HITACHI
DATA SYSTEMS

HiCommand® Protection Manager User's Guide

MK-94HC070-07

© 2006, 2007 Hitachi, Ltd., Hitachi Data Systems Corporation, ALL RIGHTS RESERVED

Notice: No part of this publication may be reproduced or transmitted in any form or by any electronic or mechanical means, including photocopying and recording, or stored in a database or retrieval system for any purpose, without the express written permission of Hitachi Data Systems Corporation (hereinafter referred to as "Hitachi Data Systems").

Hitachi Data Systems reserves the right to make changes to this document at any time without notice and assumes no responsibility for its use. Hitachi Data Systems products and services can only be ordered under the terms and conditions of Hitachi Data Systems' applicable agreements, including license agreements. All of the features described in this document may not be currently available. Refer to the most recent product announcement or contact your local Hitachi Data Systems sales office for information on feature and product availability.

This document contains the most current information available at the time of publication. When new and/or revised information becomes available, this entire document will be updated and distributed to all registered users.

Trademarks

Hitachi Data Systems is a registered trademark and service mark of Hitachi, Ltd., and the Hitachi Data Systems design mark is a trademark and service mark of Hitachi, Ltd. HiCommand is a registered trademark of Hitachi, Ltd.

Hitachi TagmaStore, Lightning 9900, Thunder 9200, Thunder 9500, and TagmaStore are trademarks of Hitachi Data Systems Corporation in the United States and other countries.

Itanium is a registered trademark of Intel Corporation or its subsidiaries in the United States and other countries.

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

Active Directory is either a registered trademark or a trademark of Microsoft Corporation in the United States and/or other countries.

Microsoft, MS-DOS, Visual Basic, and Windows are registered trademarks or trademarks of Microsoft Corp. in the U.S. and other countries.

Mozilla is a trademark of the Mozilla Foundation in the U.S and other countries.

Netscape is a registered trademark of Netscape Communications Corporation in the United States and other countries.

ODBC is Microsoft's strategic interface for accessing databases.

Sun is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries.

Sun Microsystems is a trademark or registered trademark of Sun Microsystems, Inc. in the United States and other countries.

Symantec is a U.S. registered trademark of Symantec Corporation.

VERITAS is a trademark or registered trademark of Symantec Corporation in the U.S. and other countries.

All other brand or product names are or may be trademarks or service marks of and are used to identify products or services of their respective owners.

Notice of Export Controls

Export of technical data contained in this document may require an export license from the United States government and/or the government of Japan. Please contact the Hitachi Data Systems Legal Department for any export compliance questions.

Document Revision Level

Revision	Date	Description
MK-94HC070-00	August 2004	Initial Release
MK-94HC070-01	October 2004	Revision 1, supersedes and replaces MK-94HC070-00
MK-94HC070-02	March 2005	Revision 2, supersedes and replaces MK-94HC070-01
MK-94HC070-03	May 2005	Revision 3, supersedes and replaces MK-94HC070-02
MK-94HC070-04	November 2005	Revision 4, supersedes and replaces MK-94HC070-03
MK-94HC070-05	January 2006	Revision 5, supersedes and replaces MK-94HC070-04
MK-94HC070-06	December 2006	Revision 6, supersedes and replaces MK-94HC070-05
MK-94HC070-07	May 2007	Revision 7, supersedes and replaces MK-94HC070-06

Preface

This document describes how to install, manage, and operate HiCommand® Protection Manager and how to use the Protection Manager command line interface (CLI). This document is intended for system administrators who use Protection Manager to manage data on storage subsystems. The readers of this manual should have a basic knowledge of the management functions for each of the following:

- Windows® operating system
- Database product (e.g., Exchange Server)
- Hitachi subsystem(s) (e.g., Hitachi TagmaStore™ Universal Storage Platform, Hitachi Thunder™ 9500V)
- RAID Manager
- Backup management product (e.g., VERITAS NetBackup, VERITAS Backup Exec)

This document is part of the following HiCommand Protection Manager user documentation set:

- *HiCommand® Protection Manager Console User's Guide*, MK-94HC071
- *HiCommand® Protection Manager Command Reference*, MK-94HC072
- *HiCommand® Protection Manager Messages*, MK-94HC073

Note: For troubleshooting information on HiCommand® Protection Manager, please refer to the applicable user document (e.g., *HiCommand® Protection Manager User's Guide* and/or *HiCommand® Protection Manager Command Reference*).

Note: The use of HiCommand Protection Manager and all other Hitachi Data Systems products is governed by the terms of your agreement(s) with Hitachi Data Systems.

Software Version

This document revision applies to HiCommand Protection Manager version 5.7.

Convention for Storage Capacity Values

Storage capacity values displayed by HiCommand Protection Manager are calculated based on the following values:

1 KB (kilobyte) = 1,024 bytes
1 MB (megabyte) = $1{,}024^2$ bytes
1 GB (gigabyte) = $1{,}024^3$ bytes
1 TB (terabyte) = $1{,}024^4$ bytes

Referenced Documents

- *HiCommand® Protection Manager Console User's Guide*, MK-94HC071
- *HiCommand® Protection Manager Command Reference*, MK-94HC072
- *HiCommand® Protection Manager Messages*, MK-94HC073

Comments

Please send us your comments on this document. Make sure to include the document title, number, and revision. Please refer to specific section(s) and paragraph(s) whenever possible.

- **E-mail:** doc.comments@hds.com
- **Fax:** 858-695-1186
- **Mail:**
  Technical Writing, M/S 35-10
  Hitachi Data Systems
  10277 Scripps Ranch Blvd.
  San Diego, CA 92131

Thank you! (All comments become the property of Hitachi Data Systems Corporation.)

Contents

Chapter 1 Overview of Protection Manager

1.1 Introduction to Protection Manager 2
1.1.1 Standard Functions of Protection Manager 2
1.1.2 Optional Functions for Operations on Databases Created by a DBMS 2
1.1.3 Making Protection Manager Functions Available From a GUI 2
1.2 Protection Manager Features 3
1.3 Protection Manager Functions 5
1.3.1 Backup Commands 5
1.3.2 Restore Command 5
1.3.3 Dictionary Map File 5
1.3.4 Pair Management Using Copy Groups 7
1.3.5 Support for Clustering 8
1.3.6 Commands to Reduce Operation Workload 8
1.3.7 Data Management Using a Remote Site (Storage Subsystem) 9
1.3.8 Managing Generations of Secondary Volumes 10
1.4 Protection Manager System Requirements 14
1.4.1 Host Requirements 14
1.4.2 Server Requirements 15
1.4.3 Storage Subsystem Requirements 15
1.4.4 User Authorization at Command Execution 17
1.5 Backup Management and Cluster Software Products 19
1.5.1 Database Products 19
1.6 Conventions for Syntax Explanations 20

Chapter 2 System Configurations for Protection Manager

2.1 Configuring Data Backup and Restoration Between Volumes 24
2.2 Configuring Systems to Store and Restore Data Backed Up to Tape 25
2.3 Executing the Operation Standby Cluster Configuration 26
2.4 Executing the Mutual Standby Cluster Configuration 27
2.5 Configurations for Backup and Restoration Between Subsystems (Remote Sites) 28
2.5.1 TrueCopy or UR Configuration 28
2.5.2 ShadowImage and TrueCopy Multi-Target Configuration 29
2.5.3 ShadowImage and UR Multi-Target Configuration 29
2.5.4 TrueCopy and UR Multi-Target Configuration 30
2.5.5 ShadowImage (Multi-Generation), TrueCopy, or UR Configuration 30
2.6 Precautions for Data Backup and Restoration Between Storage Subsystems 32
2.7 Restoring Data on a Host Not Used for SQL Server Database Backup 33
2.8 Configuring the System to Use the SQL Server Log-Shipping Function 35
2.9 Cascade Configuration for a SQL Server Database Backup 36
2.10 Using a Multi-Target Configuration for SQL Server Database Backup 37
2.11 Requirements When Using RAID Manager to Configure a Pair Volume 38
2.11.1 Considerations When Using TC or UR 39
2.11.2 Requirements for Exchange Database Backup 44
2.12 Volume Configuration for Backing up or Restoring Data 46
2.12.1 Examples of Disk Group Configurations Usable by Protection Manager 48
2.12.2 Precautions for Volume Configuration 51
2.12.3 Precautions for Changing a Volume Configuration 54

2.12.4 Requirements and Precautions for Configuring a Disk Group 55
2.13 Configuring to Back Up Data Using VSS 58
2.13.1 Using VSS With File Systems 59
2.13.2 Using VSS With Exchange Databases 60
2.14 Requirements and Precautions for Databases Subject to Backup 63
2.14.1 Requirements and Precautions for SQL Server Databases 63
2.14.2 Requirements and Precautions for Exchange Databases 72

Chapter 3 Installing Protection Manager

3.1 Installing Protection Manager Programs 82
3.1.1 Installing Protection Manager Copy Controller the First Time 84
3.1.2 Performing an Upgrade Installation of Protection Manager Copy Controller 85
3.1.3 Installing Protection Manager for SQL or Exchange Server the First Time 86
3.1.4 Installing an Upgrade of Protection Manager for SQL or Exchange 87
3.2 Troubleshooting Installation Errors 88
3.2.1 When Installation Fails 88
3.2.2 Examining the Contents of the drm_inst.log File 88
3.3 Uninstalling Protection Manager Programs 89

Chapter 4 Preparing to Use Protection Manager

4.1 Setting up an Environment for Protection Manager 92
4.2 Setting up RAID Manager 94
4.2.1 Backing up One Primary Volume to Multiple Generations 95
4.2.2 Backing up Multiple Primary Volumes to a Single Generation 96
4.2.3 Backing up Multiple Primary Volumes to Multiple Generations 97
4.2.4 Backup or Restore Between Storage Subsystems 98
4.2.5 Setting up a Multi-Target or Cascade Configuration 101
4.3 Setting up Protection Manager 102
4.3.1 Setting Up to Check Status of Cluster Resources 102
4.3.2 Setting Up to Check Status of a Process 104
4.3.3 Setting Up to Check Command Execution Retry Status 105
4.3.4 Setting up the Backup Option (SQL Server Database) 105
4.3.5 Setting the Dictionary Map Synchronization Option (SQL Server Database) 107
4.3.6 Setting the Recovery Option (SQL Server Database) 109
4.3.7 Setting up Restoration While Cluster Resources Are Online 109
4.4 Setting up Protection Manager Linkage with RAID Manager 112
4.4.1 Specifying Instance Numbers 112
4.4.2 Setting up Retry Count and Interval for Checking Pair Status 115
4.4.3 Specifying a Retry Count and Interval for RAID Mgr Command Busy Status 119
4.4.4 Specifying a Retry Count and Interval Appropriate to the Operation Style 119
4.4.5 Setting the Track Size for Copying Data 124
4.4.6 Specifying Volume Pair Creation 126
4.4.7 Settings for Secondary Volume Dynamic Recognition Functionality 126
4.4.8 Setting up Operation Mode When Automatically Selecting Copy Groups 127
4.4.9 Notes on Starting and Stopping RAID Manager 129
4.5 Specifying the Installation Path for RAID Manager 130
4.6 Creating a Dictionary Map File 131
4.6.1 Creating a Dictionary Map File in a Non-Cluster Configuration or on the Backup Server 131

4.6.2 Creating a Dictionary Map File in an Active-Passive Cluster Configuration . 132
4.6.3 Creating a Dictionary Map File in an Active-Active Cluster Configuration .. 134
4.7 Specifying Settings for a Cluster Configuration 137
4.7.1 When the File Server or Database Server Is in a Cluster Configuration..... 137
4.7.2 Creating a Cluster Information Configuration Definition File for VERITAS Cluster Server 139
4.8 Creating a Database Configuration Definition File 142
4.8.1 Creating a Database Configuration Definition File for SQL Server Database Backup 142
4.9 Specifying Settings Related to Linkage with a SQL Server 144
4.10 Specifying Settings for Using VSS..... 147
4.10.1 Using VSS When RM Shadow Copy Provider Version is 01-02-03/03 or Later 149
4.10.2 Using VSS When RM Shadow Copy Provider Version is Earlier Than 01-02-03/03 150
4.11 Setting up Linkage with a Backup Management Product 152
4.11.1 Registering Information for Linkage with a Backup Management Product .. 152
4.11.2 Creating a Configuration Definition File for Tape Backup 153
4.12 Creating a List File 158
4.12.1 Creating a Copy-Group List File..... 158
4.12.2 Creating a Database List File or Mount Point Directory List File..... 159
4.12.3 Creating a Transaction Log List File..... 160
4.13 Updating a Dictionary Map File..... 162
4.14 Specifying How to Mount a Secondary Volume..... 163
4.14.1 Enabling Dynamic Recognition of a Secondary Volume..... 164
4.14.2 Mounting a Secondary Volume for Static Recognition by the OS 169
4.14.3 Determining a Mount Point and Target for Secondary Volume Mounting.... 172
4.15 Preparing to Execute Extended Commands 176
4.15.1 Prerequisites for Using Extended Commands 177
4.15.2 Setting up FTP Service (Tape Backup) 179
4.15.3 Specifying How Extended Commands are Started..... 179
4.15.4 Creating a Host Environment Settings File 180
4.15.5 Creating Operation IDs 181
4.15.6 Creating Operation Definition Files for File System Backup..... 182
4.15.7 Creating Operation Definition Files for SQL Server Database Backup..... 186
4.15.8 Creating Operation Definition Files for Exchange Database Backup 191
4.15.9 Checking the Temporary Directory for Extended Commands 196
4.15.10Checking the Copy-Group List File..... 196
4.15.11Checking the FTP Service Before Backing up to Tape 197
4.16 Creating a User Script 197
4.16.1 Overview of User Scripts 197
4.16.2 Coding Rules for User Scripts..... 198
4.16.3 Examples of User Script Usage 205
4.17 Specifying Settings for Sending Mail..... 208
4.18 Backing Up a Protection Manager Operating Environment..... 210
4.18.1 Files that Need to Be Backed Up..... 210
4.18.2 Recovering the Operating Environment..... 214

Chapter 5 Using Protection Manager

5.1 Protection Manager Commands 218
5.1.1 Extended Commands..... 218

5.1.2 Basic Commands 218
5.2 Correspondence Between Extended Commands and Basic Commands 219
5.3 Precautions for Operations 223
5.3.1 General Precautions 223
5.3.2 Backup and Restore Precautions 223
5.3.3 Precautions for Backup 225
5.3.4 Precautions for Restore Operations 226
5.3.5 Precautions for Executing Commands Simultaneously 227
5.3.6 Precautions for Using Backup Servers 227
5.3.7 Precautions for Updating the Dictionary Map File 229
5.3.8 Tools That Cannot Be Used When Executing Commands 230
5.3.9 Disk Group Configuration That Does Not Allow Backup (Dynamic Disk Configuration) 231
5.3.10 About Pair Volume Resynchronization in a Cascade Configuration 232
5.4 Precautions When Executing Commands 233
5.4.1 Permissions Required for Users Executing Commands 233
5.4.2 Concurrent Execution of Commands 237
5.4.3 Retry Time When an Error Occurs 241
5.4.4 Stopping Retry Operations Caused by Commands That Cannot Execute Concurrently 241
5.4.5 Executing Multiple drmmediabackup or drmmediarestore Commands Concurrently 242
5.5 Starting and Stopping the Protection Manager Service 243
5.5.1 Starting the Protection Manager Service 243
5.5.2 Restarting the Protection Manager Service 244
5.5.3 Stopping the Protection Manager Service 244

Chapter 6 Using Protection Manager With File Systems

6.1 Operations for Backing up and Restoring a File System 246
6.1.1 Precautions When Backing up a File System 246
6.1.2 Precautions for Restoring a File System 247
6.1.3 Precautions for Executing Commands in a Cluster Environment 247
6.1.4 About Forced Termination of Commands 248
6.2 Backing up and Restoring Data Between Volumes 249
6.2.1 Processing Flow for Backing up and Restoring Data Between Volumes 250
6.2.2 Backing up a File System to a Secondary Volume 251
6.2.3 Restoring a File System to a Primary Volume 251
6.3 Backing up and Restoring by Using a Tape Device 253
6.3.1 Processing Flow for Tape Device Backup 254
6.3.2 Backing up a File System to Tape 256
6.3.3 Restoring a File System from Tape 259
6.4 Backing Up a File System by Using a User Script 262
6.4.1 Processing Flow for User Script Backup 262
6.4.2 User Script Example 264
6.4.3 Backup Execution Example 265
6.5 Backup and Restore Between Remote Sites (Remote Copy) 267
6.5.1 Backing up a File System to a Remote Site 269
6.5.2 Restoring a File System to the Local Site 271
6.5.3 Restoring a File System to a Remote Site 273
6.5.4 Recovering a File System from a Remote Site to the Local Site 274

