Manager server named **JAMAICA\_SERVER1** we have defined the Storage Agent with the name **diomede\_sta** using the `DEFINE SERVER` command. The command was issued in the administrator interface of the server.

```

Session established with server JAMAICA_SERVER1: Windows NT
  Server Version 4, Release 1, Level 3.0
  Server date/time: 04/26/2001 18:22:23 Last access: 04/26/2001 18:13:04

tsm: JAMAICA_SERVER1>define server diomede_sta serverpassword=diomede_sta hladdr
ess=193.1.1.81 lladdress=1500 commethod=tcPIP
ANR1660I Server DIOMEDE_STA defined successfully.

tsm: JAMAICA_SERVER1>

```

Besides the name and password of the Storage Agent, we also had to specify the TCP/IP address (hladdress), the port (lladdress) and the communication method along with this command.

On the Storage Agent side (that is, on diomede) we need a corresponding definition. We ran the `setstorageserver` command in the Storage Agent directory.

```

C:\Program Files\Tivoli\TSM\storageagent>dsmsta setstorageserver myname=diomede_
sta mypassword=diomede_sta servername=jamaica_server1 serverpassword=jamaica_hla
ddress=193.1.1.13 lladdress=1500
ANR0900I Processing options file C:\PROGRA
1\Tivoli\TSM\STORAG
1\dsmsta.opt.
ANR7800I DSMSERV generated at 15:55:08 on Feb 28 2001.

Tivoli Storage Manager for Windows NT
Version 4, Release 1, Level 3.0

Licensed Materials - Property of IBM

5698-TSM (C) Copyright IBM Corporation 1999,2000. All rights reserved.
U.S. Government Users Restricted Rights - Use, duplication or disclosure
restricted by GSA ADP Schedule Contract with IBM Corporation.

ANR1432I Updating device configuration information to defined files.
ANR1433I Device configuration information successfully written to devconfig.txt.
ANR2119I The SERVERNAME option has been changed in the options file.
ANR0467I The setstorageserver command completed successfully.

C:\Program Files\Tivoli\TSM\storageagent>

```

We specified the same name and password for the **myname** and **mypassword** values for the Storage Agent as in the previous definition on the Tivoli Storage Manager server.

#### 8.4.4 Creating the drive mapping

In order to share the tape drives on the SAN between the Tivoli Storage Manager server and the Storage Agent, we have to define the drive mappings on the server. The server must know how the Storage Agent “sees” the drives. The SCSI address of the drives could be different on the Storage Agent and on the server. Therefore we first must display the information that is available from the device driver on the agent machine.

We started the `tsmdlst.exe` program in the installation directory of the Storage Agent on the machine `diomede` our SANergy MDC to get the following output:

```
C:\Program Files\Tivoli\TSM\storageagent>tsmdlst

Computer Name:      DIOMEDE
TSM Device Driver:  Running

TSM Device Name  ID   LUN  Bus  Port  TSM Device Type  Device Identifier
-----
mt1.2.0.2        1    2    0    2     3570             IBM             03570C12
5424
lb1.3.0.2        1    3    0    2     LIBRARY          IBM             03570C12
5424
mt1.4.0.2        1    4    0    2     3570             IBM             03570C12
5424

C:\Program Files\Tivoli\TSM\storageagent>
```

We noticed that the Storage Agent “sees” two tapes and a “LIBRARY” which in fact is the media changer device.

Then we requested the same information from the server by issuing a `query drive` command on the administrative console of the server.

```
tsm: JAMAICA_SERVER1>q drive

Library Name  Drive Name  Device Type  Device           ON LINE
-----
3570LIB       DRIVE0      3570         MT1.4.0.3       Yes
3570LIB       DRIVE1      3570         MT1.2.0.3       Yes

tsm: JAMAICA_SERVER1>
```

Now we can define the drive mapping on the server associating the drive name with the address of the corresponding drive on the Storage Agent. Compare the LUNs of the drives to map the right drives together.

For this Storage Agent we map one of the drives by running the `define drivemapping` command on the administrative interface of the server.

```
tsm: JAMAICA_SERVER1>define drivemapping diomed_sta 3570lib drive0 device=mt1.4.0.2
Session established with server JAMAICA_SERVER1: Windows NT
Server Version 4, Release 1, Level 3.0
Server date/time: 04/26/2001 19:15:27 Last access: 04/26/2001 19:11:29

ANR8916I Drivemapping for drive DRIVE0 in library 3570LIB on storage agent
DIOMEDE_STA defined..
```

#### 8.4.5 Install Storage Agent as a service

The Storage Agent process can now be installed as a service after server to server communication is established. In the Storage Agent installation directory there is a program called `install.exe` that installs the service. We issued the following command on a command prompt on the machine `diomed`.

```
C:\Program Files\Tivoli\TSM\storageagent>install "TSM Storage Agent" "c:\program
files\tivoli\tsm\storageagent\dstasvc.exe"
Service installed

C:\Program Files\Tivoli\TSM\storageagent>
```

After installation of the Storage Agent as a service, we have started it by running the `net start` command on the command prompt on the operating system on `diomed`.

```
C:\Program Files\Tivoli\TSM\storageagent>net start "TSM Storage Agent"
The TSM Storage Agent service is starting.
The TSM Storage Agent service was started successfully.

C:\Program Files\Tivoli\TSM\storageagent>
```

---

## 8.5 LAN-free migration

The scenario is now set up to test LAN-free data migration from DiskXtender. We repeated the migration test from section 8.3, “Installing OTG DiskXtender 2000 on the SANergy MDC” on page 231 and copied some new files to the directory managed by DiskXtender. After the *start watermark* defined for the extended drives was exceeded again, migration started after a drive scan was performed from DiskXtender. To monitor the data transfer we looked at the sessions on both the Tivoli Storage Manager server and the Storage Agent.

The Storage Agent also has an administrative interface that allows us to open an administrative session and query for events or log entries. For this reason we created a simple `dsm.opt` file on `diomede` specifying the local machine, that runs the Storage Agent, as the server address.

```
*dsm.opt file for administrative sessions on the storage agent
tcpserveraddress localhost
```

We then ran the `dsmadm` command that opened an administrative session to the Storage Agent named **diomede\_sta**. In order to monitor the open connection of the Storage Agent we issued the command `query session`. The output shows a connection (session number 10) from the agent to the client node *otg*. The communication method shows as named pipe and the platform is identified as DiskXtender. At the time this screenshot was taken, a data migration was running. You see the moved amount of data so far in this session in the column named **Bytes Recvd**.

```
tsm: DIOMEDE_STA>q sess
```

| Sess Number | Comm. Method | Sess State | Wait Time | Bytes Sent | Bytes Recvd | Sess Type | Platform    | Client Name |
|-------------|--------------|------------|-----------|------------|-------------|-----------|-------------|-------------|
| 1           | Tcp/Ip       | Start      | 0 S       | 7.8 K      | 10.2 K      | Server    |             | DIOMEDE_STA |
| 3           | Tcp/Ip       | Start      | 0 S       | 22.4 K     | 28.4 K      | Server    |             | DIOMEDE_STA |
| 4           | Tcp/Ip       | IdleW      | 0 S       | 5.5 K      | 174         | Server    |             | DIOMEDE_STA |
| 8           | Tcp/Ip       | Run        | 0 S       | 2.7 K      | 108         | Admin     | WinNT       | ADMIN       |
| 9           | Tcp/Ip       | Run        | 0 S       | 33.6 K     | 728         | Admin     | WinNT       | ADMIN       |
| 10          | Named Pipe   | Run        | 0 S       | 895        | 36.3 M      | Node      | DiskXtender | OTG         |
| 11          | Tcp/Ip       | Start      | 0 S       | 414.5 K    | 391.9 K     | Server    |             | DIOMEDE_STA |

```
tsm: DIOMEDE_STA>
```

We also looked at the sessions on the Tivoli Storage Manager server at the same time.

```
tsm: JAMAICA_SERVER1>q sess
```

| Sess Number | Comm. Method | Sess State | Wait Time | Bytes Sent | Bytes Recvd | Sess Type | Platform    | Client Name |
|-------------|--------------|------------|-----------|------------|-------------|-----------|-------------|-------------|
| 29          | Tcp/Ip       | IdleW      | 3.8 M     | 10.2 K     | 7.8 K       | Server    | Windows NT  | DIOMEDE_STA |
| 31          | Tcp/Ip       | IdleW      | 29 S      | 29.1 K     | 22.6 K      | Server    | Windows NT  | DIOMEDE_STA |
| 32          | Tcp/Ip       | IdleW      | 1.2 M     | 174        | 5.5 K       | Server    | Windows NT  | DIOMEDE_STA |
| 34          | Tcp/Ip       | IdleW      | 49 S      | 17.7 K     | 3.1 K       | Node      | DiskXtender | OTG         |
| 35          | Tcp/Ip       | IdleW      | 1 S       | 233.5 K    | 51.0 K      | Node      | DiskXtender | OTG         |
| 36          | Tcp/Ip       | Run        | 0 S       | 6.8 K      | 244         | Admin     | WinNT       | ADMIN       |
| 37          | Tcp/Ip       | IdleW      | 1 S       | 646.8 K    | 684.4 K     | Server    | Windows NT  | DIOMEDE_STA |

```
tsm: JAMAICA_SERVER1>
```

From this output we see two active sessions from the node *otg*. But neither of them is transferring large quantities of data. These sessions are for metadata transfer over TCP/IP.

The migration is thus occurring LAN-free. For further verification you also can check the ports on the Fibre Channel switches, or use the tools shipped with the fabric to track dataflow.

---

## Appendix A. Linux with a SAN and SANergy

In this appendix we present various topics intended to help you work with Linux on a SAN and with SANergy.

Linux is a version of the UNIX operating system that is maintained by different user community projects, all over the world. Linux, like many other projects around the world such as Apache and Samba, takes advantage of developers, testers, and other people to create powerful and useful tools for general use.