6.6 Using a Backup Management Product That Cannot Link with Protection Manager .. 275
6.7 Backing up and Restoring a File System Using a File Share 276
6.7.1 Preparing to Back up or Restore a File System by Using a File Share 276
6.7.2 Example of Backing up a File System by Using a File Share 276
6.7.3 Example of Restoring a File System Using a File Share..... 277
6.8 Deleting Backup Data 278

Chapter 7 Using HPtM With SQL Server Databases

7.1 Operations for Backing up and Restoring a SQL Server Database..... 280
7.1.1 Precautions When Backing up and Restoring a Database 281
7.1.2 Precautions When Restoring a Database..... 282
7.1.3 Precautions When Recovering a Database..... 283
7.1.4 Precautions When Backing up and Restoring Multiple Databases..... 284
7.1.5 Database Status When Restoring and Recovering 286
7.1.6 Requirements for Transaction Log Backup 288
7.1.7 Precautions for Transaction Log Backups..... 288
7.1.8 Precautions Regarding Sequences of Transaction Logs..... 290
7.1.9 Precautions Regarding Application of Transaction Logs..... 292
7.1.10 SQL Server Database Requirements for Executing Commands 294
7.2 Backing up and Restoring Data Between Volumes..... 300
7.2.1 System Configuration 300
7.2.2 SQL Server Backup Processing Flow 301
7.2.3 Backing up a SQL Server Database to a Secondary Volume..... 302
7.2.4 Restoring a SQL Server Database to a Primary Volume..... 302
7.3 Backup and Restore Using a Tape Device..... 306
7.3.1 System Configuration 306
7.3.2 Processing Flow for Tape Device Backup 308
7.3.3 Backing up a SQL Server Database to Tape 311
7.3.4 Restoring a SQL Server Database from Tape 314
7.4 Backing Up a SQL Server Database Using a User Script..... 318
7.4.1 User Script Backup Processing Flow 318
7.4.2 User Script Example..... 320
7.4.3 Backup Execution Example 321
7.5 Operation Using SQL Server Transaction Logs..... 321
7.5.1 Configuring to Use the SQL Server Transaction Log..... 321
7.5.2 Example of Operation Using a Transaction Log Backup 323
7.5.3 Backing up a SQL Server Database..... 325
7.5.4 Backing up Transaction Logs..... 327
7.5.5 Restoring SQL Server Databases..... 327
7.5.6 Applying drmsqlrecovertool to Restore a SQL Server Database..... 327
7.5.7 Example of Operation Using Transaction Log Backup 329
7.5.8 Applying Transaction Logs to Recover Databases from Backup at a Particular Time 331
7.6 Backing up and Restoring Transaction Log Backup Files 333
7.6.1 Process Flow for Backing Up Transaction Log Backup Files..... 333
7.6.2 Backing up Transaction Logs..... 336
7.6.3 Restoring Transaction Logs 338
7.7 Backing up and Restoring Multiple Generations by Locking a Copy Group 340
7.7.1 Using Lock Functionality and Generation Management 341
7.7.2 Locking a Copy Group..... 342

7.7.3 Unlocking a Copy Group 344
7.8 Backing up and Restoring Between Remote Sites (Remote Copying) 345
7.8.1 Backing up a SQL Server Database to a Remote Site 348
7.8.2 Restoring a SQL Server Database to the Local Site 351
7.8.3 Restoring a SQL Server Database to a Remote Site 353
7.8.4 Recovering Data from the Remote Site to the Local Site 354
7.9 Backing up and Restoring in a Multi-Target Configuration 356
7.9.1 Backing up a SQL Server Database in a Multi-Target Configuration 360
7.9.2 Restoring a SQL Server Database in a Multi-Target Configuration 361
7.10 Performing Restoration and Recovery on a Host Other Than the Backup Host 363
7.11 Using the Log-Shipping Function of a SQL Server Database 368
7.11.1 Before Using the Log-Shipping Function 368
7.11.2 Setting a Distribution Target Server as an Operating Server 373
7.12 Backup and Restore in a Cascade or Multi-Target Configuration 375
7.12.1 Preparing to Backup or Restore in a Cascade or Multi-Target Configuration 377
7.12.2 Creating a User Script to Perform Backup in a Cascade Configuration 382
7.12.3 Backing up a SQL Server Database in Cascade Configuration 395
7.12.4 Backing up Transaction Logs in Cascade Configuration (No Backup Catalog) 398
7.12.5 Restoring a SQL Server Database in Cascade Configuration 398
7.13 Using a Backup Management Product That Cannot Link with Protection Manager . 403
7.14 Backing up and Restoring a SQL Server Database Using a File Share 405
7.14.1 Preparing to Back up or Restore Using a File Share 406
7.14.2 Backing up a SQL Server Database Using a File Share 406
7.14.3 Example of Restoring a Database Using a File Share 407
7.15 Restoring Data to a SQL Server Instance Different from the Backup Source 409
7.15.1 Process Flow for Restoring Data to a Different SQL Server Instance 409
7.15.2 Operation Flow 410
7.15.3 Recovery Operation Examples 411
7.16 Deleting Backup Data 413
7.17 Making a SQL Server Database Backed Up to a Secondary Volume Available 414
7.17.1 User Script Example 416
7.17.2 Data Mining and Backup Examples 418
7.18 Backing Up and Restoring When Using the SQL Server Replication Configuration .. 419
7.18.1 SQL Server System Configuration 419
7.18.2 Example When Using SQL Server Replication 421
7.18.3 Setting and Checking the sync with backup Option 423

Chapter 8 Using Protection Manager With Exchange Databases

8.1 Operations for Backing up and Restoring a Storage Group 426
8.1.1 Precautions When Backing up a Database 426
8.1.2 Precautions When Restoring a Database 427
8.1.3 Precautions on Recovery Storage Groups 430
8.1.4 Precautions When Configuring Volumes 430
8.1.5 Precautions on Database Verification 430
8.2 Backing up and Restoring Data Between Volumes 431
8.2.1 Processing Flow 432
8.2.2 Backing up Storage Groups to a Secondary Volume 433
8.2.3 Restoring Storage Groups to a Primary Volume 433
8.3 Backing up and Restoring Using a Tape Device 435
8.3.1 Backing up Storage Groups 436

8.3.2 Backing up Storage Groups to Tape 439
8.3.3 Restoring Storage Groups from Tape 444
8.4 Backing Up a Storage Group by Using a User Script 446
8.4.1 System Configuration 446
8.4.2 Processing Flow 446
8.4.3 User Script Example 447
8.4.4 Backup Execution Example 448
8.5 Restoring by Using Transaction Logs 449
8.5.1 System Configuration 451
8.5.2 Rolling Forward the Transaction Logs At the Time of Restoration 452
8.6 Backing up and Restoring Storage Groups Between Remote Sites (Remote Copy) .. 454
8.6.1 Backing up Storage Groups to a Remote Site 456
8.6.2 Restoring a Storage Group to Local Site 460
8.7 Using a Backup Management Product That Cannot Link with Protection Manager .. 463
8.8 Using File Sharing to Link the Database Server and Backup Server 465
8.8.1 Preparing to Back up or Restore Storage Groups by Using a File Share 465
8.8.2 Example of Using a File Share 465
8.8.3 Example of Restoring Storage Groups by Using a File Share 466
8.9 Deleting Backup Data 467

Chapter 9 Troubleshooting

9.1 Addressing Problems 470
9.2 Troubleshooting Extended Commands 471
9.3 Collecting Data When a Problem Occurs 472
9.3.1 Data To Be Collected (in Windows) 472
9.3.2 Log Files Output by Extended Commands 477
9.4 Collecting Required Data 480
9.4.1 Collecting Data 480
9.4.2 Adjusting the Output Level for Detailed Trace Log Information 482
9.4.3 Adjusting the Number of Log Files for Detailed Trace Log Information 482
9.5 Investigating Collected Data 483
9.5.1 Investigating Standard Output Logs 483
9.5.2 Investigating Trace Logs 483
9.5.3 Investigating the Detailed Trace Log 484
9.6 Troubleshooting Dictionary Map File Errors 487
9.6.1 Verifying Dictionary Map Files 488
9.6.2 Restoring Key Definition Files 489
9.6.3 Restoring Key Files 489
9.6.4 Restoring Missing Data Files 490
9.6.5 Restoring Incorrect Data Files 491
9.6.6 Recreating Dictionary Map Files 491
9.7 Calling the Hitachi Data Systems Support Center 491

Appendix A Sample HPtM Environment Configuration

A.1 System Configuration 494
A.1.1 Server Configuration 495
A.2 Example Settings for the RAID Manager Configuration Definition File 498
A.3 Example Settings for Protection Manager Configuration Definition File (init.conf) 501
A.4 Example Settings for the RAID Manager-Linkage Definition File (DEFAULT.dat) 502

A.5 Example of Creating Dictionary Map Files 503
A.6 Example of Registering SQL Server Information 504
A.7 Example Settings for Linkage to Backup Management Product 505
A.8 Example Settings for the Host Environment Settings File 507
A.9 Example Settings for an Operation Definition File 507
A.10 Example Settings for a Copy-Group List File 508

Appendix B Files Used in Protection Manager 509

Appendix C Replacing Primary and Secondary Volume Disks While Protection Manager Programs Are Running

C.1 Procedure for Replacing a Primary Volume Disk (in Windows) 513
C.2 Procedure for Replacing a Secondary Volume Disk 515

Appendix D Restoring from Tape to a Primary Volume in QuickShadow Configuration

D.1 QuickShadow Configuration 519
D.2 Backing up to Tape on the Backup Server 520
D.3 Restoring From Tape Directly to the Primary Volume 521

Appendix E Actions Taken for Message KAVX5105-W 523

Appendix F Actions Taken for Message KAVX5137-E

F.1 The cause of the message (KAVX5137-E) 525
F.2 Recovery procedure for message KAVX5137-E 526

Acronyms and Abbreviations 529

Index 533

List of Figures

Figure 1.1 Overview of Protection Manager Functions 4
Figure 1.2 Volume Consisting of More than One Copy Group 7
Figure 1.3 Data Management Using a Remote Site (Storage Subsystem) 9
Figure 1.4 Backing up More than One Generation 11
Figure 1.5 Backing up More than One Generation (Locking a Copy Group) 12
Figure 1.6 Backup and Restore Operations 13

Figure 2.1 System Configured to Back up and Restore Data Between Volumes 24
Figure 2.2 System Configured to Store Backed-up Data to Tape 25
Figure 2.3 Operation Standby Cluster Configuration 26
Figure 2.4 Sample Cluster Configuration of Mutual Standby Type (Active-Active) 27
Figure 2.5 TrueCopy or UR System Configuration 28
Figure 2.6 ShadowImage and TrueCopy Multi-Target Configuration 29
Figure 2.7 ShadowImage and UR Multi-Target Configuration 29
Figure 2.8 TrueCopy and UR Multi-Target Configuration 30
Figure 2.9 ShadowImage (Multi-Generation), TrueCopy, or UR Configuration Diagram 31
Figure 2.10 Sample Configuration for Concurrent Backup (Remote Copy) 32
Figure 2.11 Configuration to Restore Data on a Host Not Used for Backup 34
Figure 2.12 Configuration for the Log-Shipping Functionality 35
Figure 2.13 Cascade Configuration 36
Figure 2.14 Multi-Target Configuration for SQL Server Database Backup 37
Figure 2.15 Managing the P-VOL and S-VOL from the Same Server (Without Protection Manager) 38
Figure 2.16 When the Primary Volume for SI and the Primary Volume for TC Are the Same LDEV 40
Figure 2.17 When the Secondary Volume for SI and the Primary Volume for TC Are the Same LDEV 41
Figure 2.18 ShadowImage Cascade Configuration (Volume to Back up is Both Primary and Secondary) 42
Figure 2.19 TrueCopy-Async/UR Configuration: Restoring UserDB1 and UserDB2 Together or Separately 43
Figure 2.20 TrueCopy-Async/UR Configuration: Restoring UserDB1 and UserDB2 Together 43
Figure 2.21 Examples of Disk Set Configuration and Relationship Between a Disk Group and a Disk Set 47
Figure 2.22 Examples of Disk Group Configurations for Protection Manager 48
Figure 2.23 Data Backup and Restoration in a System Configuration With Multiple Disk Groups 49
Figure 2.24 Multiple Disk Group Configuration 50
Figure 2.25 Configuration Example of a Mounted Directory that Becomes a Backup Target 53
Figure 2.26 Configuration Example of a Mounted Directory that Does not Become a Backup Target 53
Figure 2.27 Example Configuration with Protection Product Using VSS to Back Up Data ... 58
Figure 2.28 Allocation of Database Configuration Files 68
Figure 2.29 When Multiple Database Configuration Files are Allocated to a Volume 69
Figure 2.30 Metafile Allocation 70
Figure 2.31 Metafile Allocation (in the Cluster Configuration) 71
Figure 2.32 Restoration in Units of Information Stores (Backup Target Is an Exchange Database) 77

Figure 4.1 Volume Configuration to Back up One Primary Volume to Multiple Generations95
Figure 4.2 Volume Configuration to Back up Multiple Primary Volumes to a Single Generation 96
Figure 4.3 Volume Configuration to Back up Multiple Primary Volumes to Multiple Generations 97
Figure 4.4 Example of a Configuration Definition for Backup or Restore Between Storage Subsystems 99
Figure 4.5 Configuration Definition for a Multi-Target and Cascade Configurations 101
Figure 4.6 Specifying a RAID Manager Instance Number in the Linkage Definition File (DEFAULT.dat) 114
Figure 4.7 Example of Creating a Dictionary Map File in an Active-Passive Cluster Configuration 132
Figure 4.8 Example of Creating a Dictionary Map File in an Active-Active Cluster Configuration 134
Figure 4.9 Cluster Group Definition When a Backup Target and Shared Disks are in the Same Group 138
Figure 4.10 Example of Deploying Operation Definition Files (Cluster Configuration, File System Backup) 182
Figure 4.11 Example of Locations of Operation Definition Files When the Backup is of an Exchange Database 187
Figure 4.12 Example of Locations of Operation Definition Files (When the Backup is of an Exchange Database) 192
Figure 4.13 Protecting Backup Information for Disk Backups 211
Figure 4.14 Protecting Backup Information for Tape Backups 213

Figure 5.1 Correspondence of Backup Catalog and Copy Group 224
Figure 5.2 Disk Group Configuration that Does Not Allow Backup 231

Figure 6.1 System Configuration for Backing up and Restoring Data Between Volumes (File System Backup) 249
Figure 6.2 Procedure for Backing up a File System to the Secondary Volume 250
Figure 6.3 Procedure for Restoring a File System to the Primary Volume 250
Figure 6.4 Configuration for Backup and Restore of a File System to/from Tape 253
Figure 6.5 Procedure for Backing up a File System to Tape 255
Figure 6.6 Procedure for Restoring a File System from Tape 256
Figure 6.7 System Configuration for Backup of a File System to Tape 262
Figure 6.8 Processing Flow 263
Figure 6.9 System Configuration for Backing up and Restoring a File System Between Remote Sites 268
Figure 7.1 Backing up and Restoring Multiple Databases (Same Volume Configuration) . 284

Figure 7.2 Backing up and Restoring Multiple Databases (Multiple Volume Configuration)285
Figure 7.3 Tape Restoring Multiple Databases (Multiple Volume Configuration) 286
Figure 7.4 Database Statuses During Restoration and Recovery 287
Figure 7.5 When `LOCAL BACKUP` is set to `NO` and the Secondary Volume on the Local Site is Backed up 289
Figure 7.6 Transaction Log Backups After Roll-Forward (1) 290
Figure 7.7 Transaction Log Backups After Roll-Forward (2) 291
Figure 7.8 Transaction Log Backups After Roll-Forward (3) 292
Figure 7.9 Order of Applying Transaction Log Backups (1) 293

Figure 7.10 Order of Applying Transaction Log Backups (2) 294
Figure 7.11 Backing up and Restoring Data Between Volumes (SQL Server Database Backup)300
Figure 7.12 Procedure for Backing up a SQL Server Database to the Secondary Volume .. 301
Figure 7.13 Procedure for Restoring a SQL Server Database to the Primary Volume 301
Figure 7.14 The `drmsqlrecovertool` Dialog Box 304
Figure 7.15 System Configuration for Backing up a SQL Server Database to Tape and Restoring from Tape 306
Figure 7.16 Procedure for Restoring a SQL Server Database to the Primary Volume 309
Figure 7.17 Procedure for Restoring a SQL Server Database from its Backup Tape 310
Figure 7.18 The `drmsqlrecovertool` Dialog Box 317
Figure 7.19 System Configuration for Backup of a SQL Server Database to Tape 318
Figure 7.20 Processing Flow 319
Figure 7.21 System Configuration for Operations Using the SQL Server Transaction Log .. 322
Figure 7.22 Operation Using Transaction Log Backups 324
Figure 7.23 Operation Using SQL Server Transaction Log Backup (Restoring a Database) . 329
Figure 7.24 Order of Applying Transaction Log Backups (1) 331
Figure 7.25 Order of Applying Transaction Log Backups (2) 332
Figure 7.26 Backing up SQL Server Transaction Log Backup Files 334
Figure 7.27 Example of Operation in Which a Particular Copy Group is Locked 341
Figure 7.28 System Configuration Example Used When SQL Server Databases Are Backed up and Restored Between Remote Sites 346
Figure 7.29 System Configuration for Backing up a SQL Server Database to Tape and Restoring from Tape in a Multi-Target Configuration 357
Figure 7.30 System Configuration for Restoring and Recovering from a Host not Used for Backup 363
Figure 7.31 Restoring and Recovering from a Host not Used for Backup (When Restoring) 364
Figure 7.32 System Configuration for Log-Shipping Operations 369
Figure 7.33 System Configuration for Setting a Distribution Target Server to be an Operating Server 373
Figure 7.34 Sample Configuration for Backing up to a Secondary Volume at a Remote Site (Cascade Configuration) 376
Figure 7.35 Points in Time at Which a User Script is Executed 382
Figure 7.36 Sample Cascade Configuration 387
Figure 7.37 Flow of Script-Specified Operations Performed When Local Backup is not Involved 389
Figure 7.38 Flow of Script-Specified Operations Performed When Local Backup is Involved392
Figure 7.39 System Configuration Using File Sharing to Back up and Restore a SQL Server Database 405
Figure 7.40 System Configuration for Restoring Data to a SQL Server Instance Different from the Backup Source 409
Figure 7.41 Restoring to a SQL Server Instance that is Different from the Backup Source (Using a User Script) 415
Figure 7.42 Configuration for Using the SQL Server Replication Function 419
Figure 7.43 Configuration of Backup Server for the Publisher/Distributor Server 420
Figure 8.1 Backing up and Restoring Data Between Volumes (Exchange Database backup)431

Figure 8.2 Processing Flow for Backing up Storage Groups to a Secondary Volume 432
Figure 8.3 Processing Flow for Restoring Storage Groups on the Primary Volume 432
Figure 8.4 System Configuration for Backing up and Restoring Storage Groups to and from Tape 435

Figure 8.5 Processing Flow for Backing up Storage Groups to Tape 437
Figure 8.6 Processing Flow for Restoring Storage Groups from Tape..... 438
Figure 8.7 System Configuration for Backup of a Storage Group to Tape..... 446
Figure 8.8 Processing Flow 447
Figure 8.9 Flow of Operation Using Transaction Logs 450
Figure 8.10 System Configuration for Restoring Data in Units of Information Stores by Applying Transaction Logs 451
Figure 8.11 System Configuration for Backing up and Restoring Storage Groups Between Remote Sites..... 455

Figure A.1 System Configuration for Protection Manager Environment Configuration 494

Figure E.1 Example of Replacing a Primary Volume Disk 513
Figure E.2 Example of Replacing a Secondary Volume Disk 516

List of Tables

Table 1.1 Required Storage Subsystem Volume Replication Functions 17
Table 1.2 Conventions Used in Syntax Explanations 20

Table 2.1 Types of Data Subject to Backup in Protection Manager for Exchange 44
Table 2.2 Volume Configuration When Backing up and Restoring Data 46
Table 2.3 Approximate Periods Required for Database Verification for Backup by Using VSS 62
Table 2.4 Types of Data Subject to Backup in Protection Manager for SQL 65
Table 2.5 Exchange data products supported by Protection Manager for Exchange 72
Table 2.6 Types of Data Subject to Backup in Protection Manager for Exchange 73
Table 2.7 Conditions for Allocating Files When Restoring Backup Data in Units of Information Stores 76
Table 3.1 Installation Destinations of Protection Manager Programs and Environment Variable Setup 82
Table 3.2 Error Cause Output to drm_inst.log 88
Table 3.3 Servers From Which Protection Manager Programs are Deleted 89

Table 4.1 Protection Manager Environment Setup Tasks Needed for Each Kind of Server 92
Table 4.2 Parameters in init.conf 104
Table 4.3 CLU_RETRY_WAIT Value for Each Type of Cluster Software 104
Table 4.4 Process Checking Parameters in init.conf 105
Table 4.5 Retry Parameters in init.conf 105
Table 4.6 Backup Option Parameter in init.conf 107
Table 4.7 Dictionary Map Synchronization Parameter in init.conf 109
Table 4.8 Recovery Option Parameter in init.conf 109
Table 4.9 Setting up Restoration Parameter in init.conf 110
Table 4.10 Cluster Type Parameters and Status in CLU_MSCS_RESTORE 110
Table 4.11 SQL Server Database Cluster Resource Status Parameters 110
Table 4.12 SQL Server Database Cluster Resource Status Parameters (No Resync Option) 111
Table 4.13 Cluster Resource Status Parameters for Exchange Database Backups 111
Table 4.14 Parameters to Specify an Instance Number in DEFAULT.DAT 113
Table 4.15 Parameters in DEFAULT.DAT for Retry Count and Interval 115
Table 4.16 RETRY_WAIT Parameter Values for Each Storage Subsystem 116
Table 4.17 Parameters in DEFAULT.DAT to Specify the Retry Count and Pair Status Check Interval 116
Table 4.18 Parameters to Specify a Retry Count and Interval at Busy Status 119
Table 4.19 Parameters in Copy Parameter Definition Files 122
Table 4.20 Specifying Track Size in DEFAULT.DAT 124
Table 4.21 Parameter in the RAID Manager-Linkage Definition DEFAULT.dat File for Automatic Pair Creation 126
Table 4.22 Parameter for Using the Secondary Volume Dynamic Recognition 127
Table 4.23 Parameter for Operation Mode When Automatically Selecting Copy Groups .. 129
Table 4.24 Parameter for Specifying the Installation Path for RAID Manager 130
Table 4.25 Specifying Modification of the Pair Status if an Error Occurs 130
Table 4.26 Directories that Must be Crated on a Shared Disk 137
Table 4.27 Parameters for the Cluster Product 139
Table 4.28 Relationship Between SQL_AUTORECOVERY_TIME Parameter and Restored Objects 144

Table 4.29 Parameter Defined in the sqlinit_instance-name.conf File 146
Table 4.30 Parameter Defined in the sqlinit_instance-name.conf File 146
Table 4.31 Default Settings for Windows services related to RM Shadow Copy Provider . 147
Table 4.32 vsscom.conf Parameters 148
Table 4.33 System Environment Variables Required to Use VSS (File Server or Database Server) 149
Table 4.34 System Environment Variables Required to Use VSS (Backup Server) 150
Table 4.35 System Environment Variables Required to Use VSS (File Server or Database Server) 150
Table 4.36 System Environment Variables Required to Use VSS (Backup Server) 152
Table 4.37 Parameters Defined in the Configuration Definition File for Tape Backup (For NetBackup)..... 154
Table 4.38 Parameters Defined in the Configuration File for Tape Backup (for Backup Exec)155
Table 4.39 Determining a Mount Point When the Command for Mounting a Secondary Volume is Executed With a Backup ID Specified 173
Table 4.40 Determining a Mount Point When the Command for Mounting a Secondary Volume is Executed..... 174
Table 4.41 Determining a Mount Target When the Command for Mounting a Secondary Volume is Executed..... 175
Table 4.42 Extended Command Execution Preparation Tasks Needed for Each Kind of Server 176
Table 4.43 Value and Maximum Value Length of Each Entry in a Host Environment Settings File 180
Table 4.44 Example: Contents and Maximum Value Length of Each Entry in an Operation Definition File When the Backup is of a File System 183
Table 4.45 Cluster Resources in a Cluster Configuration (Backup of File System) 184
Table 4.46 Operation IDs for Which an Operation Definition File is Created (Cluster Configuration, File System Backup)..... 185
Table 4.47 Contents and Maximum Value Length of Each Entry in an Operation Definition File When the backup is of a SQL Server Database 188
Table 4.48 Cluster Resources When the Backup is of a SQL Server Database..... 189
Table 4.49 Example of VDI Metafile Storage Directory 190
Table 4.50 Operation IDs for Which to Create Operation Definition Files When the Backup Is of a SQL Server and in a Cluster Configuration 190
Table 4.51 Specifications for Each Entry in an Operation Definition File When the Backup is of an Exchange Database 193
Table 4.52 Cluster Resources When the Backup is of an Exchange Database..... 194
Table 4.53 Operation ID for Which a Definition File Is Created on Backup of an Exchange Database in Cluster Configuration 195
Table 4.54 Naming Conventions for the Temporary Directory for Extended Commands.. 196
Table 4.55 Directories Used for Inter-Server FTP Transmission and Reception 197
Table 4.56 Coding Rules for User Scripts 198
Table 4.57 Entries in User Scripts..... 198
Table 4.58 Script Environment Variables 203
Table 4.59 Commands That can Be Specified in User Pre-processing and User Post-processing 205
Table 4.60 Examples of User Script Usage..... 207
Table 4.61 mail.conf Parameters 208

Table 5.1 Correspondence Between Extended Commands and Basic Commands (Used for Backing up and Restoring File Systems) 219
Table 5.2 Correspondence Between Extended Commands and Basic Commands (Common Commands) 219
Table 5.3 Correspondence Between Extended Commands and Basic Commands (Tape Commands) 220
Table 5.4 Correspondence Between Extended and Basic Commands (Utility Commands)221
Table 5.5 Correspondence Between Extended and Basic Commands (Protection Manager for SQL)..... 221
Table 5.6 Correspondence Between Extended and Basic Commands (Protection Manager for Exchange)..... 222
Table 5.7 Required Permissions for Commands Used to Back up and Restore a File System233
Table 5.8 Required Permissions for Common System Commands 234
Table 5.9 Required Permissions for Tape System Commands 235
Table 5.10 Required Permissions for Utility Commands 235
Table 5.11 Required Permissions for Commands Used to Back up a SQL Server Database 236
Table 5.12 Required Permissions for Commands Used to Back up an Exchange Database 236
Table 5.13 Concurrent Command Execution When Backing up and Restoring a File System237
Table 5.14 Concurrent Execution of Common Commands 238
Table 5.15 Concurrent Execution of Tape-Related Commands 239
Table 5.16 Concurrent Execution of Utility Commands 239
Table 5.17 Concurrent Command Execution When Backing up a SQL Server Database 240
Table 5.18 Concurrent Command Execution When Backing up an Exchange Database 240
Table 5.19 Functions that Require that the Protection Manager Service Be Running..... 243

Table 6.1 User Script Creation Example 264

Table 7.1 Service Status 294
Table 7.2 SQL Server Database Status Where a Command Can Be Executed (1) 295
Table 7.3 SQL Server Database Status Where a Command Can Be Executed (2) 297
Table 7.4 Relationship Between the Type of SQL Server Database and Executable Commands 299
Table 7.5 User Script Creation Example 320
Table 7.6 Coding Rules for User Script Files 383
Table 7.7 Entries in User Script Files (continues on following pages)..... 384
Table 7.8 Example of Writing a Sample Script When Local Backup is not Involved..... 390
Table 7.9 Example of Writing a Sample Script When Local Backup is Involved..... 393

Table 8.1 User Script Creation Example 448

Table 9.1 OS Log Information for Windows Systems 472
Table 9.2 Protection Manager Information 473
Table 9.3 RAID Manager Information 475
Table 9.4 Database Information (When the Backup is of a SQL Server Database or Exchange Database) 476
Table 9.5 Information Pertaining to Protection Manager Console 476
Table 9.6 Items Output to the Log File for Extended Commands 478
Table 9.7 Items Output to the Detailed Trace Log 484
Table 9.8 Sources of Messages (Controlled Objects) and Corresponding Message IDs 485

Table A.1 Server Configuration for Protection Manager Environment Configuration).... 495
Table A.2 Storage Subsystem Configuration for Protection Manager Environment Configuration 496
Table A.3 Storage Subsystem Volume Configuration for Protection Manager Environment Configuration 496
Table A.4 Examples of Shared Directory Usage for Protection Manager Environment Configuration) 497

Table B.1 List of Files Used in Protection Manager 509

Chapter 1 Overview of Protection Manager

Protection Manager software systematically controls storage subsystems, backup/recovery products, databases, and other system components to provide efficient and reliable data protection, easing the workload of system administrators. This chapter describes the features and functions of Protection Manager, and describes system requirements for use.

- Introduction to Protection Manager (see section 1.1)
- Protection Manager Features (see section 1.2)
- Protection Manager Functions (see section 1.3)
- Protection Manager System Requirements (see section 1.4)
- Backup Management and Cluster Software (see section 1.5)
- Conventions for Syntax Explanations (see section 1.6)

1.1 Introduction to Protection Manager

This section introduces Protection Manager programs and their functions.

1.1.1 Standard Functions of Protection Manager

Protection Manager Copy Controller provides the basic functions of Protection Manager. Copy Controller serves as a base for optional program products and for performing backup or restore operations on file system data.

1.1.2 Optional Functions for Operations on Databases Created by a DBMS

- Protection Manager for SQL is used for backup or restore operations on SQL Server databases.
- Protection Manager for Exchange is used for backup or restore operations on Exchange databases.

1.1.3 Making Protection Manager Functions Available From a GUI

Protection Manager Console enables you to use the functions provided by Protection Manager from GUI windows. You can visually check and manipulate the volume information of the backup targets or the backup catalog to achieve efficient system operation. This product supports operations on file systems, Exchange databases, or SQL Server databases in Windows® environments. For details on Protection Manager Console, see the *HiCommand® Protection Manager Console User's Guide*.

Protection Manager Console also enables you to use Setup GUI. Setup GUI is a function that enables you to use the GUI to set the execution environment required for operating Protection Products. Since the GUI is used, you do not need to execute any commands or edit any configuration files. You simply set the required items in the proper order. For details on how to use Setup GUI to set up an environment, see the *HiCommand Protection Manager Console User's Guide*.

1.2 Protection Manager Features

An enterprise information system must be able to safely protect important data without interrupting other processing jobs, and restart such processing jobs quickly if an error occurs. To meet these needs, the storage subsystems of the TagmaStore, Lightning, and Thunder Series provide RAID management (through RAID Manager) and volume replication functions (ShadowImage, etc.). These replication functions can quickly create a replica of a logical volume within a storage subsystem, without using a host. A system administrator who manages backups can use these functions to back up a replica (secondary volume) of a primary volume to tape without disconnecting the primary volume, which can continue the current processing job.

Until now, extensive expertise in databases, file systems, the logical volume manager, and RAID devices was required to use volume replication functions to back up a database during online operations (an online backup). Furthermore, the operation for doing so was complex and vulnerable to human error. Because backup operations differ depending on the type of database application or file system, the learning process was also difficult.

Protection Manager programs resolve these problems to enable data management using simple operations without complex procedures. Protection Manager protects important data with mimimal interruption of job processing while providing high-speed, reliable backup and restore operations using TagmaStore/Lightning/Thunder Series functions.

Protection Manager programs internally manage information about the relationships between the backup target object and the associated logical unit among the RAID devices, the relationship between the primary and secondary volumes, and the backup history. This information is contained in a dictionary map file. Protection Manager programs reference the information in the dictionary map file to control the database and RAID devices automatically, thus reducing the system administrator’s workload and the storage system’s total cost of ownership (TCO).

Figure 1.1 provides an overview of Protection Manager functions.