These tools follow strict development and licensing rules to maintain the quality and value of the products while protecting the intellectual property of the projects from being used inappropriately. For example, anybody is free to run, copy, modify and distribute modified versions of programs protected by this type of license (the most important of which is the GNU General Public License, or GNU GPL) but they are not allowed to add restrictions of their own to the program.

What this means, in practical terms, is that experienced volunteers can work together to build a high quality product available to anyone freely, without worrying that their work will be hijacked by someone else to create a proprietary and closed product. The open nature of this type of development has many advantages, but the discussion of them is beyond the scope of this Redbook. For more information, see the GNU Project Web site:

[www.gnu.org](http://www.gnu.org)

Although Linux and the other projects mentioned are freely available, several companies create *distributions* of Linux. These distributions contain Linux and many other components (for example, Apache, Samba, XFree86) that the company has integrated and tested. Therefore, a customer can purchase a distribution without worrying that there are missing components or incompatibilities. Also, the companies that produce these distributions may provide support services, should a customer wish it. The distribution of Linux we used in this redbook is produced by RedHat, and we used both v6.2 and v7.0.

---

### A.1 Linux on a SAN

Operating Linux hosts on a SAN environment is not greatly different than normal Linux operations, but there are a few pieces of information that may make your life a little easier.

### A.1.1 Helpful Web sites

The current Linux Fibre Channel “How To” documentation can be found at:

[http://www.sistina.com/gfs/howtos/fibrechannel\\_howto/](http://www.sistina.com/gfs/howtos/fibrechannel_howto/)

Sistina is also currently responsible for some of the Linux storage management projects, specifically the Global File System (GFS) and Logical Volume Manager (LVM) projects.

The University of Minnesota Fibre Channel group has many good resources to assist in all aspects of running Linux on SAN. They can be found at:

<http://www.borg.umn.edu/fc/>

Both the Sistina and University of Minnesota sites contain a wealth of information on all SAN and Fibre Channel related topics.

### A.1.2 Linux devices

When Linux loads the driver for a SCSI device (such as a Fibre Channel HBA), it will associate a device file with each SCSI disk found. These device files are found in the `/dev` subdirectory and follow the naming convention of `/dev/sd*`, where `*` is a letter representing the specific devices, starting with `a`. For example, the first SCSI drive found will be `/dev/sda`, the second will be `/dev/sdb`, and so on.

The partitions on each of these disks are denoted by the device files `/dev/sd*#`, where `#` is the number of the partition on the disk. Partition 1 on the first SCSI disk is `/dev/sda1`.

When a SCSI device driver finds devices, it will place a list of those found in the file `/proc/scsi/scsi`. Issue the concatenate (`cat`) command to display the contents of that file:

```
[root@rainier /root]# cat /proc/scsi/scsi
Attached devices:
Host: scsi0 Channel: 00 Id: 00 Lun: 00
  Vendor: DEC      Model: HSG80      Rev: V85S
  Type:   Direct-Access      ANSI SCSI revision: 02
Host: scsi0 Channel: 00 Id: 01 Lun: 00
  Vendor: DEC      Model: HSG80      Rev: V85S
  Type:   Direct-Access      ANSI SCSI revision: 02
Host: scsi0 Channel: 00 Id: 01 Lun: 01
  Vendor: DEC      Model: HSG80      Rev: V85S
  Type:   Direct-Access      ANSI SCSI revision: 02
```

### A.1.3 Linux device drivers for Fibre Channel HBAs

As with all other add-on adapter cards, a Fibre Channel HBA will require a device driver. These are distributed by the vendors (certain Qlogic card drivers are included in the current Linux kernel builds). The installation of the drivers varies significantly between vendors, so follow their instructions.

Device drivers on current kernels are distributed as modules. A module can be loaded when needed, or it can be compiled directly into the kernel. It is generally recommended that HBA drivers be kept as loadable modules. This allows you to unload and reload at will, rather than having to reboot.

The `lsmod` command allows you to display the modules currently loaded.

```
[root@rainier /root]# lsmod
Module                Size  Used by
ide-cd                 23628  0 (autoclean)
qla2x00               173816  17
sanergy                7912   1
nfsd                  143844  8 (autoclean)
nfs                    28768   1 (autoclean)
lockd                  31176   1 (autoclean) [nfsd nfs]
sunrpc                 52964   1 (autoclean) [nfsd nfs lockd]
olympic                13904   1 (autoclean)
eeepro100             16180   1 (autoclean)
agpgart                18600   0 (unused)
usb-uhci               19052   0 (unused)
usbcore                42088   1 [usb-uhci]
```

Note the device driver module (`qla2x00`) in the preceding example. If a change is made to the configuration of your SAN, you will need to remove and re-insert the module into the current environment. The `rmmod` and `insmod` commands provide these functions.

```
[root@rainier /root]# rmmod qla2x00
[root@rainier /root]# insmod qla2x00
Using /lib/modules/2.2.16-22/scsi/qla2x00.o
```

For a more detailed discussion of `lsmod`, `rmmod` and `insmod`, please see their corresponding man pages.

Typically you will want to have the device driver module loaded at boot time. The installation program or instructions that came with the HBA should give the instructions for this configuration. We provide a brief set of instructions, should you ever need to do this manually.

Place the device driver in the appropriate directory. This is almost certainly `/lib/modules/kernel-version/scsi`, where `kernel-version` is the actual version number of your kernel. For example, our device driver module was in `/lib/modules/2.2.216-22/scsi`.

Next, make an alias entry in `/etc/conf.modules` or `/etc/modules.conf`, depending upon which file your kernel uses.

```
[root@rainier /root]# cat /etc/modules.conf
[alias eth0 eeepro100
alias tr0 olympic
alias scsi_hostadapter qla2x00
alias parport_lowlevel parport_pc
alias usb-controller usb-uhci
```

The preceding screen snapshot shows our alias entry in `/etc/modules.conf` for the `qla2x00` module.

Make a bootable image that includes the module. The following command makes a new bootable image `/boot/linux_w_qlogic` on a system running the 2.2.16-22 version of the Linux kernel.

```
[root@rainier /root]# mkinitrd /boot/linux_w_qlogic 2.2.16-22[
```

Add a boot entry for the new bootable image. This is done by editing the `/etc/lilo.conf` file and then executing `lilo` to update the boot-loading program. The screenshot shows an example `lilo.conf` that has the original boot entry, as well as a new one with the HBA driver module.

```
[root@rainier /root]# cat /etc/lilo.conf
boot=/dev/hda1
map=/boot/map
install=/boot/boot.b
prompt
timeout=50
message=/boot/message
linear
default=linux_w_qlogic

image=/boot/vmlinuz-2.2.16-22
    label=linux
    initrd=/boot/initrd-2.2.16-22.img
    read-only
    root=/dev/hda5
image=/boot/vmlinuz-2.2.16-22
    label=linux_w_qlogic
```

After making the entry in `/etc/lilo.conf`, run `lilo -t -v` to validate the file. If no errors are reported, run `lilo` with no parameters to update the boot loader.

```
[root@rainier /root]# lilo -t -v
LILO version 21, Copyright 1992-1998 Werner Almesberger
Reading boot sector from /dev/hda1
Merging with /boot/boot.b
Boot image: /boot/vmlinuz-2.2.16-22
Added linux
Boot image: /boot/vmlinuz-2.2.16-22
Mapping RAM disk /boot/linux_w_qlogic
Added linux_w_qlogic *
```

After this is completed, the device module will be loaded at the next boot. As always, the module can be loaded manually using `insmod`.

#### A.1.4 Accessing the SCSI disks

At this point, you have hopefully attached the Linux machine to the SAN and installed and loaded device driver module for the HBA. For some configurations, at the present time, Linux requires that the SAN attached disks begin at LUN 0 and be sequential. So, you will successfully access the disks if they have LUNs 0, 1, and 2, but not if they have LUNs 0, 3, and 5.

Before configuring your SAN, check with the documentation for your HBA and Linux distribution to find out if sequentially ordered LUNs beginning at 0 are required.

Assuming that you have successfully loaded your device module and that there are now entries for SAN-attached disks in `/proc/scsi/scsi`, you are now ready to access them. If you need to create a file system, see the man pages for `mke2fs`.

If the disk is *not* to be used with SANergy, but just mounted as any other SCSI disk, mount it using its `/dev/sd*#` device file. For example:

```
[root@rainier /root]# mount /dev/sdc1 /mnt/disk1
```

See the man pages for the `mount` command and the `/etc/fstab` file for details on the mount commands parameters and how to have a file system automatically mounted at boot time. You can also issue the `mount` command without any parameters to see what file systems are currently mounted.

### A.1.5 SANergy and the SCSI disks

If the disk is to be used with SANergy, it is important to know if and how you need to mount them.

#### A.1.5.1 Accessing a disk as a SANergy host

If the disk is owned by another SANergy machine (as its MDC), then you do not need to mount the disk directly. In fact, it is important you do not, as data corruption may occur. When you will be sharing the volume as a SANergy host (a machine that shares a volume but does not own it), then you will need to mount the NFS share exported from the MDC. If you have mounted an NFS export from a SANergy MDC and both of you have access to the appropriate disk, you will be able to have fused I/O, via SANergy.

A typical NFS mount command may look like this:

```
[root@rainier /root]# mount -t nfs aldan:/G /G
```

This command mounts the share `/G` owned by the MDC `aldan` at the mountpoint `/G` on the Linux SANergy host.

Refer to the *mount* and */etc/fstab* man pages for more details, for example, on how to automatically mount NFS shares at boot time.

After following these steps, you should now be able to have fused I/O to that volume. Start the SANergy configuration tool by issuing */usr/SANergy/config*. This will start the SANergy GUI using Netscape. Do a performance test to the mounted NFS share to validate that you can fuse.