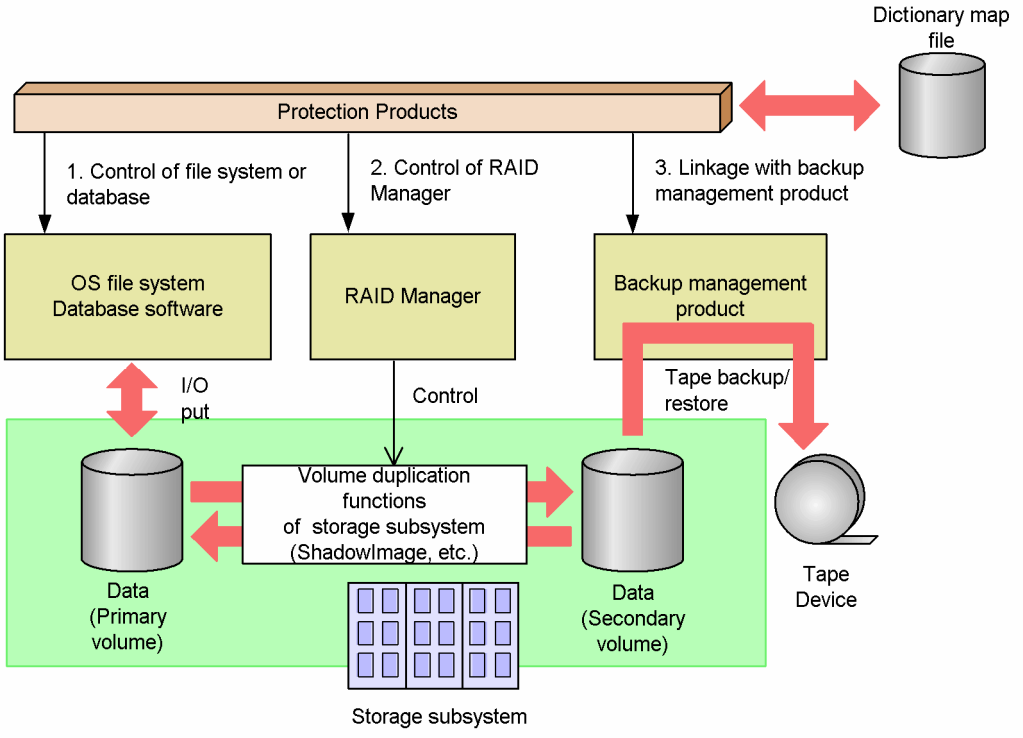


Figure 1.1 Overview of Protection Manager Functions

The following describes how Protection Manager programs control other software components. The numbers shown below correspond to those in the figure:

1. Protection Products programs bring file systems or databases that exist in storage subsystems to a quiescent state to ensure reliable backup operations.
2. Protection Manager programs control RAID Manager to perform volume backup or restore operations by using the volume replication functions of the storage subsystems.
3. Protection Manager programs operate in conjunction with backup management products to control tape backup or restore operations.

Note: Internally, Protection Products use the command line interface (CLI) function of RAID Manager and a backup management product. To use Protection Products, these products must have already been set up and must be ready to use.

1.3 Protection Manager Functions

This section describes key functions and benefits of Protection Manager.

1.3.1 Backup Commands

Protection Manager programs provide two backup commands. One backup command uses the volume replication function of the storage subsystem to perform high-speed disk copying. The other backup command backs up data from the secondary volume to tape in collaboration with the backup management product. You can use the backup command most suitable for specific backup targets such as file systems or database products to simplify backup operations, which might otherwise require extensive expertise.

By providing a user script file, you can perform a series of operations to back up data from a primary volume (via a secondary volume) to tape by using a single backup command.

1.3.2 Restore Command

Protection Manager programs provide a command that uses volume replication functions to restore a database or file system from a backup. You can use the restore command most suitable for a specific backup target such as a file system or database product to simplify restore operations, which might otherwise require extensive expertise.

1.3.3 Dictionary Map File

Protection Manager programs detect configuration definition information required for backup or restore operations, such as information about file systems and databases, and disk management information about RAID devices. This information is then used to create a file called the dictionary map file. Protection Manager programs refer to the dictionary map file while controlling RAID Manager, thus reducing the workload of the system administrator.

The dictionary map file consists of the following files:

Application map file

This file stores information about relationships between backup-target database objects (such as journal logs) and their associated files.

Core map file

This file stores information about relationships between mount-point directories for file systems and disk numbers across RAID devices.

Copy group map file

This file stores information about mapping between primary and secondary volumes. Protection Manager programs can have up to three generations of secondary volumes for each primary volume.

Backup catalog

The backup catalog stores the backup operation history of Protection Manager and information for generation management. When a backup is performed, a new record containing information about the backup is created in the backup catalog. When a Protection Manager program restores data from a backup, the program reference information in the backup catalog includes:

- Backup ID
- Backup start date and time
- Backup source information
- Backup destination information

The *backup ID* is an ID used to identify backup data. Protection Manager programs automatically assign a 10-digit backup ID when data is backed up, and register the ID in the backup catalog.

A new backup ID is assigned whenever data is backed up. When there are multiple sets of backup data because you are using generation management that associates multiple secondary volumes with a single primary volume, or because you have backed up data from secondary volumes to tape, you can restore the data by specifying the backup ID. If you want to identify backup data by use of a fixed character string, you can also specify a backup comment.

If you execute the Protection Manager commands in a cluster configuration, you must specify the location of the dictionary map file to be used for each virtual server. For details on cluster configurations, see section 1.3.5. For details on how to create the dictionary map file in a cluster configuration, see section 4.6.

1.3.4 Pair Management Using Copy Groups

A copy group is a pair of primary and secondary volumes for which synchronization is started or stopped by the volume replication functions or the RAID Manager functions (pair volume). Figure 1.2 shows a volume that consists of more than one copy group.

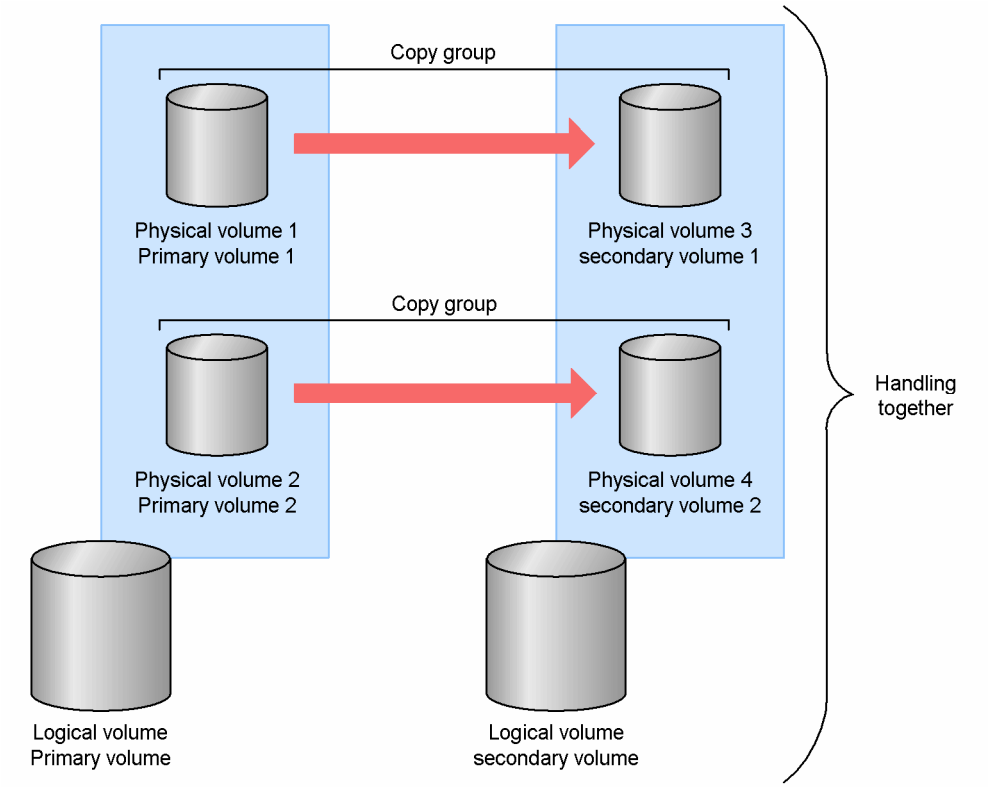


Figure 1.2 Volume Consisting of More than One Copy Group

If the logical volume containing the data you want to back up consists of more than one physical volume, create copy groups (pair volumes) using the primary and secondary physical volumes. Protection Manager programs handle more than one copy group together to start or stop synchronization of the entire logical volume to be backed up.

The name of a copy group consists of a RAID Manager group name (*dev_group*) and a RAID Manager pair logical volume name (*dev_name*). For example, if the *dev_group* is `VG01` and the *dev_name* is `dev01`, the copy group name must be `VG01,dev01`.

1.3.5 Support for Clustering

Protection Manager programs support failover clustering based on cluster software for a variety of operating systems. Applying the cluster configuration to file servers and database servers allows you to continue a backup operation without stopping the system if a failure occurs.

Protection Manager programs enable you to automatically control clusters to take advantage of high-availability systems based on cluster configurations. When a database is taken offline to perform a cold backup or restoration while the cluster resources that monitor the DBMS are still online, a failover usually occurs. This requires controlling cluster resources manually. However, when data is backed up in the cold backup mode or restored using Protection Manager, cluster resources that monitor the DBMS are automatically taken offline and then put back online once processing finishes.

Protection Manager programs support mutual standby cluster configurations (active-active) as well as operation standby cluster configurations (active-passive).

For details on cluster configurations, see sections 2.3 and 2.4.

In a cluster environment using MSCS, you can select whether to perform restoration while the cluster resources that monitor the DBMS are in the online state or offline state. In a restoration in which the cluster resources are in the offline state, you need to place offline all databases that share the DBMS services. However, in a restoration in which the cluster resources are in the online state, you can restore backup data by placing offline only those databases for which backup data is to be restored.

Whether you can select restoration while the cluster resources are in the online state depends on the storage subsystem type or the OS functionality.

For details on how to select restoration while the cluster resources are in the online state, see section 4.3.7.

1.3.6 Commands to Reduce Operation Workload

Protection Manager programs provide two types of commands to reduce operation workload:

- Basic commands are used for backup, restore, or other operations on backup data, and for setting up Protection Manager environments.
- Extended commands are scripts that automatically execute a series of commands. Extended commands are provided by Protection Manager programs, and make it easier to perform backup tasks using Protection Manager commands.

Execute basic commands in combination to perform Protection Manager operations. For example, to perform a tape backup you must execute two commands, a basic command for mounting to the secondary volume and another command for backing up data on the secondary volume into a tape device.

1.3.7 Data Management Using a Remote Site (Storage Subsystem)

Protection Manager programs allow you to perform data management on a remote site (storage subsystem), such as backing up data to a remote site, or restoring data from a remote site.

Protection Manager programs enable backup to and restoration from a secondary volume both in a local site (which is the storage subsystem where the primary volume resides) and a remote site (which is a storage subsystem different from the one where the primary volume resides).

Figure 1.3 illustrates the relations between physical volumes and a local site or a remote site.

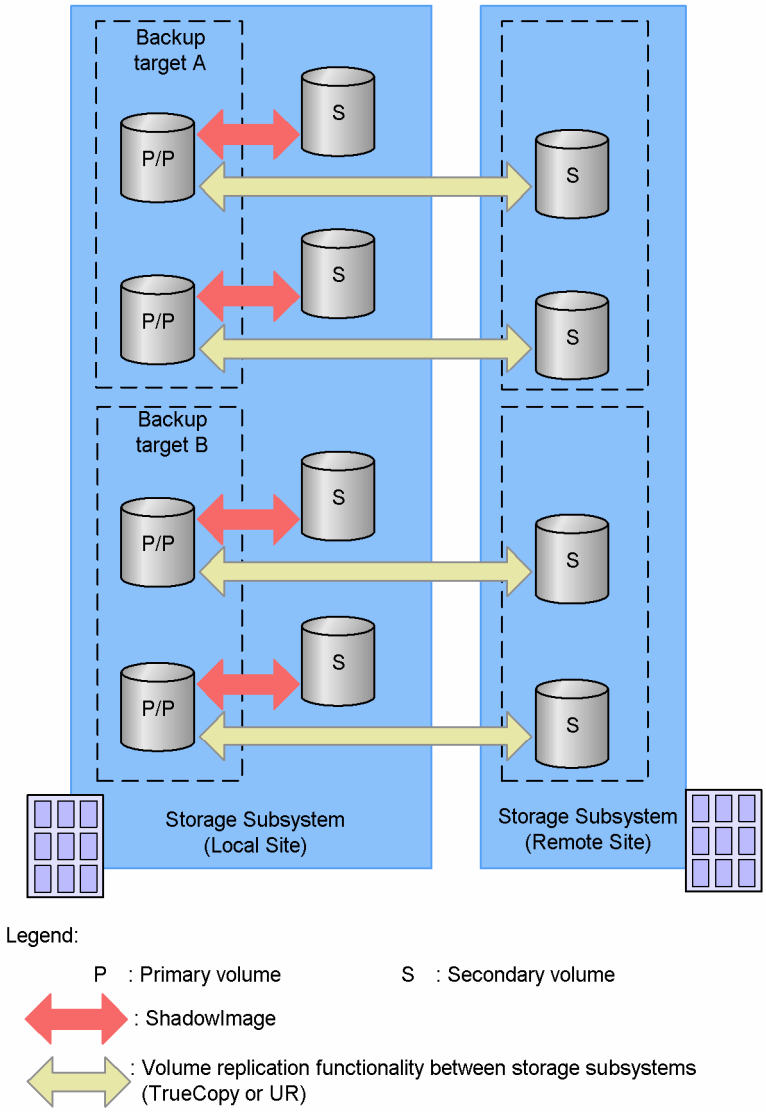


Figure 1.3 Data Management Using a Remote Site (Storage Subsystem)

By using a remote site (storage subsystem), you can use a secondary volume in a storage subsystem that is physically different from the storage subsystem of the primary volume to back up or restore data. Protection Manager programs use TrueCopy or UR to manage backup or restoration operations on a remote site.

In this manual, the volume copy functionality that replicates a volume to a remote site, such as TrueCopy or UR, is generically called volume copy between storage subsystems (or remote copy functionality). By backing up data to a remote site, you can restore the data from the remote site if a problem occurs in a local site storage subsystem. This improves data stability. For example, even if the data of a local site is damaged due to a disaster, you can use the data in a remote site to recover it in a short period (disaster recovery).

Protection Manager programs can perform the following operations on a remote site:

- Back up and restore data
- Re-synchronize a copy group
- Display resource information
- Lock a copy group
- Back up data into a tape and restore data from a tape
- Mount or unmount a secondary volume

1.3.8 Managing Generations of Secondary Volumes

Protection Manager programs can have multiple secondary volumes for each primary volume, and manage more than one secondary volume generation.

The following two methods can be used to select a secondary volume to be used as backup destination: Protection Manager programs automatically select a secondary volume, or the user selects a desired volume. The first method is available only when the volume copy within a storage subsystem is used. To allow a user to select a secondary volume as a backup destination, use a *name identifying the generation*. Protection Manager programs create this name in a following format, and you specify such a name when you back up data:

`local_`*`MU#`* `or remote_`*`MU#`*

where:

`local`: Specify this when the volume copy within a storage subsystem is used.

`remote`: Specify this when a volume copy between storage subsystems is used.

`MU#`: The MU number defined in the RAID Manager configuration definition file (`horcmn.conf`)

For example, when you back up data to a secondary volume of TrueCopy, the name identifying the generation is `remote_0`.

In this example, generations 1 to 3 are backed up to secondary volumes 1 to 3 respectively, and generations 4 or more are backed up again from the secondary volume 1.

Figure 1.4 shows an example of backing up more than one generation.

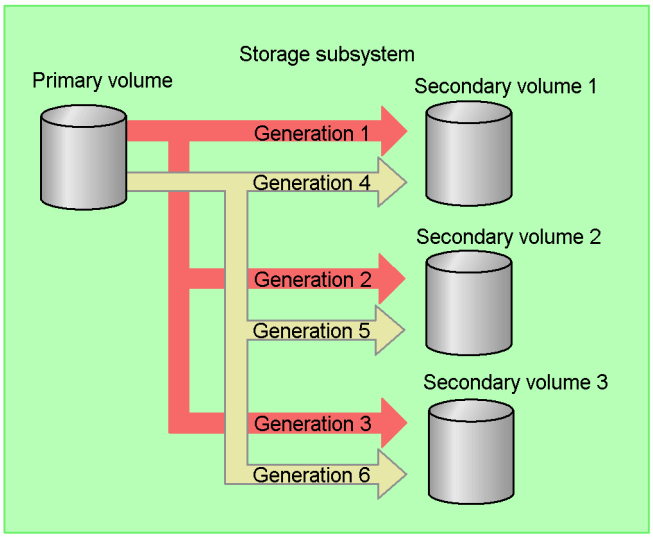


Figure 1.4 Backing up More than One Generation

Normally, Protection Manager programs automatically determine the secondary volume used as the backup destination. However, you can use a Protection Manager programs command to keep (lock) the contents in a specific secondary volume, and continue backup operations using other secondary volumes only.