#### **A.1.5.2 Accessing a disk as a SANergy MDC**

Accessing a disk as a SANergy MDC is only slightly different from accessing a disk that is not shared. Let us take the example of mounting the first partition on a non-shared SCSI disk, which is the third SCSI drive found. You could mount the volume using this command:

```
[root@rainier /root]# mount /dev/sdcl /mountpoint
```

When you install SANergy, the device files on those busses you elect to manage will no longer work. SANergy prevents you from accessing these shared disks using normal device files. New device files are created by SANergy and are stored in the directory */dev/SANergyCDev/*. So, to mount these drives, you would simply use the device files in that directory (the actual device files have the same name, just a different directory). To mount the same device that is now being managed by SANergy, use the command:

```
[root@rainier /root]# mount /dev/SANergyCDev/sdcl /mountpoint
```

---

## **A.2 Using SANergy on Linux**

At this point, you should have access to the SCSI disks, either normally or as a SANergy host or MDC. So, how do you actually put SANergy to work?

On UNIX, when exploiting fused access to shared volumes, you will have to set specific environment variables that direct applications to use the SANergy library for I/O operations. In Linux, for example, you will set and export an environment variable named *LD\_PRELOAD* with the value */lib/libSANergy.so*. If you were to issue the commands at a shell prompt, they would be:

```
[root@rainier /root]# LD_PRELOAD=/lib/libSANergy.so
export LD_PRELOAD
```

You could validate that the variable was set by issuing the *env* command to see all environment variables of this session, or display the contents of the specific *LD\_PRELOAD* variable by typing:

```
[root@rainier /root]# echo $ LD_PRELOAD
/lib/libSANergy.so
```

If you are running as a SANergy host, you can also validate fused I/O by setting the environment variables and copying files to or from a fused volume. You should see the fused I/O counts increment as the copy takes place.

Setting the environment variables manually is, obviously, not sufficient to most applications. To get fused I/O when using an application, you can either set the variables universally on this system, or set them when invoking the application itself via the provided startup scripts (see 2.3.2.3, “Using SANergy with a process” on page 55).

To set an environment variable universally, edit */etc/profile* and add the same commands used as before. This will ensure that the variable is set whenever the Linux operating systems itself starts.

Although it is easier to set it universally, it may be more appropriate to set the variables for specific applications. Many applications, such as Samba and Apache, are started by Linux using scripts located in the */etc/rc.d/init.d* subdirectory.

You can manually stop or start these services by invoking those same scripts with the *start* or *stop* parameters. For example, to stop Apache, you would issue this command:

```
[root@rainier /root]# etc/rc.d/init.d/httpd stop
```

The script to start/stop Samba is `/etc/rc.d/init.d/smb`, and for the NFS server, it is `/etc/rc.d/init.d/nfs`.

Since these services are started as scripts, we can edit them and insert the lines that set the environment variables. For example, in 6.4, “Web serving” on page 179 we required Apache to access data stored on a shared SANergy volume, and we wished it to be able to do so over the SAN (fused). We added these lines to `/etc/rc.d/init.d/httpd`:

```
[#!/bin/sh
#
# Startup script for the Apache Web Server
# Tivoli testing - set environment variable for SANergy
LD_PRELOAD=/lib/libSANergy.so
export LD_PRELOAD
```

Next, we stopped and started Apache using by entering these commands:

```
root@rainier /root]# /etc/rc.d/init.d/httpd stop
root@rainier /root]# /etc/rc.d/init.d/httpd start
```

We then monitored the SANergy statistics using the SANergy GUI while the Web site was being accessed. to verify that fused access was occurring.

Some applications that you wish to have fused access may not be started by a script, but are simply started as an executable program. In these cases, you will need to create a script that sets the environment variables and then invokes the program (assuming they are not set universally, already).

There is one important point that we need to make about using SANergy with Linux. In order to gain fused access to a SANergy volume, an application must have the authority to access the SANergy device files that are doing the fused access.

If you are running as a SANergy host to the volume in question, access is done via the devices named `/dev/SANergyCDev/raw*`. As discussed earlier, if you are running as an MDC for this volume, you will use the device file corresponding to the appropriate SCSI disk, but located in the directory `/dev/SANergyCDev` (for example, `/dev/SANergy/CDev/sda1`). You can list the owner, group, and read/write permissions for the device files by issuing the command:

```
root@rainier /root]# ls -al /dev/SANergy/CDev/
total 108
drwxr-xr-x  2 root  root      8192 Apr 20 21:11 .
drwxr-xr-x 13 root  root    98304 Apr 20 19:10 ..
crwxrwxrwx  1 root  root    162,  0 Apr  9 15:54 raw
crwxrwxrwx  1 root  root    162,  1 Apr 20 19:10 raw1
crwxrwxrwx  1 root  root    162,  2 Apr 20 19:10 raw2
crwxrwxrwx  1 root  root    162,  3 Apr 20 21:11 raw3
brwxrwxrwx  1 root  disk     8,  0 Aug 24  2000 sda
brwxrwxrwx  1 root  disk     8,  1 Aug 24  2000 sda1
brwxrwxrwx  1 root  disk     8, 10 Aug 24  2000 sda10
brwxrwxrwx  1 root  disk     8, 11 Aug 24  2000 sda11
brwxrwxrwx  1 root  disk     8, 12 Aug 24  2000 sda12
brwxrwxrwx  1 root  disk     8, 13 Aug 24  2000 sda13
brwxrwxrwx  1 root  disk     8, 14 Aug 24  2000 sda14
brwxrwxrwx  1 root  disk     8, 15 Aug 24  2000 sda15
brwxrwxrwx  1 root  disk     8,  2 Aug 24  2000 sda2
(etc)
```

You can use the `chmod`, `chown`, and `chgrp` commands to manipulate the access controls on device files, just like any other file. See the man pages for these commands for more information on setting UNIX file security.

When we implemented Apache on Linux and wished it to have fused I/O to the shared volume, we had to change the permissions on the `raw*` device drivers to enable the http daemon to have access (this machine was accessing the shared volume as a host, therefore we were using the raw devices). We issued the command:

```
root@rainier /root]# chmod +777 /dev/SANergy/CDev/raw*
```

We then validated the fused I/O by monitoring the SANergy statistics while Web accesses were occurring.

---

## Appendix B. Windows server drive mapping and SANergy

This appendix covers a possible solution to the issue raised in “Tivoli Storage Manager on Windows — considerations” on page 207. We have discussed a scenario where a SANergy mapped volume was defined as a destination device class for a backup set.

Use of mapped drives, also known as shares, by a Windows Tivoli Storage Manager server service is currently dependent on the drive having been mapped before the server tries to use it. Also, it must be mapped to the actual service and not just to a user logon session, which is normally the case. In other words, logging on to a Windows user such as Administrator and mapping the drive in question will not give the Tivoli Storage Manager server service the correct access to that drive. At the time of writing this redbook, the Tivoli Storage Manager server does not support access to shares via its *Universal Naming Convention* (UNC) identifier.

This appendix covers the detailed considerations of using mapped drives and one way to allow a Tivoli Storage Manager server service to use mapped, and therefore SANergy, drives. Unless otherwise indicated, we use the term *server* to mean Tivoli Storage Manager Windows server running as a service. This is the normal way a Tivoli Storage Manager server would run on Windows. It is possible to run the server as a session or in a window of a logged-on user, otherwise referred to as running in the foreground. However, this discussion does not address that mode of operation.

Our configuration used a Windows2000 Tivoli Storage Manager server, jamaica, which mapped a network drive via SANergy from the MDC called aldan.

---

### B.1 Detailed considerations

This section covers in detail some of the considerations for using a mapped drive or share by a Tivoli Storage Manager server.

### **B.1.1 Access to the drive**

It is necessary but not sufficient to specify a service Log On account which can perform the drive mapping. The account, or user id may be that of a local user, or it may be a domain user, with permissions to access the remote drive, that is, to issue the appropriate Windows `net use` command.

The actual drive mapping could be accomplished with a `net use` command invoked by the TSM server itself. Unfortunately, there is currently no method for directly invoking Windows operating system commands from the server, since mechanisms such as administrative schedules or server scripts do not have this ability. Therefore we are proposing an alternative way in B.2, “A procedure for mapping a drive to a server” on page 273.

Once the mapping has been accomplished, SANergy will take care of fusing the drive, assuming SANergy setup has been done properly for the drive. As we have previously recommended, the use of SANergy should be tested on the server system independently of Tivoli Storage Manager to ensure it functions correctly.

### **B.1.2 What type of share?**

The drive to be mapped, that is, the share, must have an explicitly defined share name. Use of the default or predefined administrative share will not work. An example of an administrative share is C\$ for the C drive. This is a requirement of SANergy, and the definition of a proper share should be done as part of the normal SANergy MDC setup. We used our explicitly defined share `\part1`.

### **B.1.3 When to map drives**

Shares, including SANergy volumes, to be used by the server should be mapped as soon as possible after the server is started so that any activity that needs the drives has access to it. In our case, we initiate drive mapping as soon as the server indicates it is up and running.

### **B.1.4 Connection persistence**

The connection to the mapped drive is maintained through the use of the `/persistent` flag on the `net use` command. This will ensure that the drive will get remapped to the server even after a lost connection, such as a reboot of the MDC. A SANergy volume is unusable during an MDC reboot and until the remapping is complete on the server.

### B.1.5 Server access to the mapped drive

When making a reference to a drive or drive with subdirectory in a Tivoli Storage Manager device class definition, the server will allow references to unknown drives but not to unknown subdirectories. In the first example below, the device class definition worked even though there was no I: drive, either local or mapped.

```
tsm: JAMAICA_SERVER1>define devclass sanbackupbad devtype=file maxcap=1g dir="i:"  
ANR2203I Device class SANBACKUPBAD defined.  
tsm: JAMAICA_SERVER1>
```

However, the next device class definition example fails. The drive letter exists and other subdirectories on the drive have been used successfully, but the subdirectory referred to in the define command does not exist.

```
tsm: JAMAICA_SERVER1>define devclass sanbackupbad devtype=file maxcap=1g dir="g:  
\tsmbaddir"  
ANR2020E DEFINE DEVCLASS: Invalid parameter - G:\TSMBADDIR.  
ANS8001I Return code 3.  
tsm: JAMAICA_SERVER1>
```

This error is an indication that either the subdirectory does not exist, that the drive on which the subdirectory resides is not accessible, or both.