In this example, after generation 1 is obtained, the copy group of generation 1 is locked to keep the contents of the backup for the generation 1. As a result, the remaining secondary volumes (secondary volume 2 and 3) are used to back up generation 4 and following generations. Figure 1.5 illustrates the backup of more than one generation when a specific copy group is locked.

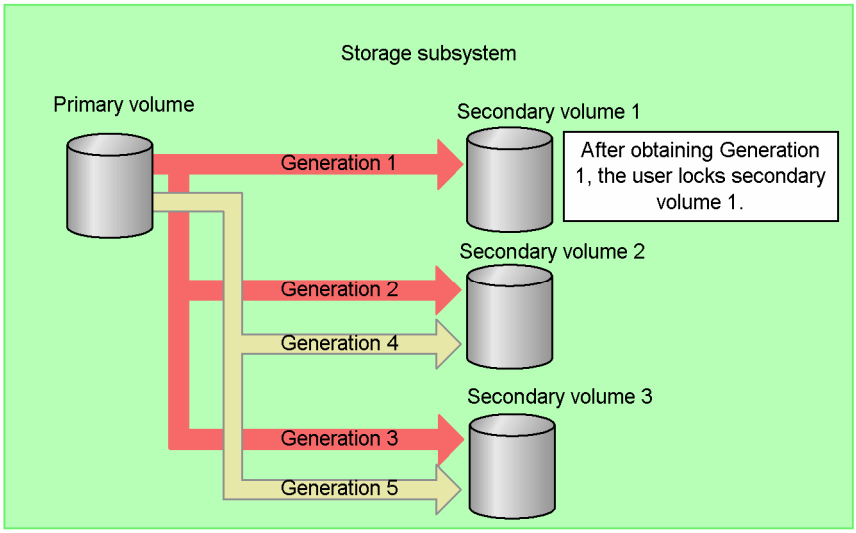


Figure 1.5 Backing up More than One Generation (Locking a Copy Group)

When you are operating multiple generations, the sequence in which the secondary volumes are used for backup does not change regardless of which generation you restore. The number of generations that Protection Manager can manage depends on the volume replication functions. As shown in Figure 1.6, the backup and restore operations are the same regardless of the number of generations.

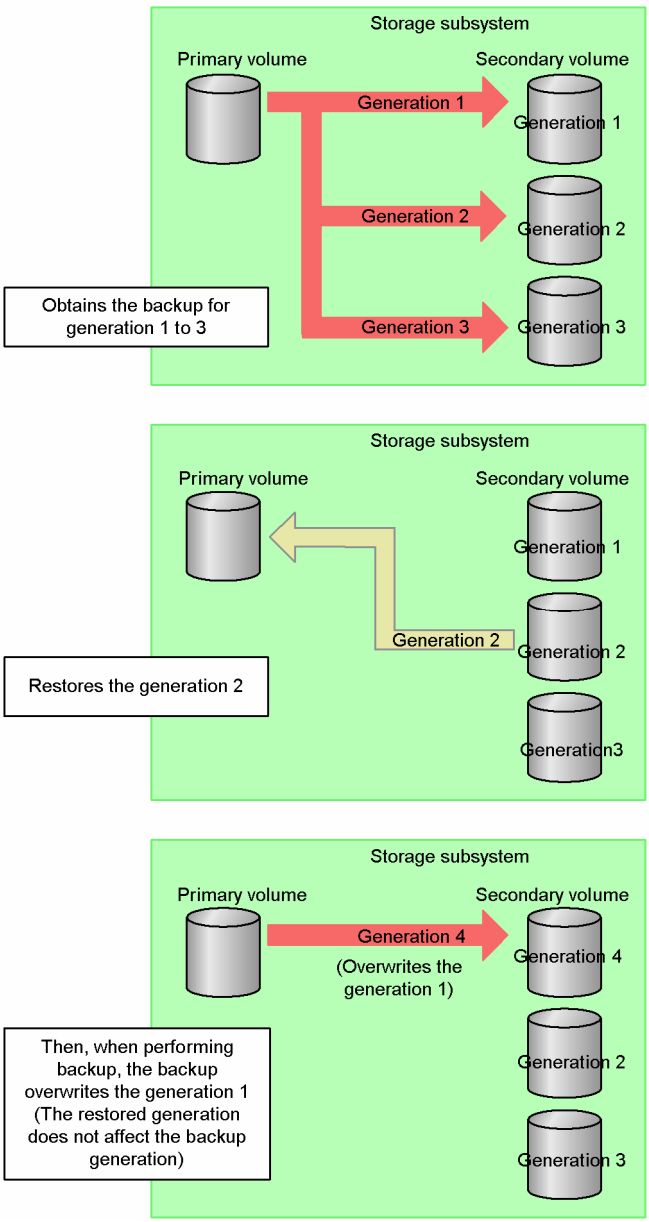


Figure 1.6 Backup and Restore Operations

1.4 Protection Manager System Requirements

This section describes the host, server, and storage subsystem conditions required when you use Protection Products to configure a system, and describes user permissions required for executing commands.

1.4.1 Host Requirements

Protection Manager supports the following Windows® operating systems:

- Windows 2000 Server
  - Microsoft® Windows 2000 Server Operating System
  - Microsoft Windows 2000 Advanced Server Operating System
  - Microsoft Windows 2000 Datacenter Server Operating System
- Windows Server™ 2003
  - Microsoft Windows Server 2003, Standard Edition Operating System
  - Microsoft Windows Server 2003, Enterprise Edition Operating System
  - Microsoft Windows Server 2003, Datacenter Edition Operating System
  - Microsoft Windows Server 2003, Standard x64 Edition Operating System
  - Microsoft Windows Server 2003, Enterprise x64 Edition Operating System
  - Microsoft Windows Server 2003, Datacenter x64 Edition Operating System
- Windows Server 2003 R2
  - Microsoft Windows Server 2003 R2, Standard Edition Operating System
  - Microsoft Windows Server 2003 R2, Enterprise Edition Operating System
  - Microsoft Windows Server 2003 R2, Datacenter Edition Operating System
  - Microsoft Windows Server 2003 R2, Standard x64 Edition Operating System
  - Microsoft Windows Server 2003 R2, Enterprise x64 Edition Operating System
  - Microsoft Windows Server 2003 R2, Datacenter x64 Edition Operating System

Windows Server 2003 hereafter refers to Windows Server 2003 and Windows Server 2003 R2.

The following applications must be installed and set up before installing Protection Manager:

- RAID Manager

  Because Protection Products internally use the command line interface of RAID Manager (CLI) function, RAID Manager must have already been set up and must be ready to use.
- Logical volume manager (LVM). The following LVM products are supported:
  - Standard logical disk manager (LDM) of the operating system
  - VERITAS Volume Manager for Windows#

  # Protection Products do not work with the VERITAS Dynamic Multiple Pathing function.

1.4.2 Server Requirements

The following are the requirements for operating the server.

- Setting Environment Variables

If the OS is Windows, do not set HORCMINST or HORCC_MRCF in the system environment variable. Also, do not set HORCM_CONF or HORCMPERM in the system environment variable or the user environment variable.

- About the versions of the OS and Protection Products

When the OS is not Windows, make sure that the OS version of the file server or database server is the same as that of the backup server.

Use the same version of Protection Manager on the file server or database server and on the backup server.

1.4.3 Storage Subsystem Requirements

Protection Manager supports the following storage subsystems:

- Hitachi TagmaStore Universal Storage Platform (USP)
- Hitachi TagmaStore Universal Storage Platform NSC55 (NSC)
- Hitachi Lightning 9900™ V Series (9900V)
- Hitachi Lightning 9900
- Hitachi TagmaStore Adaptable Modular Storage (AMS)
- Hitachi Thunder 9500™ V Series (9500V)
- Hitachi Thunder 9200™
- Hitachi TagmaStore Workgroup Modular Storage series

Volume replication functionality is required to use Protection Manager with any of the above storage subsystems.

Table 1.1 shows supported volume replication functions.

Note: The command device for RAID Manager must be configured on the storage subsystem.

Table 1.1 Required Storage Subsystem Volume Replication Functions

Storage Subsystem	Volume Replication Function
TagmaStore Universal Storage Platform TagmaStore Universal Storage Platform NSC55	Hitachi ShadowImage[1] Hitachi TrueCopy[2] Hitachi TrueCopy Asynchronous[2] Copy-on-write SnapShot[3] Hitachi Universal Replicator[4]
Lightning 9900V	Hitachi ShadowImage[1] Hitachi TrueCopy[2]
Lightning 9900	Hitachi ShadowImage Hitachi TrueCopy
TagmaStore AMS Series TagmaStore WMS Series	ShadowImage[1] TrueCopy[2] Copy-on-write SnapShot[3]
Thunder 9500V Series	Hitachi ShadowImage[1] Hitachi QuickShadow[3] Hitachi TrueCopy Basic[2]
Thunder 9200	Hitachi ShadowImage Hitachi TrueCopy

[1]In this manual, this name is abbreviated to ShadowImage.

[2]In this manual, this name is abbreviated to TrueCopy.

[3]In this manual, this name is a abbreviated to QuickShadow.

[4]In this manual, this name is abbreviated to UR.

1.4.4 User Authorization at Command Execution

The following authorizations are required for executing Protection Manager commands:

- OS administrator authorization

  A user executing a Protection Manager command must have local Administrator authorization on the server on which the command is to be executed.

- Database access authorization
  - When backing up a SQL Server database:

    When a user executes a Protection Manager command, the user is authenticated by Windows before access to the SQL Server is granted. Therefore, a user executing a Protection Manager command must have been registered as a member having the sysadmin fixed server role for a SQL Server.
  - When backing up an Exchange database:

A user executing a Protection Products command must have been assigned to the Enterprise Admins group or the Exchange Domain Servers group of a domain.

For details on authorization required for each command, see section 5.4.1.

1.5 Backup Management and Cluster Software Products

Protection Manager can enhance system functionality by linking with backup management and cluster software, and automating tape backups. This section describes software products that can be linked to Protection Manager.

The following backup management products can be used with Protection Products:

- NetBackup
- Backup Exec

The following cluster software products can be used with Protection Manager:

- Microsoft Cluster Service* or VERITAS Cluster Server

  *The standard and majority node set (MNS) quorums are available in Microsoft Cluster Service.

1.5.1 Database Products

To use the Protection Manager for SQL and Protection Manager for Exchange options, you must first install Protection Manager Copy Controller on the host where you install the options. The operating environment (32- or 64-bit) of an optional product must match that of Protection Manager - Copy Controller. For example, if Protection Manager for Exchange Server 2007 is used, Protection Manager - Copy Controller (x64) must be installed.

The following database products can be used with Protection Manager:

- Exchange:
  - Microsoft Exchange 2000 Enterprise Server
  - Microsoft Exchange 2000 Server
  - Microsoft Exchange Server 2003 Enterprise Edition
  - Microsoft Exchange Server 2003 Standard Edition
  - Microsoft Exchange Server 2007 Enterprise Edition
  - Microsoft Exchange Server 2007 Standard Edition

  Note: Protection Manager for Exchange uses Microsoft Collaboration Data Objects for Exchange 2000 Management (also "collaboration data object for exchange management" or CDOEXM). Exchange Server system management tools must be installed before installing the files necessary for executing commands.

- SQL:
  - Microsoft SQL Server™ 2000 Enterprise Edition
  - Microsoft SQL Server 2000 Standard Edition
  - Microsoft SQL Server2005 Enterprise Edition
  - Microsoft SQL Server 2005 Standard Edition
  - Microsoft SQL Server 2005 Workgroup Edition

1.6 Conventions for Syntax Explanations

Table 1.2 lists conventions used in syntax explanations. Syntax definitions appear as follows:

`StoreDatabase` `[A|B] {C|D|E} (`*`database-name`* `...)`

Table 1.2 Conventions Used in Syntax Explanations

Example font or symbol	Convention
`StoreDatabase`	The user should enter code-font characters exactly as shown.
`database-name`	In actual commands, the user must replace the italicized information with suitable characters.
`SD`	Bold code-font characters indicate abbreviations for commands.
A	Underlined characters are the system default when you omit all the items enclosed in brackets. ***Example:*** `[A|B]` indicates that the system uses `A` if you do not specify either `A` or `B`.
|	Only one of the options separated by a vertical bar can be used at one time. ***Example:*** `A|B|C` indicates `A` or `B` or `C`.
{}	One of the items enclosed in braces and separated by a vertical bar must be specified. ***Example:*** `{C|D|E}` indicates that you must specify one of the items from `C`, or `D`, or `E`.
`[ ]`	Items enclosed in brackets are optional. ***Examples:*** `[A]` indicates that you can specify `A` or nothing. `[B|C]` indicates that you can specify B or C or nothing.
`...`	The item or items preceding ellipses (...) can be repeated. To specify multiple items, use a one-byte space to delimit them. ***Example:*** `A B ...` indicates that you can specify B as many times as necessary after A.
`()`	The items enclosed by the parentheses are in the range to which | or ... are applied.

This manual uses terms from the English version of the Windows OS. If you are not using the English version of Windows, substitute the following terms by the corresponding terms in your language version of Windows:

- Administrator
- All Users
- Application Data
- Documents
- Documents and Settings
- Program Files
- System32

Chapter 2 System Configurations for Protection Manager

This section describes the primary configuration patterns used for systems based on Protection Manager. It also describes requirements for any database subject to backup and precautions to be taken before running Protection Manager.

- Configuring Data Back up and Restoration Between Volumes (see section 2.1)
- Configuring Systems to Store and Restore Data Backed up to Tape (see section 2.2)
- Executing the Operation Standby Cluster Configuration (see section 2.3)
- Executing the Mutual Standby Cluster Configuration (see section 2.4)
- Configurations for Data Backup and Restoration Between Storage Subsystems (Remote Sites) (see section 2.5)
- Precautions for Data Backup and Restoration Between Storage Subsystems (see section 2.6)
- Restoring Data on a Host Not Used for SQL Server Database Backup (see section 2.7)
- Configuring the System to Use SQL Server's Log-Shipping Function (see section 2.8)
- Cascade Configuration for a SQL Server Database Backup (see section 2.9)
- Using a Multi-Target Configuration for a SQL Server Database Backup (see section 2.10)
- Requirements When Using RAID Manager to Configure a Pair Volume (see section 2.11)
- Server Operation Requirements (see section 2.12)
- Configuration for Backing up Data Using VSS (see section 2.13)
- Requirements and Precautions for a Database Subject to Backup (see section 2.14)

2.1 Configuring Data Backup and Restoration Between Volumes

To back up or restore data between a primary volume and a secondary volume, connect a file server or a database server to the storage subsystem.

In this configuration, install the Protection Manager programs in the file server or the database server, and then execute backup and restoration commands.

An example of a system configured to back up and restore data between volumes is shown in Figure 2.1. A file server or database server allows you to manage only the primary volume.

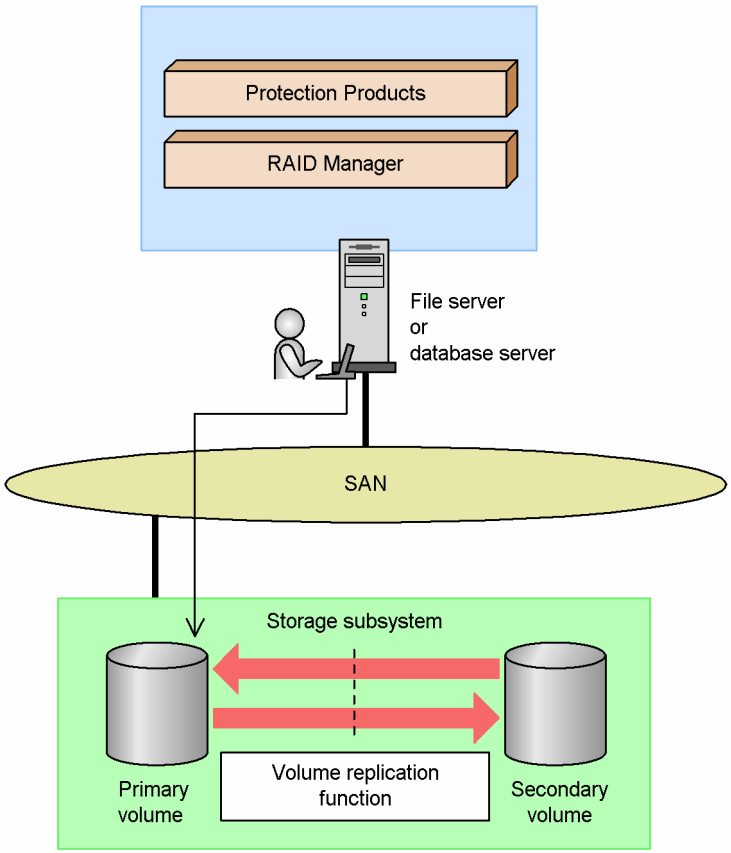


Figure 2.1 System Configured to Back up and Restore Data Between Volumes

2.2 Configuring Systems to Store and Restore Data Backed Up to Tape

It is possible to configure a system to store backed-up data to tape and to restore data from tape.

To store data backed up to tape, the system requires two servers: a file server or a database server that manages the primary volume, and a backup server connected to the tape device.

Install Protection Manager programs both on the file server or database server and on the backup server, and execute commands for the primary volume from the file server or database server, and then for the secondary volume from the backup server.

Figure 2.2 shows a system configured to store backed-up data to tape. A file server or database server allows you to manage only the primary volume. A backup server allows you to manage only the secondary volume.

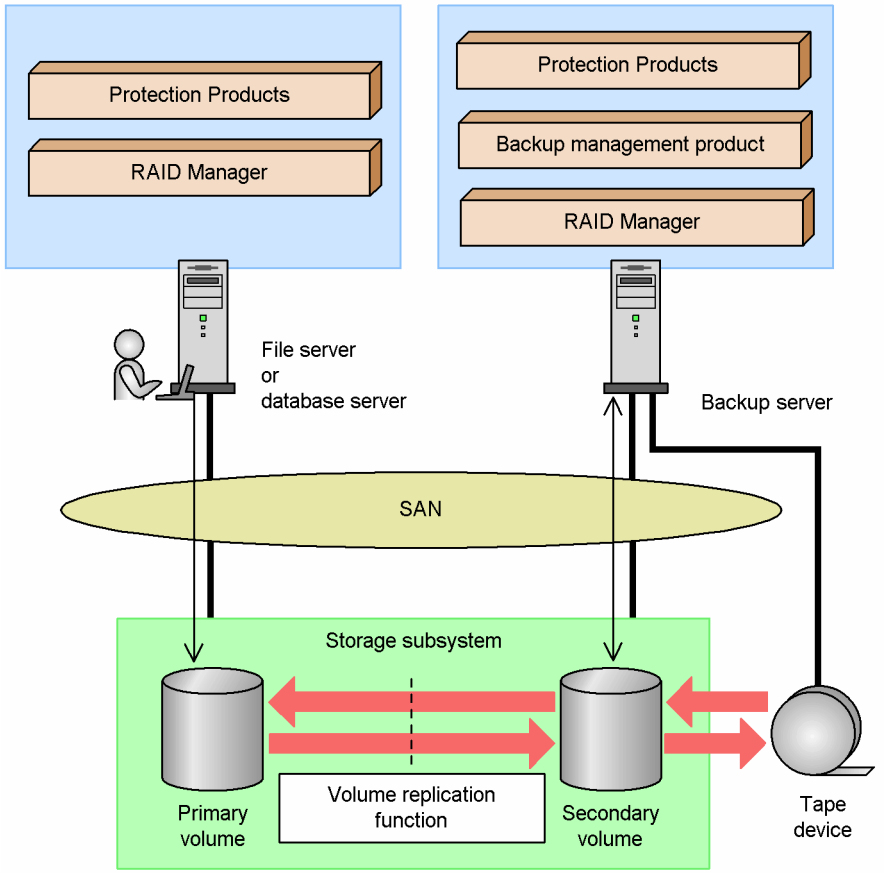


Figure 2.2 System Configured to Store Backed-up Data to Tape

2.3 Executing the Operation Standby Cluster Configuration

Protection Manager programs can handle a two-node operation standby cluster configuration (active-passive) on a database server and file server.

Figure 2.3 shows an example of an operation standby (active-passive) cluster configuration. You can build a cluster configuration using file servers or database servers. You cannot allocate one cluster server to a file server or database server and the other cluster server to a backup server.

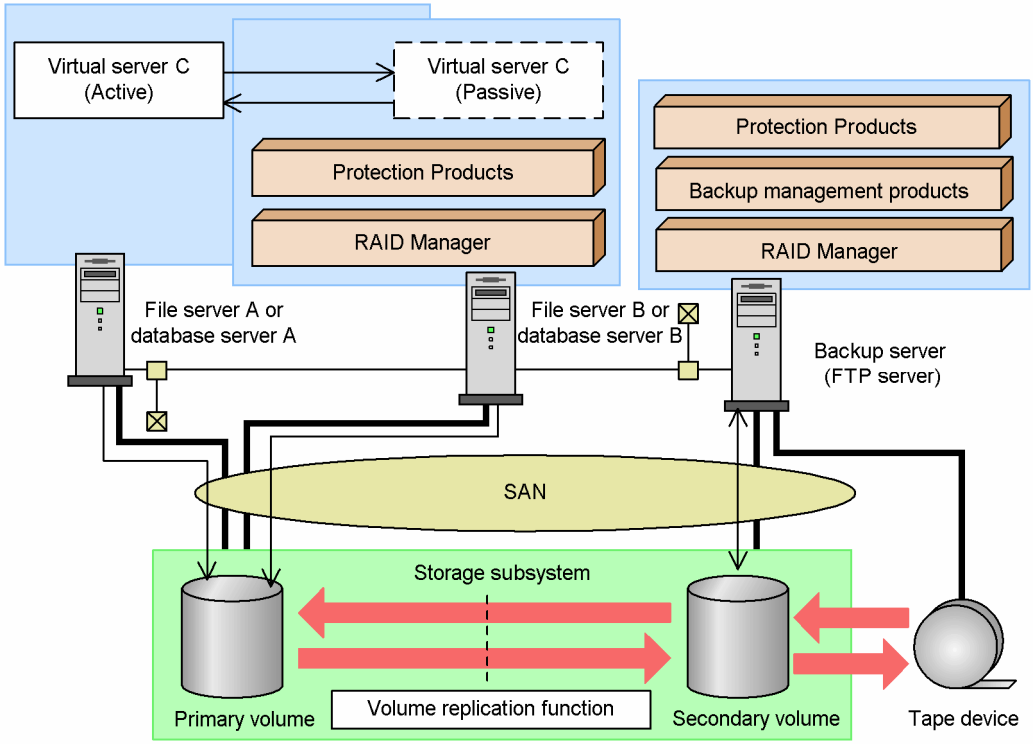


Figure 2.3 Operation Standby Cluster Configuration

To execute operation standby (active-passive) Protection Manager commands in a cluster configuration, first assign a storage destination for the dictionary map file used by each virtual server. For more information on assigning the dictionary map file in a cluster configuration, see section 4.6.2.

2.4 Executing the Mutual Standby Cluster Configuration

Protection Manager programs can handle a cluster configuration of the mutual standby type (active-active) on a database server and file server. A cluster configuration of the mutual standby type is a multiple structure of operation standby type environments.

Figure 2.4 shows an example of a mutual standby (active-active) cluster configuration. You can build a cluster configuration using file servers or database servers. You cannot allocate one cluster server to a file server or database server and the other cluster server to a backup server.

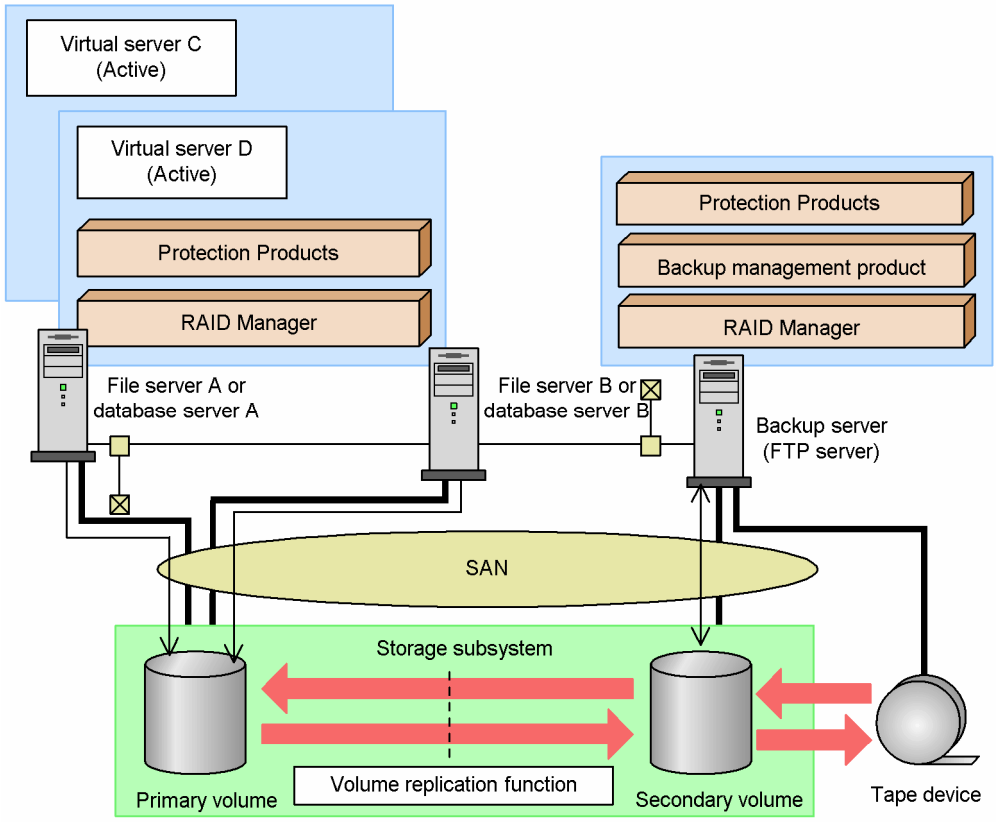


Figure 2.4 Sample Cluster Configuration of Mutual Standby Type (Active-Active)

To execute Protection Manager commands in a mutual standby (active-active) cluster configuration, create a dictionary map file for each virtual server. Additionally, you must assign a storage destination for the dictionary map file used by each virtual server. For more information on assigning the dictionary map file in a cluster configuration, see section 4.6.3.

2.5 Configurations for Backup and Restoration Between Subsystems (Remote Sites)

Protection Manager supports backup and restoration between storage subsystems (remote sites) using TrueCopy and UR to guard against storage subsystem failures on a local site.

This section explains configurations for data back up and restoration between storage subsystems (remote sites).

Protection Manager supports the following configurations for backing up and restoring data between storage subsystems:

- TrueCopy or UR configuration
- ShadowImage and TrueCopy multi-target configuration
- ShadowImage and UR multi-target configuration
- TrueCopy and UR multi-target configuration
- ShadowImage (multi-generation), TrueCopy, or UR configuration

The following explains each of these configurations.

2.5.1 TrueCopy or UR Configuration

Protection Manager supports the TrueCopy or UR configuration as the basic configuration for backing up and restoring data between storage subsystems. Figure 2.5 shows a TrueCopy or UR system configuration diagram.

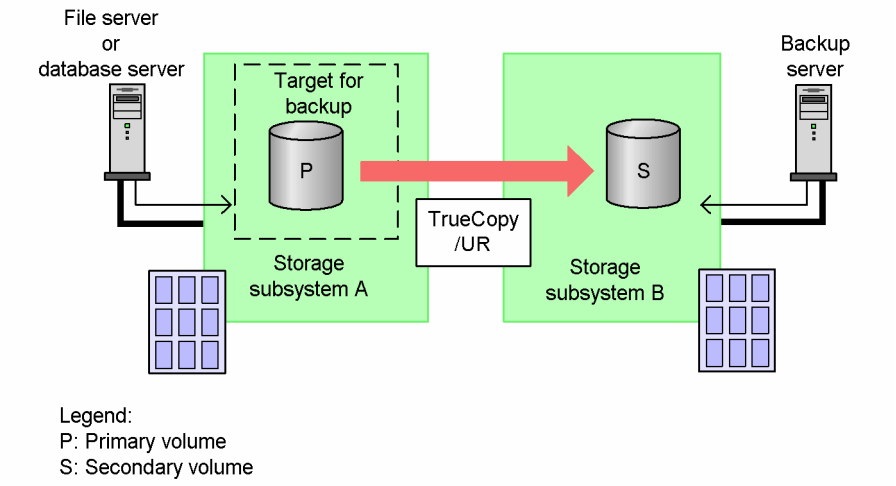


Figure 2.5 TrueCopy or UR System Configuration

2.5.2 ShadowImage and TrueCopy Multi-Target Configuration

Protection Manager supports a multi-target configuration for ShadowImage and TrueCopy. Figure 2.6 shows the ShadowImage and TrueCopy multi-target configuration diagram.

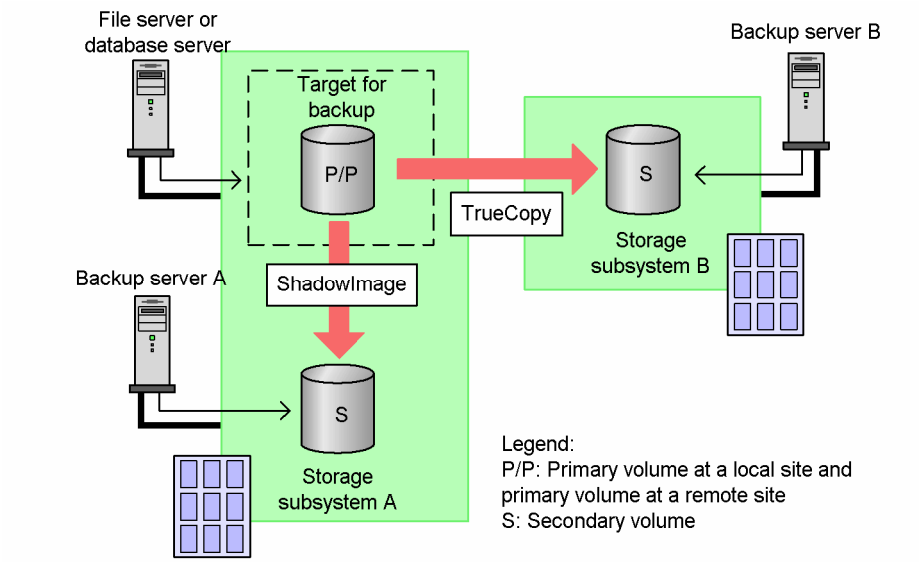


Figure 2.6 ShadowImage and TrueCopy Multi-Target Configuration

2.5.3 ShadowImage and UR Multi-Target Configuration

Protection Manager supports a multi-target configuration for ShadowImage and UR. Figure 2.7 shows the ShadowImage and UR multi-target configuration diagram.

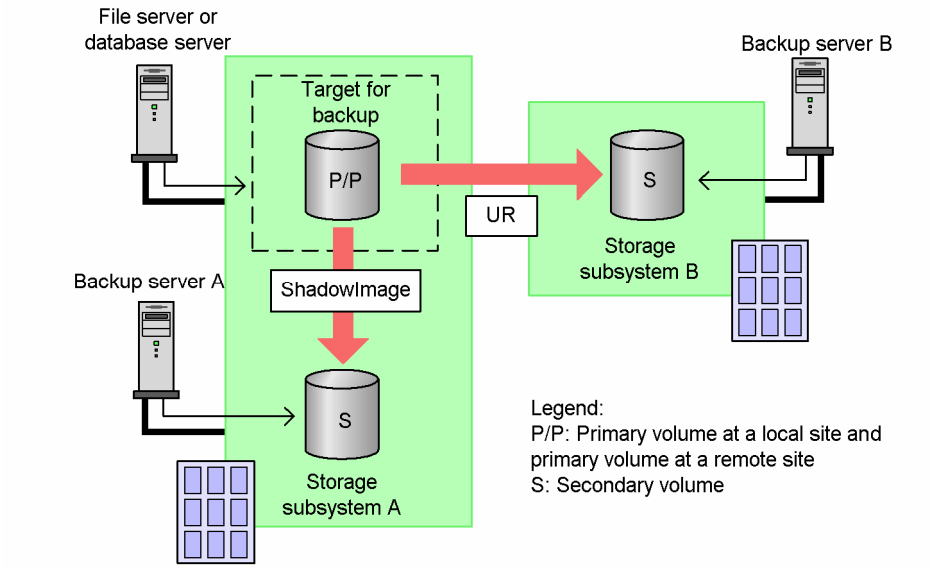


Figure 2.7 ShadowImage and UR Multi-Target Configuration

2.5.4 TrueCopy and UR Multi-Target Configuration

Protection Manager supports a multi-target configuration for TrueCopy and UR.