---

## B.2 A procedure for mapping a drive to a server

We exploited the Tivoli Storage Manager event logging and user exit facilities to perform the drive mapping. Execution of the required net use command was done from a DOS batch file invoked by a server *Dynamic-Link Library* (DLL) user exit.

Following are the detailed steps to get a drive mapped to a Windows Tivoli Storage Manager server and therefore make a SANergy drive usable to it.

1. Perform normal SANergy installation and setup for the necessary drives. SANergy fusing should be tested outside of Tivoli Storage Manager.
2. Modify the server service to use a log on account which has permission to issue the `net use` command. We used the Administrator account. The default service log-on account, System, does not allow execution of the `net use` command. Modification of the log on account can be accomplished by accessing the Windows Services and changing the log on property of the server service to be changed. On Windows 2000, these steps can be used: **Start->Programs->Administrative Tools->Services**. Scroll down to the TSM server, right click on it, then select **Properties** from the pull down menu. Click on the **Log On** tab and finally select **This account** and enter the log on account. Figure 199 shows the panel after modification for our TSM server on jamaica.



Figure 199. Tivoli Storage Manager service log on properties

3. Create the batch file which will execute the `net use` command. We used `tsminit.bat` which was called by our user exit. We placed the batch file in `C:\WINNT\SYSTEM32` to make it executable generally. The batch file would contain a statement of this general form:

```
net use z: \\<MDC_host_name>\share_name password /user:username  
/persistent:yes
```

Our net use command was:

```
net use g: \\aldan\part1 sanuser1 /user:sanuser1 /persistent:yes
```

We were not using a domain controller so used an `aldan` logon account instead. The password and user parameters belong to a user with correct permissions on the system which owns the drive — in our case, the SANergy MDC.

4. Create the DLL to be used by the server user exit. Ours was called `userexit.dll`. The sample source code used to create our user exit is listed in B.3, “Sample source listings” on page 275. Modify the source as required, create the `.dll`, and move it into the same directory where your server option file resides. In our case that was:

```
c:\program files\Tivoli\TSM\server1
```

5. Add the following statement to the `dsmserv.opt` file for the server. This will cause logging to the user exit receiver to start whenever the server is restarted. It will also cause the routine `SANSYNCHEXIT` in the DLL `USEREXIT` to be invoked for each enabled event.

```
userexit yes USEREXIT SANSYNCHEXIT
```

6. Select a server event to trigger the execution of the BAT file, and enable that event on the server. We chose the message `ANR0916`, which is issued when the server is up and ready to use. We set this up by issuing the administrator command:

```
ENable EEvent USEREXIT ANR0916
```

7. Restart the server to recognize the configuration changes just made.

---

### B.3 Sample source listings

Samples of the source code used for our `userexit.dll` are listed below. Please consult your program development documentation for information on creation of a DLL. You can download softcopy of this source code by following the instructions in Appendix D, “Using the additional material” on page 289.

### B.3.1 userexit.c listing

```
/******  
* ADSTAR Distributed Storage Manager (adsm) *  
* Server Component *  
* * *  
* 5639-B9300 (C) Copyright IBM Corporation 1997 (Unpublished) *  
*****/  
/*****  
* Name: USEREXITSAMPLE.C *  
* * *  
* Description: Example user-exit program that is invoked by *  
* the ADSM V3 Server *  
* * *  
* Environment: ***** *  
* ** This is a platform-specific source file ** *  
* ** versioned for: "WINDOWS NT/2000" ** *  
* ***** *  
* To setup: *  
* 1) in dsmserv.opt add the following line *  
* userexit yes userexit sanSynchExit *  
* 2) From admin command line, enter the following *  
* enable event userexit ANR0916 *  
* To start: *  
* 1) Enter the following ADMIN commands *  
* begin eventlogging userexit *  
* * *  
*****/  
// #include <wincrypt.h>  
#include <stdio.h>  
#include <stdlib.h>  
#include <sys/types.h>  
#include <process.h>  
#include <io.h>  
#include <windows.h>  
#include "USEREXITSAMPLE.H"  
DWORD TlsIndex;  
  
/*****  
*** Do not modify below this line. ***  
*****/  
#define DllExport __declspec(dllexport)  
/*****  
* Procedure: sanSynchExit *  
* If the user-exit is specified on the server, a valid and *  
* appropriate event will cause an elEventRecvData structure *  
* (see USEREXITSAMPLE.H) to be passed to a procedure named *  
* sanSynchExit that returns a void. *  
* * *  
* This procedure can be named differently: *  
* ----- *  
* The procedure name must match the function name specified in *  
* the server options file (4th arg). The DLL name generated from *  
* this module must also match in the server options file *  
* (3rd arg). *  
* * *  
*****
```

```

* INPUT : A (void *) to the elEventRecvData structure
* RETURNS: Nothing
*****/
DllExport void SANSYNCHEXIT( void *anEvent )
{
    /* Typecast the event data passed */
    LPVOID lpMsgBuf;
    elEventRecvData *eventData = (elEventRecvData *)anEvent;
    /*****
    *** Do not modify above this line. ***
    *****/
    THREADINFO *pti;
    int irc;

    pti = TlsGetValue(TlsIndex);
    if (( eventData->eventNum == USEREXIT_END_EVENTINUM ) ||
        ( eventData->eventNum == END_ALL_RECEIVER_EVENTINUM ) )
    {
        /* Server says to end this user-exit. Perform any cleanup, */
        /* but do NOT exit() !!! */
        if (pti->bFileOpen)
        {
            fprintf(pti->UserExitLog,
                "sanSynchExit: Closing UserExitLog %0.2d:%0.2d:%0.2d %0.2d/%0.2d/%0.4d\n",
                (int)eventData->timeStamp.hour,
                (int)eventData->timeStamp.min,
                (int)eventData->timeStamp.sec,
                (int)eventData->timeStamp.mon,
                (int)eventData->timeStamp.day,
                ((int)eventData->timeStamp.year + 1900));
            fflush(pti->UserExitLog);
            fclose(pti->UserExitLog);
            pti->bFileOpen = FALSE;
            pti->bInitComplete = FALSE;
        }
        return;
    }
    if (!pti->bInitComplete)
    {
        /* if initialization is not yet complete*/
        /* then initialize pti structure and then*/
        /* open the userlog file*/
        sprintf(pti->szFileName, "SanExitLog.TEXT");
        pti->UserExitLog = fopen(pti->szFileName, "a");
        if (pti->UserExitLog == NULL)
        {
            pti->bFileOpen = FALSE;
        }
        else
    }
}

```

```

{
    pti->bFileOpen = TRUE;
    fprintf(pti->UserExitLog,
        "sanSynchExit: Opening %s %0.2d:%0.2d:%0.2d %0.2d/%0.2d/%0.4d\n",
        pti->szFileName,
        (int)eventData->timeStamp.hour,
        (int)eventData->timeStamp.min,
        (int)eventData->timeStamp.sec,
        (int)eventData->timeStamp.mon,
        (int)eventData->timeStamp.day,
        ((int)eventData->timeStamp.year + 1900));

    fprintf(pti->UserExitLog, "=====\n");
}
    pti->bInitComplete = TRUE;
} /* end of initialization*/

if (eventData->eventType == TSM_SERVER_EVENT)
{
    if (eventData->eventNum == 916)
    {
        fprintf(pti->UserExitLog,
            "TSM %s ready for operations at %0.2d:%0.2d:%0.2d %0.2d/%0.2d/%0.4d\n",
            eventData->serverName,
            (int)eventData->timeStamp.hour,
            (int)eventData->timeStamp.min,
            (int)eventData->timeStamp.sec,
            (int)eventData->timeStamp.mon,
            (int)eventData->timeStamp.day,
            ((int)eventData->timeStamp.year + 1900));

        iRC = (int)system("tsminit.bat");
        // fprintf(pti->UserExitLog, "TsmInit.bat execution - return code: %d\n", iRC);
        if (iRC != 0)
        {
            pti->dwLastError = GetLastError();
            FormatMessage(FORMAT_MESSAGE_ALLOCATE_BUFFER |
                FORMAT_MESSAGE_FROM_SYSTEM |
                FORMAT_MESSAGE_IGNORE_INSERTS,
                NULL,
                pti->dwLastError,
                MAKELANGID(LANG_NEUTRAL, SUBLANG_DEFAULT), // Default language
                (LPTSTR) &lpMsgBuf,
                0,
                NULL);
            fprintf(pti->UserExitLog, "\"system\" execution error (%ld):
%s\n", pti->dwLastError, lpMsgBuf);
        }
        else
        {
            fprintf(pti->UserExitLog, "TsmInit.bat execution successful\n");
        }
        fflush(pti->UserExitLog);
    }
}
return; /* For picky compilers */
} /* End of sanSynchExit() */

```

```

BOOL WINAPI DllMain(
    HMODULE hmod,
    DWORD Reason,
    LPVOID lpvReserved)

{
    THREADINFO *pti;
    UNREFERENCED_PARAMETER(hmod);

    switch (Reason)
    {
    case DLL_PROCESS_ATTACH:
        // fprintf(stderr, "sanSynchExit: DLL_PROCESS_ATTACH\n");
        TlsIndex = TlsAlloc();

    case DLL_THREAD_ATTACH:
        // fprintf(stderr, "sanSynchExit: DLL_THREAD_ATTACH\n");
        pti = (THREADINFO *)LocalAlloc(LPTR, sizeof(THREADINFO));
        if (!pti)
        {
            fprintf(stderr, "sanSynchExit: Thread Attach Failure\n");
            return(FALSE);
        }
        TlsSetValue(TlsIndex, pti);
        pti->bInitComplete = FALSE;
        pti->bFileOpen = FALSE;
        break;

    case DLL_THREAD_DETACH:
        // fprintf(stderr, "sanSynchExit: DLL_THREAD_DETACH\n");
    case DLL_PROCESS_DETACH:
        pti = TlsGetValue(TlsIndex);
        LocalFree(pti);
        if (Reason == DLL_PROCESS_DETACH)
        {
            // fprintf(stderr, "sanSynchExit: DLL_PROCESS_DETACH\n");
            TlsFree(TlsIndex);
        }
        break;
    default:
        break;
    }
    return TRUE;
}

```

## B.3.2 userexit.h listing

```
/******  
 * Name:          USEREXITSAMPLE.H  
 * Description:   Declarations for a user-exit  
 * Environment:   WINDOWS NT/2000  
******/  
  
#ifndef _H_USEREXITSAMPLE  
#define _H_USEREXITSAMPLE  
  
#include <stdio.h>  
#include <sys/types.h>  
  
/**** Do not modify below this line ****/  
  
#define BASE_YEAR      1900  
  
typedef short          int16;  
typedef int            int32;  
  
#ifndef uchar  
typedef unsigned char  uchar;  
#endif  
/* DateTime Structure Definitions - TSM representation of a timestamp */  
  
typedef struct  
{  
    uchar year;        /* Years since BASE_YEAR (0-255)*/  
    uchar mon;         /* Month (1 - 12) */  
    uchar day;         /* Day (1 - 31) */  
    uchar hour;        /* Hour (0 - 23) */  
    uchar min;         /* Minutes (0 - 59)*/  
    uchar sec;         /* Seconds (0 - 59)*/  
} DateTime;  
  
/*****  
 * Some field size definitions (in bytes) *  
******/  
  
#define MAX_SERVERNAME_LENGTH    64  
#define MAX_NODE_LENGTH          64  
#define MAX_COMMNAME_LENGTH      16  
#define MAX_OWNER_LENGTH         64  
#define MAX_HL_ADDRESS           64  
#define MAX_LL_ADDRESS           32  
#define MAX_SCHED_LENGTH         30  
#define MAX_DOMAIN_LENGTH        30  
#define MAX_MSGTEXT_LENGTH       1600  
  
/*****  
 * Event Types (in eEventRecvData.eventType) *  
******/  
  
#define TSM_SERVER_EVENT          0x03 /* Server Events */  
#define TSM_CLIENT_EVENT         0x05 /* Client Events */
```

```

/*****
 * Application Types (in elEventRecvData.applType) *
 *****/

#define TSM_APPL_BACKARCH 1 /* Backup or Archive client */
#define TSM_APPL_HSM 2 /* Space manage client */
#define TSM_APPL_API 3 /* API client */
#define TSM_APPL_SERVER 4 /* Server (ie. server to server) */

/*****
 * Event Severity Codes (in elEventRecvData.sevCode) *
 *****/

#define TSM_SEV_INFO 0x02 /* Informational message. */
#define TSM_SEV_WARNING 0x03 /* Warning message. */
#define TSM_SEV_ERROR 0x04 /* Error message. */
#define TSM_SEV_SEVERE 0x05 /* Severe error message. */
#define TSM_SEV_DIAGNOSTIC 0x06 /* Diagnostic message. */
#define TSM_SEV_TEXT 0x07 /* Text message. */

/*****
 * Data Structure of Event that is passed to the User-Exit. *
 * The same structure is used for a file receiver *
 *****/

typedef struct evRdata
{
    int32 eventNum; /* the event number. */
    int16 sevCode; /* event severity. */
    int16 applType; /* application type (hsm, api, etc) */
    int32 sessId; /* session number */
    int32 version; /* Version of this structure (1) */
    int32 eventType; /* event type
                     * (TSM_CLIENT_EVENT, TSM_SERVER_EVENT) */
    DateTime timeStamp; /* timestamp for event data. */
    uchar serverName[MAX_SERVERNAME_LENGTH+1]; /* server name */
    uchar nodeName[MAX_NODE_LENGTH+1]; /* Node name for session */
    uchar commMethod[MAX_COMMNAME_LENGTH+1]; /* communication method */
    uchar ownerName[MAX_OWNER_LENGTH+1]; /* owner */
    uchar hlAddress[MAX_HL_ADDRESS+1]; /* high-level address */
    uchar llAddress[MAX_LL_ADDRESS+1]; /* low-level address */
    uchar schedName[MAX_SCHED_LENGTH+1]; /* schedule name if applicable */
    uchar domainName[MAX_DOMAIN_LENGTH+1]; /* domain name for node */
    uchar event[MAX_MSGTEXT_LENGTH]; /* event text */
} elEventRecvData;

/*****
 * Size of the Event data structure *
 *****/

#define ELEVENTRECVDATA_SIZE sizeof(elEventRecvData)

/*****
 * User Exit EventNumber for Exiting *
 *****/

```

```

#define USEREXIT_END_EVENTINUM 1822 /* Only user-exit receiver to exit */
#define END_ALL_RECEIVER_EVENTINUM 1823 /* All receivers told to exit */

/*****
*** Do not modify above this line. ***
*****/

/***** Additional Declarations *****/
#define MAX_SESSIONS 50
typedef struct {
    PVOID pNextEvent;
    elEventRecvData elEventContent;
    } elEventRecord;
typedef struct {
// int iSessCount;
    BOOL bInitComplete;
    BOOL bFileOpen;
// int iYearOpen;
// int iMonOpen;
// int iDayOpen;
    DWORD dwLastError;
    char szFileName[20];
    FILE *UserExitLog;
    } THREADINFO;

// void StartSession(THREADINFO *pti, int sessNo, DateTime StartDateTime);
// void EndSession(THREADINFO *pti, int sessNo, DateTime EndDateTime);
// void AddSession(THREADINFO *pti, elEventRecvData *eventData);

endif

```

---

## Appendix C. Using ZOOM to improve small file I/O performance

ZOOM is a feature introduced with SANergy Version 2.2 that allows certain types of small file operations to operate faster than the base SANergy. The following information is extracted and adapted from a Technical Note written by the SANergy development team.

---

### C.1 Background

The standard SANergy base product is well designed for systems that tend to use a high proportion of large files. With larger files, a higher performance should be achieved. ZOOM is a feature of SANergy designed to improve the performance of serving small files.

---

### C.2 ZOOM questions and answers

This appendix describes the techniques used by ZOOM, as well as some possible restrictions, through a series of questions and answers.

#### C.2.1 What does ZOOM do?

ZOOM accelerates access to small files that are opened for read-only usage. It does not improve the performance of files opened for write, create, or update. It also improves the performance of popular file inspection APIs such as “stat()” and “opendir()” which are heavily used for most file, Web, and backup activities.

#### C.2.2 Can files still be written on a ZOOM machine?

Yes — full read-write capability to the files is still available, and all data is coherent, that is, changes made by one system are seen by all the others. Of course, only files opened for read capability will have the opportunity to be accelerated by ZOOM.

#### C.2.3 Will all files opened for “read” be accelerated by ZOOM?

Typically, yes — but if a file has been changed somewhere in the system, it will be logged as being in a “changed state”. Files in this state will result in all activity bypassing ZOOM, which typically means occurring at NFS speeds.

#### **C.2.4 How does ZOOM work?**

The ZOOM client machines will double-mount the specified file system. They will mount it once as an ordinary NFS read-write file system and that is the mount-point or filesystem that the applications should use. The clients will also mount the file system directly as a regular hard disk type mount, but specify the read-only flag on the mount command. It is typical for the mount to specify the “hide” flag as well, so that applications are not aware of the disk.

The ZOOM logic watches all file opens, and if it believes that a file open meets the ZOOM criteria (if the file has been opened for read-only access and has not been noted as being in a changed state), it will redirect the open to the fast hard mount access point. Otherwise, the file open and thus all the subsequent activity will continue down the original path, that is, the NFS path.

#### **C.2.5 Which SANergy MDCs support ZOOM?**

When using ZOOM, a server provides ZOOM services to clients for one or more file systems. It is possible, but not necessary, for it to run on the SANergy MDC. Because of this, the question of which MDCs support ZOOM is no longer applicable. However, the platform of the ZOOM server must match that of the clients; therefore, the ZOOM server platform must match the SANergy host platform. More information related to this can be found in section C.2.17, “What configurations are appropriate?” on page 288.

The ZOOM server does not serve out file block allocation information. Thus, unlike the SANergy MDC, it is not tied directly into the filesystem or the operating system.

#### **C.2.6 What does the ZOOM server do?**

It manages the list of files that are in a “changed” state. Because it is running on the only system that can ultimately make changes to the file system, it sees all the changes and broadcasts those changes to the ZOOM clients, as appropriate. The ZOOM server is not a driver. It is a simple daemon.

#### **C.2.7 Will ZOOM work with HBA failover or dual paths?**

This is not a ZOOM issue. Unlike regular SANergy, ZOOM never issues direct I/O to raw devices and always issues its operations at the file system level. Each ZOOM member has to be able to mount the shared-storage as a file system directly (but read-only). As long as the file system mounts, ZOOM does not care what technology is used under the file system. HBA failover and dual path I/O happen at an architectural layer below the file system.

### **C.2.8 Will ZOOM work with the Veritas Volume manager?**

As long as each computer can mount the file system correctly but as read-only, ZOOM will work. It might be necessary to procure additional licenses for the volume manager for the ZOOM machines. This depends on the requirements of the volume manager.

### **C.2.9 Will ZOOM work with other file systems?**

ZOOM has no sensitivity to the file system type. You can use any file system that you desire, as long as you can mount it on all systems directly and simultaneously. Again, it must be mounted read-write on the ZOOM server system and read-only on all the ZOOM clients. This may require you to obtain additional licenses for the file system software. Check with your file system provider.