Figure 2.8 shows the TrueCopy and UR multi-target configuration diagram.

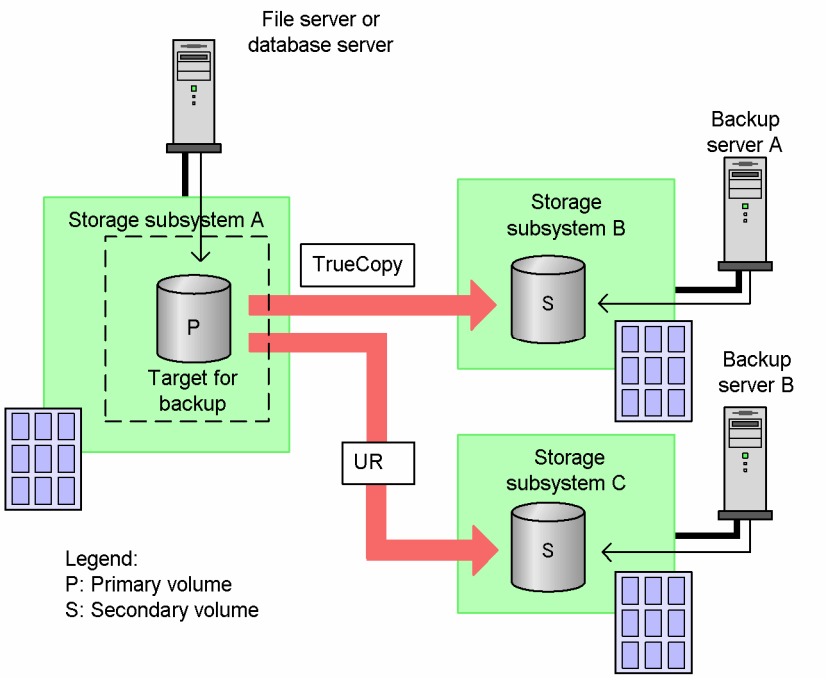


Figure 2.8 TrueCopy and UR Multi-Target Configuration

2.5.5 ShadowImage (Multi-Generation), TrueCopy, or UR Configuration

Protection Manager supports a system configuration for ShadowImage (multi-generation) and TrueCopy or UR.

Figure 2.9 shows the ShadowImage (multi-generation), TrueCopy, and UR configuration diagram.

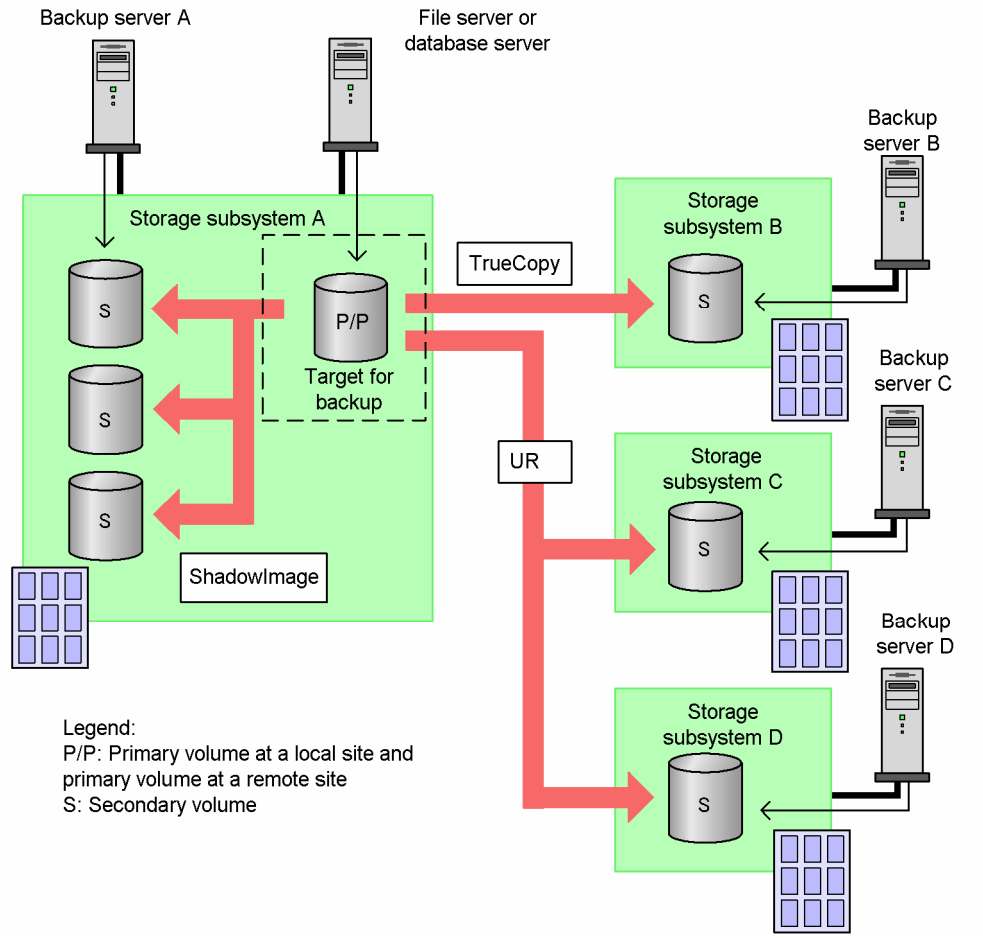


Figure 2.9 ShadowImage (Multi-Generation), TrueCopy, or UR Configuration Diagram

2.6 Precautions for Data Backup and Restoration Between Storage Subsystems

This section explains precautions for data backup and restoration between storage subsystems (remote sites).

Regarding pair creation of paired volumes between storage subsystems, Protection Manager does not automatically create a pair of paired volumes when it backs up data between remote sites. Users must create a pair of paired volumes in advance.

Regarding data backup from one primary volume to multiple volumes between storage subsystems, Protection Manager supports only data backup from one primary volume to one secondary volume. It cannot back up data from one primary volume to multiple secondary volumes.

Regarding data backup from multiple primary volumes to multiple secondary volumes between storage subsystems, concurrent backup from multiple primary volumes to multiple secondary volumes requires that the secondary volumes have the same generation number. To assign the same generation number to multiple secondary volumes, specify the same number in `MU#`s defined in the RAID Manager configuration definition file (`horcmn.conf`). For the system configuration, check the hardware specifications.

Figure 2.10 shows a configuration diagram for concurrent backup from multiple primary volumes to multiple secondary volumes.

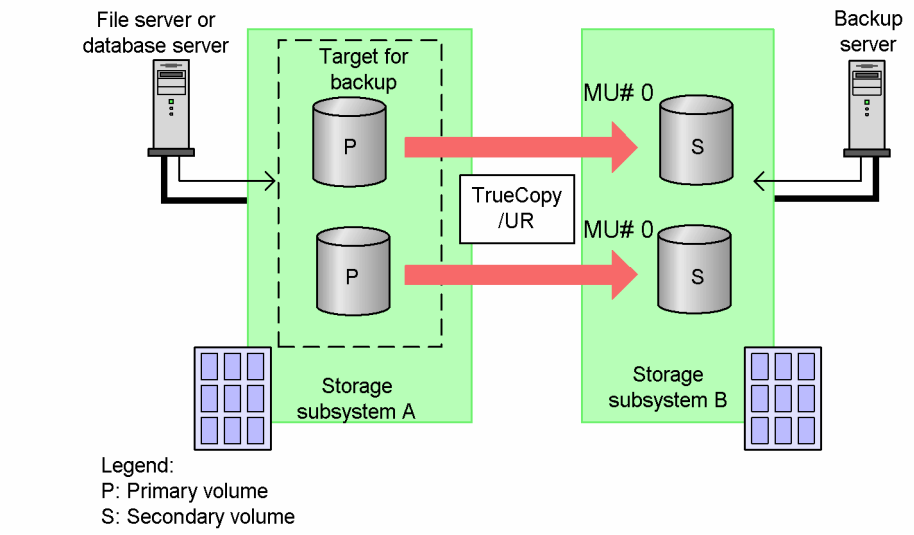


Figure 2.10 Sample Configuration for Concurrent Backup (Remote Copy)

RAID Manager instances between a local site and a remote site in a storage subsystem are not automatically started. Start the RAID Manager instances on the remote site in advance.

2.7 Restoring Data on a Host Not Used for SQL Server Database Backup

When the backup is of a SQL Server database, data on a host that differs from the host used for backup can be restored. To configure the system to restore data on a host for a SQL Server database other than the one used for initial backup, the system requires two servers: *database server 1* for backup, and *database server 2* for restoration. Operations are usually performed on database server 1, but when an error occurs on database server 1, the operations are switched to database server 2.

In this configuration, install the Protection Manager programs both in database server 1 and in database server 2. Execute backup commands on database server 1, and restoration commands on database server 2, respectively.

Figure 2.11 shows a system configured to restore data on a host other than the host used for backup.

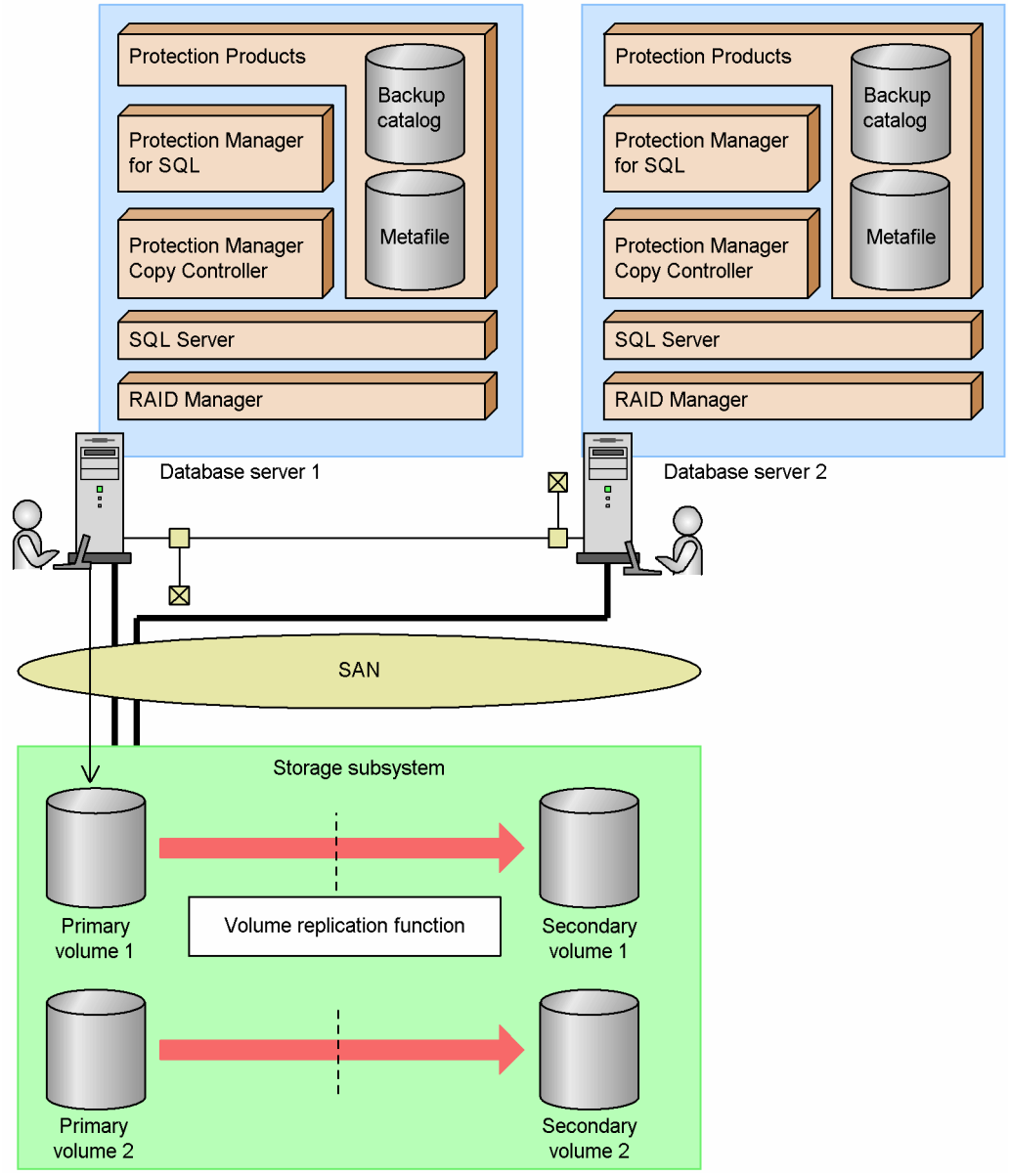


Figure 2.11 Configuration to Restore Data on a Host Not Used for Backup

2.8 Configuring the System to Use the SQL Server Log-Shipping Function

When the backup is of a SQL Server database, you can use the log-shipping function of Microsoft SQL Server 2000 Enterprise Edition and Microsoft SQL Server 2005. When you use this function, the system needs two servers: a source server that distributes log files, and a destination server to which the log files are distributed.

When using this configuration, install the Protection Manager programs both in the source server and in the destination server.

Figure 2.12 shows an example of a system configured to use SQL Server's log-shipping functionality.

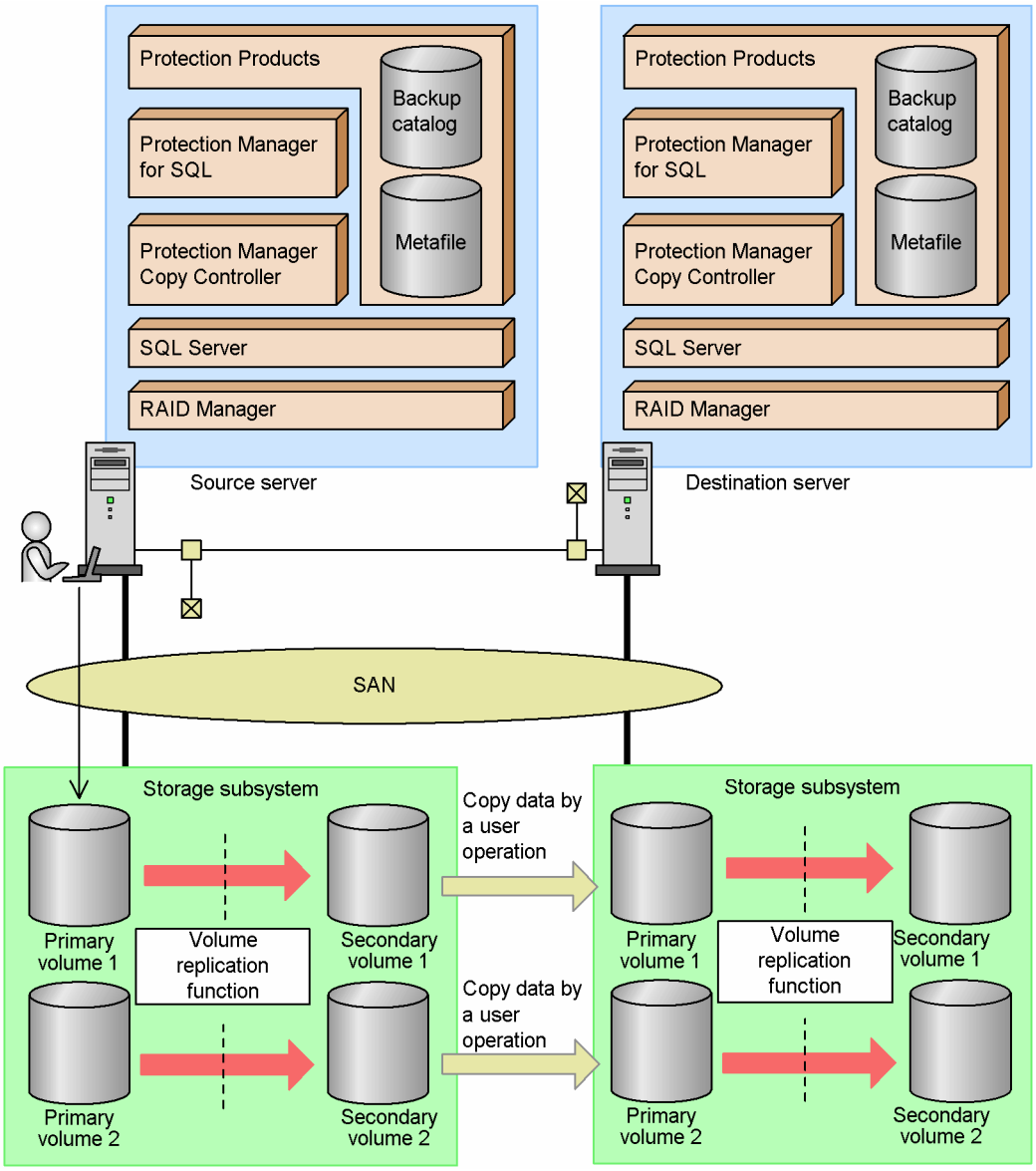


Figure 2.12 Configuration for the Log-Shipping Functionality

2.9 Cascade Configuration for a SQL Server Database Backup

When backing up a SQL Server database, Protection Manager programs can handle the backup to a secondary volume at a remote destination. The backup configuration in this case is called a cascade configuration because it has multiple hierarchies. Use user scripts for data backup and restoration in a cascade configuration. However, in a cascade configuration consisting of ShadowImage volumes only, user scripts cannot be used.