### **C.2.10 How fast is ZOOM?**

Within a few instructions on each file system `open()`, ZOOM determines the final target for the open and then steps out of the way. The subsequent operations will occur at their natural NFS or hard mounted speeds. The actual performance numbers achieved can vary significantly; however, one performance test using a random set of files showed 1400 file opens per second using NFS, compared with 35,000 when using ZOOM.

Computers with a large amount of RAM usable for file system cache will have the best performance. Although NFS can do some caching, it is minimal compared to the caching that a native file system can do. A hard mounted file system can cache directory and inode entries, which can dramatically increase performance of files that have not even been previously referenced.

### **C.2.11 What are the exact steps that occur when a file is changed?**

When a file is changed, first the ZOOM server needs to notice this change. This is accomplished with the `stat` daemon. Once it discovers a file that is updated, it then checks to see which ZOOM client computers are interested in changes to that tree. Each ZOOM client can be interested in different areas of the file system. For each one that is interested, the ZOOM server sends a message to the client and awaits an acknowledgement. Both the ZOOM server and clients keep the list of changed files in memory. The expectation is that this list will be fairly small on a per-watched-tree basis. If the list gets very large, over 200 entries, housekeeping of this list could start to consume significant CPU resources.

### **C.2.12 What happens if the same file changes frequently?**

It all depends on how ZOOM is configured. Typically, once ZOOM notifies a particular host that a file is in a changed state, it will not tell that client again, because once it is in a changed state, the client will not use the read-only mount, and will instead revert to the read-write system via NFS for the data anyway. One example of this is a log file.

### **C.2.13 What is saturation?**

ZOOM is really intended to work on file trees that seldom change. Yet, changes are a reality of life. A few changes here and there and ZOOM will operate properly. But what if something occurs which initiates many rapid changes to a tree? One example is copying in an entire new data set. On a per-watched-tree basis, ZOOM maintains a value for the maximum number of files to be in a changed state; this is configured during the zoom-watch command. Once that level is hit, ZOOM will no longer monitor that directory for changes and basically shuts that tree down from ZOOM acceleration on all clients.

### **C.2.14 Can ZOOM ignore some directories that change too often?**

Currently, ZOOM watches directory trees, which includes all sub-directories under the watched directory. It is possible that some of those directories do not need ZOOM and might have characteristics that make it unsuitable for ZOOM (too many new files being created too often, for instance). There is not a specific command to disable a sub-tree. However, the administrator can achieve that behavior by creating a sub-watch tree and setting the saturation level to 1, which will effectively disable that tree quickly.

### **C.2.15 What is an inappropriate type of system usage for ZOOM?**

It would be inappropriate to configure ZOOM on trees where there are a great many changes taking place continuously. If the changes are located mostly in portions of the tree, ZOOM can be disabled on that portion. Or, if the changes are happening to the same small set of files over and over again, but frequently, ZOOM will ignore those files entirely, allowing activity to take place on NFS. The rest of the tree and the other trees would still benefit.

If the customer is expecting large files to be “written” fast, he should consider using regular SANergy. SANergy is best used when customers have files that are megabytes in size and read-write capability is desired. Regular SANergy can be told “not to accelerate” smaller files, for example, files under 50 KB. This should be done, because SANergy’s attempt to accelerate such files can actually slow them down. However, the transactions will then only occur at LAN speeds at best, even for reading files.

ZOOM makes the reading of small files occur at the natural speed of a hard mounted file system. Proper configuration can make a big difference in overall system performance and satisfaction. ZOOM offers a number of run-time “views” to show performance and sizes of trees and the amount of changes occurring in the system. A successful ZOOM installation will have a system administrator who is intimate with how ZOOM work, and who is willing to put some time into properly configuring the system.

### **C.2.16 What is the best way to start with ZOOM and to test it?**

It is best to first configure your networks in a typical client-server arrangement. Select the computers to be the file system “owners”, which are the ones that will hard mount the storage read-write, and then configure all of the clients as you would without SANergy or ZOOM. That is, configure permissions, exports, mounts, and so on. Then test the system heavily and be sure it works as desired. Understand that it will be slower than when ZOOM is active.

Next, start up the ZOOM server daemon. Then start the ZOOM client daemon on a single computer and configure the “mappings” and establish a single watch-directory. Issue the normal SANergy commands to activate the SANergy libraries, and try an experiment of copying, such as reading a file from the watched directory. (Note: Send the output to /tmp or some other non-watched directory.) Check the ZOOM stats in the GUI or the command line tool and be sure the expected figures have changed. Run some other tests by hand and possibly execute some performance tests.

Finally, start up the real application, making sure that you are launching it from a session that has the SANergy environment variables set. Carefully monitor the application and the ZOOM stats and the standard SANergy error logs. Everything should run just as it did prior to starting ZOOM, except faster.

### **C.2.17 What configurations are appropriate?**

All of the members in the ZOOM configuration must use exact the same system type. Actually, they must be all able to natively mount the desired file systems. There are some file systems that are supported on various platforms. For instance, UFS can be mounted on a Linux system, and thus Solaris and Linux could potentially both work together in a ZOOM environment. Any system can also be a regular SANergy system, client, or MDC, and can talk to any other SANergy system, even if it is a different platform type. Certainly, regular LAN connectivity to any other system type is allowed as well.

Here is an example configuration: We have a Solaris MDC and as SANergy hosts, and we have both an AIX and a Solaris system. To implement ZOOM in this environment, first we could set up a ZOOM server on the Solaris MDC and a ZOOM client on the Solaris SANergy host. The Solaris ZOOM server could be on a third Solaris system, that is, not on the MDC. For AIX, we would set up the AIX SANergy host as a ZOOM client. Its ZOOM server would have to be a second AIX system.

---

## Appendix D. Using the additional material

This redbook also contains additional material downloadable from the Web. We described the use of this material in B.1.5, "Server access to the mapped drive" on page 273. See the appropriate section below for instructions on using or downloading each type of material.

---

### D.1 Locating the additional material on the Internet

The support material associated with this redbook is available in softcopy on the Internet from the IBM Redbooks Web server. Point your Web browser to:

<ftp://www.redbooks.ibm.com/redbooks/SG246146>

Alternatively, you can go to the IBM Redbooks Web site at:

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

Select the **Additional materials** and open the directory that corresponds with the redbook form number.

---

### D.2 Using the Web material

The additional Web material that accompanies this redbook includes the following:

| <i>File name</i>    | <i>Description</i>  |
|---------------------|---------------------|
| <b>SG246146.zip</b> | Zipped Code Samples |

#### D.2.1 System requirements for downloading the Web material

The following system configuration is recommended for downloading the additional Web material.

|                          |                    |
|--------------------------|--------------------|
| <b>Hard disk space:</b>  | .5 MB minimum      |
| <b>Operating System:</b> | Windows 95/NT/2000 |
| <b>Processor:</b>        | Pentium or higher  |
| <b>Memory:</b>           | 64 MB              |

#### D.2.2 How to use the Web material

Create a subdirectory (folder) on your workstation and copy the contents of the Web material into this folder. Unzip using your favorite zip utility.



## **Appendix E. Special notices**

This publication is intended to help professionals plan and implement a SANergy installation. The information in this publication is not intended as the specification of any programming interfaces that are provided by any of the solutions or products mentioned. See the PUBLICATIONS section of the IBM Programming Announcement for each described IBM product for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

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

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact IBM Corporation, Dept. 600A, Mail Drop 1329, Somers, NY 10589 USA.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers

attempting to adapt these techniques to their own environments do so at their own risk.

Any pointers in this publication to external Web sites are provided for convenience only and do not in any manner serve as an endorsement of these Web sites.

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

e (logo)®   
IBM ®  
AT  
Current  
Magstar  
SP  
NetView

Redbooks  
Redbooks Logo   
CT  
Enterprise Storage Server  
OS/2  
XT

The following terms are trademarks of other companies:

Tivoli, Manage. Anything. Anywhere., The Power To Manage., Anything. Anywhere., TME, NetView, Cross-Site, Tivoli Ready, Tivoli Certified, Planet Tivoli, and Tivoli Enterprise are trademarks or registered trademarks of Tivoli Systems Inc., an IBM company, in the United States, other countries, or both. In Denmark, Tivoli is a trademark licensed from Kjøbenhavns Sommer - Tivoli A/S.

C-bus is a trademark of Corollary, Inc. in the United States and/or other countries.

Solaris, Java and all Java-based trademarks and logos are trademarks or registered trademarks of Sun Microsystems, Inc. in the United States and/or other countries.

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

ActionMedia, LANDesk, MMX, Pentium and ProShare are trademarks of Intel Corporation in the United States and/or other countries.

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

SET, SET Secure Electronic Transaction, and the SET Logo are trademarks owned by SET Secure Electronic Transaction LLC.

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



---

## Appendix F. Related publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this redbook.

---

### F.1 IBM Redbooks

For information on ordering these publications see “How to get IBM Redbooks” on page 299.

- *Using Tivoli Storage Manager in a SAN Environment*, SG24-6132
- *Planning and Implementing an IBM SAN*, SG24-6116
- *Designing an IBM Storage Area Network*, SG24-5678
- *Introduction to Storage Area Network, SAN*, SG24-5470
- *IBM e(logo)server xSeries Clustering Planning Guide*, SG24-5678
- *Getting Started with Tivoli Storage Manager: Implementation Guide*, SG24-5416
- *Tivoli Storage Manager Version 3.7: Technical Guide*, SG24-5477
- *Storage Networking Virtualization: What's it all about?*, SG24-6210
- *IP Storage Networking: IBM NAS and iSCSI Solutions*, SG24-6240
- *IBM Modular Storage Server An Introduction Guide*, SG24-6103
- *Samba Installation, Configuration, and Sizing Guide*, SG24-6004
- *Tivoli Storage Management Concepts*, SG24-4877
- *Using Tivoli Storage Manager in a SAN Environment*, SG24-6132
- *Getting Started with Tivoli Storage Manager: Implementation Guide*, SG24-5416

---

## F.2 IBM Redbooks collections

Redbooks are also available on the following CD-ROMs. Click the CD-ROMs button at [ibm.com/redbooks](http://ibm.com/redbooks) for information about all the CD-ROMs offered, updates and formats.

| CD-ROM Title   | Collection Kit Number |
|--|-----------------------|
| IBM System/390 Redbooks Collection                                 | SK2T-2177             |
| IBM Networking Redbooks Collection                                 | SK2T-6022             |
| IBM Transaction Processing and Data Management Redbooks Collection | SK2T-8038             |
| IBM Lotus Redbooks Collection                                      | SK2T-8039             |
| Tivoli Redbooks Collection   | SK2T-8044             |
| IBM AS/400 Redbooks Collection                                     | SK2T-2849             |
| IBM Netfinity Hardware and Software Redbooks Collection            | SK2T-8046             |
| IBM RS/6000 Redbooks Collection                                    | SK2T-8043             |
| IBM Application Development Redbooks Collection                    | SK2T-8037             |
| IBM Enterprise Storage and Systems Management Solutions            | SK3T-3694             |

---

## F.3 Other resources

The following publications are also relevant as a further information source:

- *Tivoli SANergy Administrator's Guide*, GC26-7389
- *Tivoli Storage Manager for Windows Administrator's Guide*, GC35-0410
- *Tivoli Storage Manager for Windows Administrator's Reference*, GC35-0411
- *Tivoli Storage Manager for Windows Using the Backup-Archive Client*, SG26-4117
- *OTG DiskXtender2000 Data Manager System Guide*  
(available at <http://www.otg.com/tivoli/Data Manager System Guide.pdf>)

---

## F.4 Referenced Web sites

These Web sites are also relevant as further information sources:

- <http://www.samba.org/>  
Website for the Samba filesharing product
- <http://www.microsoft.com/>  
Information on Microsoft Operating Systems and applications
- <http://www.microsoft.com/support>  
Microsoft support Web site
- <http://www.microsoft.com/windows2000/sfu/>  
Microsoft Network Information Service
- <http://www.tivoli.com/sanergy>  
Main Tivoli SANergy Web site
- <http://www.tivoli.com/tsm>  
Main Tivoli Storage Manager Web site
- <http://www.tivoli.com/tsnm>  
Main Tivoli Storage Network Manager Web site
- [http://www.tivoli.com/support/sanergy/sanergy\\_req.html](http://www.tivoli.com/support/sanergy/sanergy_req.html)  
Tivoli SANergy Web site
- <http://www.tivoli.com/support/sanergy/maintenance.html>  
Tivoli SANergy available patches
- <http://www.otg.com/tivoli>  
Main OTG DiskXtender2000 Web site
- <http://www.rpmfind.net>  
Useful Linux utilities



## How to get IBM Redbooks

This section explains how both customers and IBM employees can find out about IBM Redbooks, redpieces, and CD-ROMs. A form for ordering books and CD-ROMs by fax or e-mail is also provided.

- **Redbooks Web Site** [ibm.com/redbooks](http://ibm.com/redbooks)

Search for, view, download, or order hardcopy/CD-ROM Redbooks from the Redbooks Web site. Also read redpieces and download additional materials (code samples or diskette/CD-ROM images) from this Redbooks site.

Redpieces are Redbooks in progress; not all Redbooks become redpieces and sometimes just a few chapters will be published this way. The intent is to get the information out much quicker than the formal publishing process allows.

- **E-mail Orders**

Send orders by e-mail including information from the IBM Redbooks fax order form to:

|                            | <b>e-mail address</b>   |
|----------------------------|---|
| In United States or Canada | pubscan@us.ibm.com  |
| Outside North America      | Contact information is in the "How to Order" section at this site:<br><a href="http://www.elink.ibm.com/pbl/pbl">http://www.elink.ibm.com/pbl/pbl</a> |

- **Telephone Orders**

|                           |  |
|---------------------------|--|
| United States (toll free) | 1-800-879-2755   |
| Canada (toll free)        | 1-800-IBM-4YOU   |
| Outside North America     | Country coordinator phone number is in the "How to Order" section at this site:<br><a href="http://www.elink.ibm.com/pbl/pbl">http://www.elink.ibm.com/pbl/pbl</a> |

- **Fax Orders**

|                           |  |
|---------------------------|--|
| United States (toll free) | 1-800-445-9269   |
| Canada                    | 1-403-267-4455   |
| Outside North America     | Fax phone number is in the "How to Order" section at this site:<br><a href="http://www.elink.ibm.com/pbl/pbl">http://www.elink.ibm.com/pbl/pbl</a> |

This information was current at the time of publication, but is continually subject to change. The latest information may be found at the Redbooks Web site.

### IBM Intranet for Employees

IBM employees may register for information on workshops, residencies, and Redbooks by accessing the IBM Intranet Web site at <http://w3.itso.ibm.com/> and clicking the ITSO Mailing List button. Look in the Materials repository for workshops, presentations, papers, and Web pages developed and written by the ITSO technical professionals; click the Additional Materials button. Employees may access MyNews at <http://w3.ibm.com/> for redbook, residency, and workshop announcements.



---

## Abbreviations and acronyms

|              |  |              |                                       |
|--------------|--|--------------|---------------------------------------|
| <b>ANSI</b>  | American National Standards Institute        | <b>MSS</b>   | Modular Storage Server                |
| <b>CIFS</b>  | Common Internet File Support                 | <b>NAS</b>   | Network Attached Storage              |
| <b>CLI</b>   | Command Line Interface                       | <b>NFS</b>   | Network File System                   |
| <b>DLL</b>   | Dynamic-Link Library                         | <b>NIS</b>   | Network Information Service           |
| <b>FC</b>    | Fibre Channel                                | <b>NTFS</b>  | NT File System                        |
| <b>GPFS</b>  | General Parallel File System                 | <b>RAID</b>  | Redundant Array of Independent Disks  |
| <b>GUI</b>   | Graphical User Interface                     | <b>RDBMS</b> | Relational Database Management System |
| <b>HA</b>    | High Availability                            | <b>SAN</b>   | Storage Area Network                  |
| <b>HBA</b>   | Host Bus Adapter                             | <b>SCSI</b>  | Small Computer Systems Interface      |
| <b>HSM</b>   | Hierarchical Storage Management              | <b>SFU</b>   | Services for UNIX                     |
| <b>IBM</b>   | International Business Machines Corporation  | <b>SMB</b>   | Server Message Block                  |
| <b>IDS</b>   | Intelligent Disk Subsystem                   | <b>SMS</b>   | Systems Managed Storage               |
| <b>IIS</b>   | Internet Information Services                | <b>SNMP</b>  | Simple Network Management Protocol    |
| <b>iSCSI</b> | SCSI over IP                                 | <b>SQL</b>   | Structured Query Language             |
| <b>ITSO</b>  | International Technical Support Organization | <b>TDP</b>   | Tivoli Data Protection                |
| <b>JBOD</b>  | Just a Bunch of Disks                        | <b>UFS</b>   | UNIX File System                      |
| <b>LAN</b>   | Local Area Network                           | <b>UNC</b>   | Universal Naming Convention           |
| <b>LRU</b>   | Least Recently Used                          | <b>URL</b>   | Universal Resource Locator            |
| <b>LUN</b>   | Logical Unit Number                          | <b>VMM</b>   | Virtual Memory Manager                |
| <b>LVM</b>   | Logical Volume Manager                       |              |                                       |
| <b>MDC</b>   | Meta-Data Controller                         |              |                                       |
| <b>MFT</b>   | Master File Table                            |              |                                       |
| <b>MSCS</b>  | Microsoft Cluster Server                     |              |                                       |



## Index

### A

- acdirmax 90
- acdirmin 90
- acregmax 90
- acregmin 90
- actime 90
- administrative share 24
- AIX
  - cfgmgr command 76
  - creating a mount point 67
  - iostat command 93
  - logical volume manager 75
  - lsdev command 76
  - lspp command 67
  - mknfsmnt command 68
  - showmount command 67
  - Virtual Memory Manager 94
  - vmstat command 93
- ANSI 6
- Apache Web server 179, 270

### B

- backup
  - SAN-based 5
- backup and recovery 13
- bypass 14

### C

- cache line 98
- caching modes 101
- CIFS 8, 9, 21, 176
  - chaining shares 176
  - file sharing setup 21
  - Samba server 57
  - setup on Windows 69
- CIFS share
  - with MSCS 126
- connectivity
  - multiple hosts 7
- CPU impact of 11

### D

- data sharing 7
- database

- on shared storage 181
  - SQL Server 182
  - with SANergy 182, 223
- datagram 104
- disk drive
  - identifying 41
- disk volumes
  - dynamically striped 175
  - fusing 50, 53
  - hardware striped 174
  - setting MDC 42
  - setting SANergy ownership 41
- DLL 273
- dynamic disk 132, 170, 174
  - creating 152

### F

- failover 105
- Fibre Channel 4
- file
  - open 284
- file access failover 104
- file servers
  - benefits of volume sharing 12
- file sharing
  - CIFS 21
  - NFS 25
  - permissions 23
  - SAN-based 13
  - setup 19, 57
  - UNIX 58
  - Windows 21
  - with Samba 57
- file system
  - exported permissions 28, 31
  - exporting 28
  - mount 284
  - mounting 67
  - NFS export 59
- file system mapping
  - persistence 272
- fused file size limit 102
- fused files
  - excluding 103
- fused I/O 19
  - reverting to the LAN 104

## G

General Parallel File System 7  
group ids 27  
group mapping 27

## H

HBA 16, 20, 57  
heterogeneous disk sharing 10  
hierarchical storage management 9, 227  
high availability 105, 171  
homogeneous sharing 7  
hyperextension 101

## I

intelligent storage subsystems 4  
Internet Information Services 179  
interoperability  
    SAN components 6  
iostat 93  
iSCSI 5

## L

LAN-based sharing 8  
LAN-free backup 191  
Linux  
    defining HBAs 263  
    device drivers 262  
    lilo 265  
    linuxconf utility 35  
    nproc parameter for nfsd 90  
    with a SAN 261  
    with SANergy 267  
logging 103  
LUNs  
    dynamic striping 175  
    hardware striping 174

## M

managed buses 39  
master file table 94  
MDC  
    installation 70  
    SANergy installation 38  
    Windows 19  
Meta Data Controller. see MDC  
Microsoft  
    IIS 179

Services for UNIX 30

SFU 170

SQL Server 182

Microsoft Cluster Server 177

Modular Storage Server 173

mount 90, 266, 284

    acrdirmax parameter 90

    acrdirmmin parameter 90

    acregmax parameter 90

    acregmin parameter 90

    actime parameter 90

MSCS

    Cluster Administrator 120

    defining CIFS share 126

    module for SANergy 107

    quorum resource 106

    resource groups 107

    resource types 107

    with SANergy 107

## N

NAS 4, 5, 181

    benefits of 11

    vs SANergy 12

net use command 274

Netscape 74

netstat 93

Network Attached Storage. see NAS

Network Information Service. see NIS

NFS 9, 284

    exporting file systems 59

    file sharing 25

    mount command 90, 266

    mounting filesystems 35

    nfsstat command 90, 92

    nservers parameter 90

    re-exporting file systems 176

    setup on AIX 67

    verifying exports 67

NFS server

    export permissions 28

    exported permissions 31

    exporting file systems 28

    group mapping 27

    Hummingbird Maestro 25

    MS Services for UNIX 30

    name mapping 27

    permissions 26

- reloading export list 29
- usage statistics 26
- user name mapping 32
- nfsd 90
  - nproc parameter 90
- nfsstat 90, 92
- NIS 32
- NTFS 19

## O

- OTG DiskXtender 2000 227
  - installing 231
  - lan-free implementation 253

## P

- permissions 23

## Q

- quorum disk 106

## R

- RAID0 170
- RAID5 170

## S

- Samba 13, 57, 177
  - configuration file 64
  - environment for SANergy 72
  - graphical configuration 61
  - server setup 60
  - starting the daemon 61
  - swat administration tool 61
- SAN 4
  - backups 5
  - benefits of 5
  - components 6
  - data sharing 7
  - file sharing 13
  - interoperability 6
  - software exploitation 6
  - standards 6
- SAN I/O 11
- SANergy 9, 104
  - and small files 14, 102
  - cache settings 97
  - caching modes 101
  - config.txt file 80

- environment variables 79
- fuse command 77, 91
- fused command 78
- fused file size limit 78
- fused I/O 19
- fusion exclusion list 103
- HA component 105
- high availability 105, 171
- hyperextension 101
- installation 71
- introduction 3
- key command 76
- LAN-free backup 191
- licensing 16
- limitations 14
- logging 103
- managed buses 39, 82
- MDC installation 38
- minfused command 78
- minimum fused file size 102
- MSCS component 116
- MSCS module 107
- owner command 77, 91
- patches 38, 43, 71
- performance tester 50, 54, 84
- performance tuning 96
- permanent configuration settings 80
- requirements 15
- SANergy host installation 44, 52
- SANergyconfig command 96
- SANergyshsetup 76
- SANergyshsetup command 79
- setting MDC for disks 42
- setting volume ownership 41
- setup tool 84
- stats command 78
- supported platforms 16
- unfuse command 79
- unfused I/O 19
- UNIX implementation 57
- UNIX startup scripts 55
- version command 76
- volume ownership 73
- vs NAS 12
- with databases 182, 223
- with HSM 228
- with Linux 267
- with MSCS 107
- with strip sets 134

- with Tivoli Storage Manager 191
- SANergy host 19
  - installation 74
- SANergy MDC
  - setup 70
- SCSI protocols 4
- SCSI reserve/release 107
- Services for UNIX 170
- small files and SANergy 14, 102
- SMB 9, 60
- smbd 61
- SNMP 39
- Solaris
  - nsservers parameter 90
  - owner command 73
- space management 227
- standards
  - SAN 6
- stat daemon 285
- Storage Agent
  - installing 254
  - SETSTORAGESEVER command 256
  - setup as a service 258
  - tsmdlst command 257
- Storage Area Network. see SAN
- storage consolidation 12
- storage virtualization 9, 169, 180
- stripe set 132
  - creating 136
- Systems Managed Storage 8

**T**

- T11 standard 6
- TCP/IP 11
- Tivoli Storage Manager 191
  - admscsi device driver 255
  - DEFINE DRIVEMAPPING command 258
  - DEFINE SERVER command 255
  - drive access at startup 271
  - dsm.opt file for HSM 231
  - ENABLE EVENT command 275
  - QUERY DRIVE command 257
  - QUERY SESSION command 251, 253, 259
  - REGISTER NODE command 229
  - Storage Agent 254
  - userexit 275
- Tivoli Storage Network Manager 6

**U**

- UNC 271
- unfused I/O 19
- UNIX
  - chmod command 67
  - file sharing 58
  - fusing disk volumes 53
  - installing SANergy host 52
  - mkdir command 67
  - mount command 36, 67
  - mounting NFS filesystems 35
  - processes with SANergy 55
  - SANergy implementation 57
  - SANergy startup scripts 55
  - vmstat command 94
- user id logon properties 274
- user ids 27

**V**

- virtual memory manager 94
- vmstat 93, 94
- Volume sharing 12

**W**

- Web serving 12, 179
  - Apache 179, 270
- Windows 19
  - administrative share 24
  - administrator account 274
  - disk signature 80
  - dynamic disk 132
  - filesharing 21
  - fusing disk volumes 50
  - installing SANergy host 45
  - mapping network drive 33
  - Master File Table 94
  - preventing drive letter assignment 47
  - stripe set 132

**Z**

- zoning 20, 44, 57, 74
- ZOOM 283
  - configurations 288

## IBM Redbooks review

Your feedback is valued by the Redbook authors. In particular we are interested in situations where a Redbook "made the difference" in a task or problem you encountered. Using one of the following methods, **please review the Redbook, addressing value, subject matter, structure, depth and quality as appropriate.**

- Use the online **Contact us** review redbook form found at [ibm.com/redbooks](http://ibm.com/redbooks)
- Fax this form to: USA International Access Code + 1 845 432 8264
- Send your comments in an Internet note to [redbook@us.ibm.com](mailto:redbook@us.ibm.com)

|   |  |
|---|--|
| <b>Document Number</b>  | SG24-6146-00   |
| <b>Redbook Title</b>  | A Practical Guide to Tivoli SANergy  |
| <b>Review</b>   | <br><br><br><br><br><br><br><br>   |
| <b>What other subjects would you like to see IBM Redbooks address?</b>  | <br><br><br>   |
| <b>Please rate your overall satisfaction:</b>   | <input type="radio"/> Very Good <input type="radio"/> Good <input type="radio"/> Average <input type="radio"/> Poor  |
| <b>Please identify yourself as belonging to one of the following groups:</b>  | <input type="radio"/> Customer <input type="radio"/> Business Partner <input type="radio"/> Solution Developer<br><input type="radio"/> IBM, Lotus or Tivoli Employee<br><input type="radio"/> None of the above |
| <b>Your email address:</b><br>The data you provide here may be used to provide you with information from IBM or our business partners about our products, services or activities. | <input type="checkbox"/> Please do not use the information collected here for future marketing or promotional contacts or other communications beyond the scope of this transaction.                             |
| <b>Questions about IBM's privacy policy?</b>  | The following link explains how we protect your personal information.<br><a href="http://ibm.com/privacy/yourprivacy/">ibm.com/privacy/yourprivacy/</a>  |





Redbooks

# A Practical Guide to Tivoli SANergy

(0.5" spine)

0.475" <-> 0.875"

250 <-> 459 pages







# A Practical Guide to Tivoli SANergy



**Redbooks**

**Application transparent, supercharged SAN-based filesharing**

**Practical installation and configuration scenarios**

**High availability and performance tuning tips**

Providing shared access to files is an important aspect of today's computing environment, as it allows easier control and consolidation of data and other assets and enhances information flow through an organization. Traditionally filesharing has been done over the traditional TCP/IP LAN. Storage Area Networks (or SANs) provide the ability for storage to be accessed and moved across a separate dedicated high-speed network. Tivoli SANergy transparently brings these two concepts together, providing all the benefits of LAN-based filesharing at the speed of the SAN.

This IBM Redbook provides an introduction to Tivoli SANergy in various environments, including various flavors of UNIX, Microsoft Windows NT, and Windows 2000. It covers installation and setup of the product, with advice and guidance on tuning and performance. It also describes integrating SANergy with other products, including Tivoli Storage Manager and Microsoft Cluster Services.

This book is written for IBM, Tivoli, customer, vendor, and consulting personnel who wish to gain an understanding of the Tivoli SANergy product and how best to use it in their environments. We assume a basic understanding of filesharing concepts in various operating system environments, as well as SAN concepts and implementation.

**INTERNATIONAL TECHNICAL SUPPORT ORGANIZATION**

**BUILDING TECHNICAL INFORMATION BASED ON PRACTICAL EXPERIENCE**

IBM Redbooks are developed by the IBM International Technical Support Organization. Experts from IBM, Customers and Partners from around the world create timely technical information based on realistic scenarios. Specific recommendations are provided to help you implement IT solutions more effectively in your environment.

**For more information:**  
[ibm.com/redbooks](http://ibm.com/redbooks)

SG24-6146-00

ISBN 0738421979