Figure 2.13 shows an example of a cascade configuration consisting of ShadowImage volumes and TrueCopy volumes for the backup of a SQL Server database.

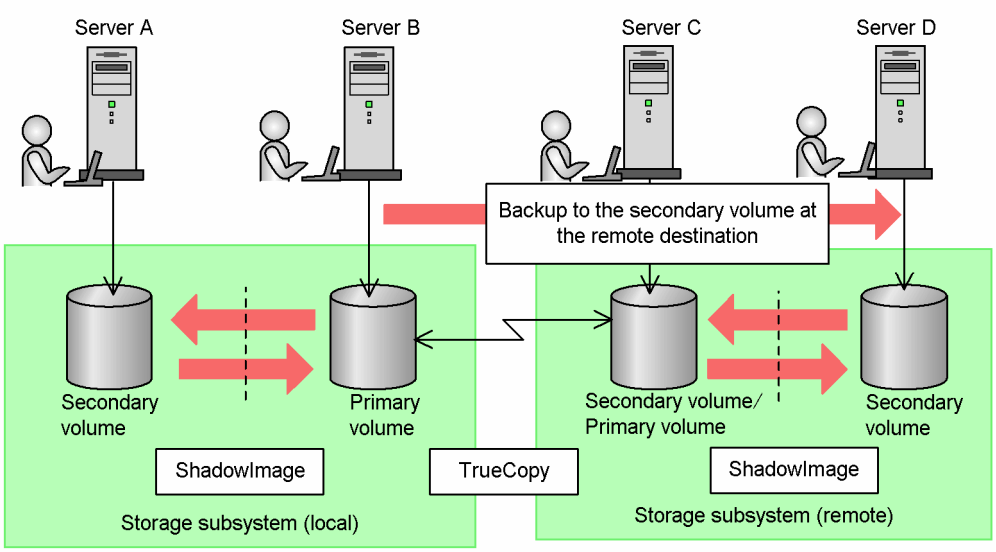


Figure 2.13 Cascade Configuration

2.10 Using a Multi-Target Configuration for SQL Server Database Backup

When backing up a SQL Server database, Protection Manager programs can handle a multi-target configuration backing up data to a remote destination and to a secondary volume with a local environment at the same time. Use user scripts for data backup and restoration in multi-target configuration.

An example of a multi-target configuration is shown in Figure 2.14.

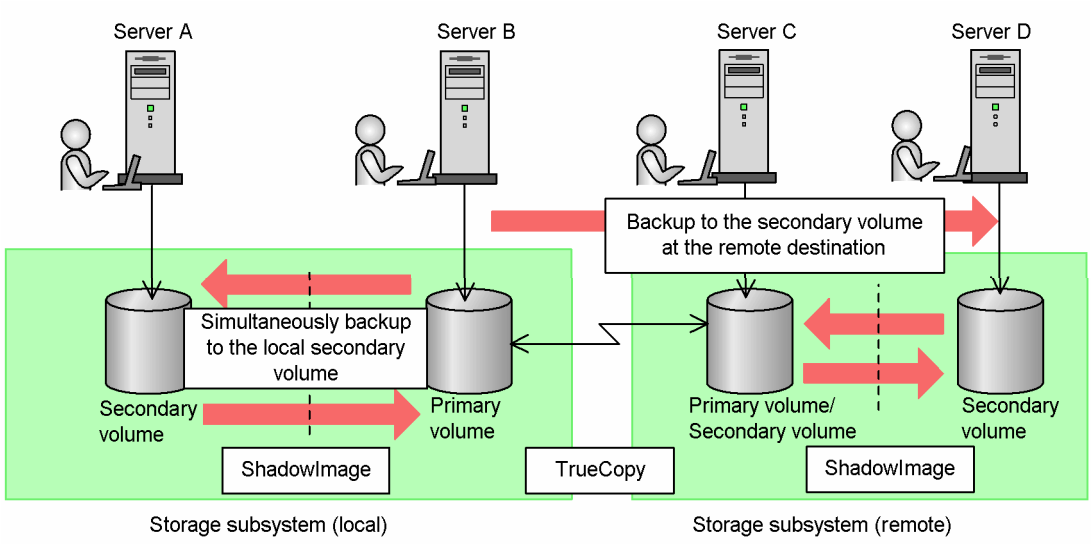


Figure 2.14 Multi-Target Configuration for SQL Server Database Backup

2.11 Requirements When Using RAID Manager to Configure a Pair Volume

Protection Manager programs use the pair volumes defined by RAID Manager to back up or restore file systems or database objects. When you create a pair volume by using RAID Manager, abide by the following rules:

- You cannot configure a paired volume so that the same server manages both the primary volume and secondary volume as physical disks.

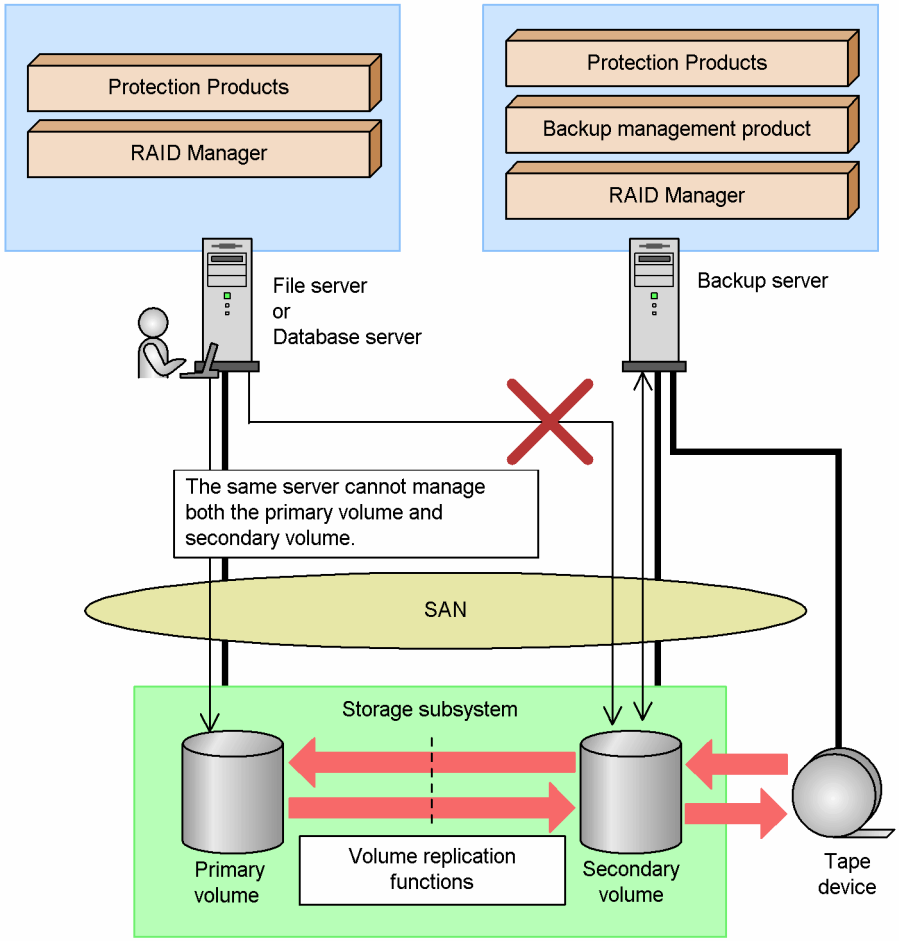


Figure 2.15 Managing the P-VOL and S-VOL from the Same Server (Without Protection Manager)

- A copy group must consist of pair volumes for the same application. When a copy group consists of more than one pair volume and there are pair volumes for different applications in the copy group, an unexpected backup or restoration may be performed.
- When using ShadowImage, configure the pair volume in the storage subsystem. When using TrueCopy, configure the pair volume between the storage subsystems.
- When the primary volume and the secondary volume are in the pair status, do not restart the server. If a mount operation is performed while the server is restarting, the primary volume and secondary volume might not be mounted correctly and a data inconsistency might occur.

- If there are multiple generations of backups (secondary volume), all the volumes might be mounted when the backup server restarts. Before executing the Protection Manager command, use the disk management functionality of the operating system to unmount the volumes that are not to be used for backup or restoration. Protection Manager programs operate the pair volumes for ShadowImage, QuickShadow, TrueCopy, or UR.
- Protection Manager programs cannot back up or restore pair volumes for Cross-System Copy or Hitachi HiCopy.
- Protection Products programs cannot back up or restore volumes paired by Cross-System Copy, Hitachi HiCopy, Hitachi CruiseControl or Volume Migration.
- Depending on the version of RAID Manager, the model of the storage subsystem, and the version of the microcode, the functionality for acquiring the copy type of the pair volume (copy group) might not be supported. Because of this, Protection Manager programs might not be able to correctly recognize whether the copy type of a pair volume (copy group) is supported, since the copy type cannot be retrieved. Thus, if you configure a system in which Cross-System Copy or Hitachi HiCopy pair volumes and ShadowImage or QuickShadow pair volumes co-exist, because the Cross-System Copy and Hitachi HiCopy pair volumes (copy groups) are not supported, make sure that they are disregarded as backup or restoration targets, by locking them with the `drmcgctl` command.

2.11.1 Considerations When Using TC or UR

- When using TrueCopy, do not enter an MU number in the MU# setting in the RAID Manager configuration definition file (`horcm`*`n`*`.conf`). If `0` is defined for the MU# setting, volume information for the TrueCopy pair will not be stored in the dictionary map when you execute a `drm`*`XX`*`display`* command with the `-refresh` option specified. Therefore, the volume information cannot be displayed by the Protection Products command, and the pair volume cannot be backed up.

  `drm``XX`*`display` indicates the `drmfsdisplay`, `drmsqldisplay`, or `drmexgdisplay` command.
- When using UR, define `h0` (`h` and a generation number) for the MU# setting in the RAID Manager configuration definition file (`horcm`*`n`*`.conf`). If `0` is defined for MU#, the UR information will not be stored in the dictionary map when you execute a `drm`*`XX`*`display` command with the `-refresh` option specified. Therefore, volume information cannot be displayed by the Protection Products command, and the pair volume cannot be backed up.
- When using TrueCopy or UR, make sure that you start the RAID Manager instance that manages the secondary volume (the RAID Manager instance that manages the primary volume when you are executing the command in a backup server) in advance.
- When using TrueCopy or UR, make sure that you create a pair before executing a Protection Manager command. If you do not, the pair volume cannot be backed up.
- When performing a backup or restoration on a database server, make sure that you unmount the secondary volume in advance. If you do not, unexpected backup or restoration might be performed.
- When you want to mix the pair volumes for ShadowImage and TrueCopy, the system configurations shown in Figure 2.16 can be used.

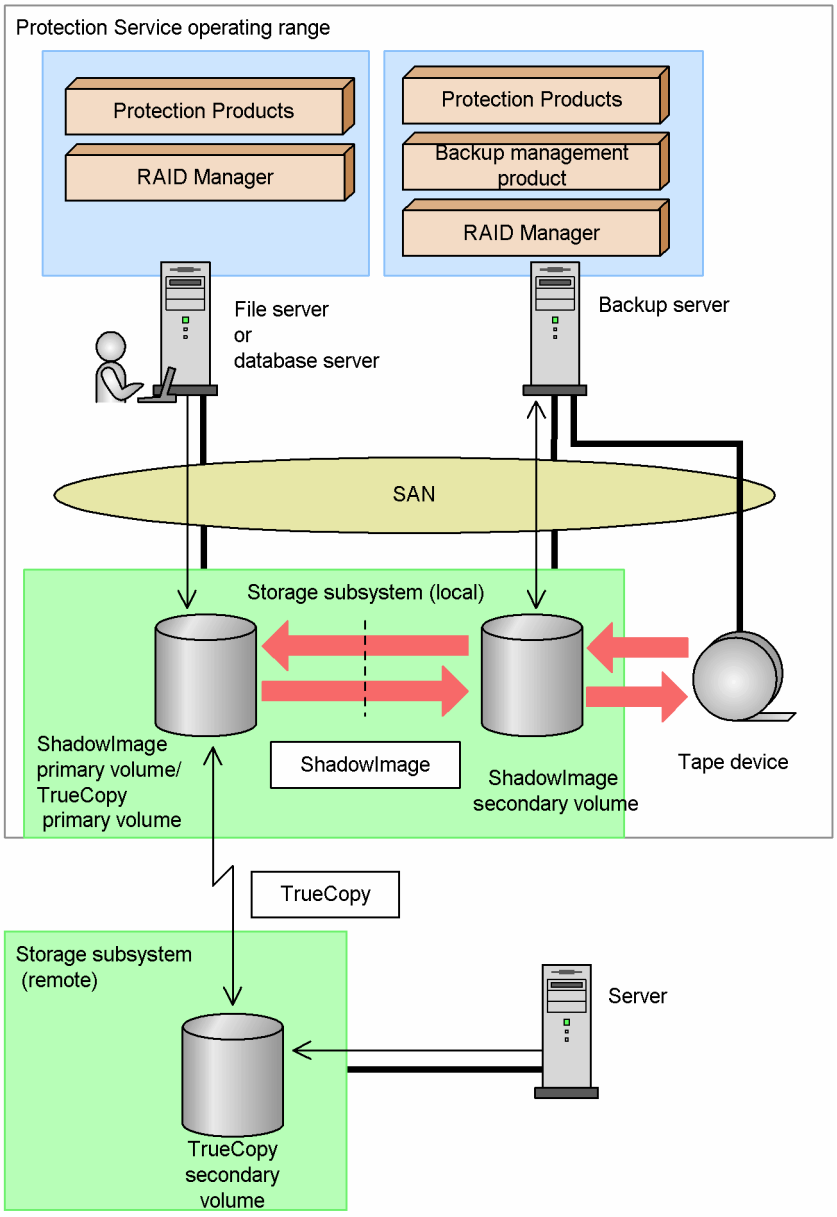


Figure 2.16 When the Primary Volume for SI and the Primary Volume for TC Are the Same LDEV

When using the configuration shown in Figure 2.16 to restore the paired volume for ShadowImage using Protection Manager, set the status of the paired volume for TrueCopy to `SMPL` or `PSUS` (`SSUS`). When you restore the pair volume for ShadowImage with the status of a pair volume for TrueCopy `PAIR`, an error message indicating that the status of the copy group is invalid is output and the restore operation fails.

When using the configuration shown in Figure 2.17 to back up or restore the paired volume for ShadowImage using Protection Products, set the status of the paired volume for TrueCopy to SMPL or PSUS (SSUS). When you back up or restore a pair volume for ShadowImage with the status of a pair volume for TrueCopy PAIR, an error message indicating that the status of the copy group is invalid is output and the backup or restore operation fails.

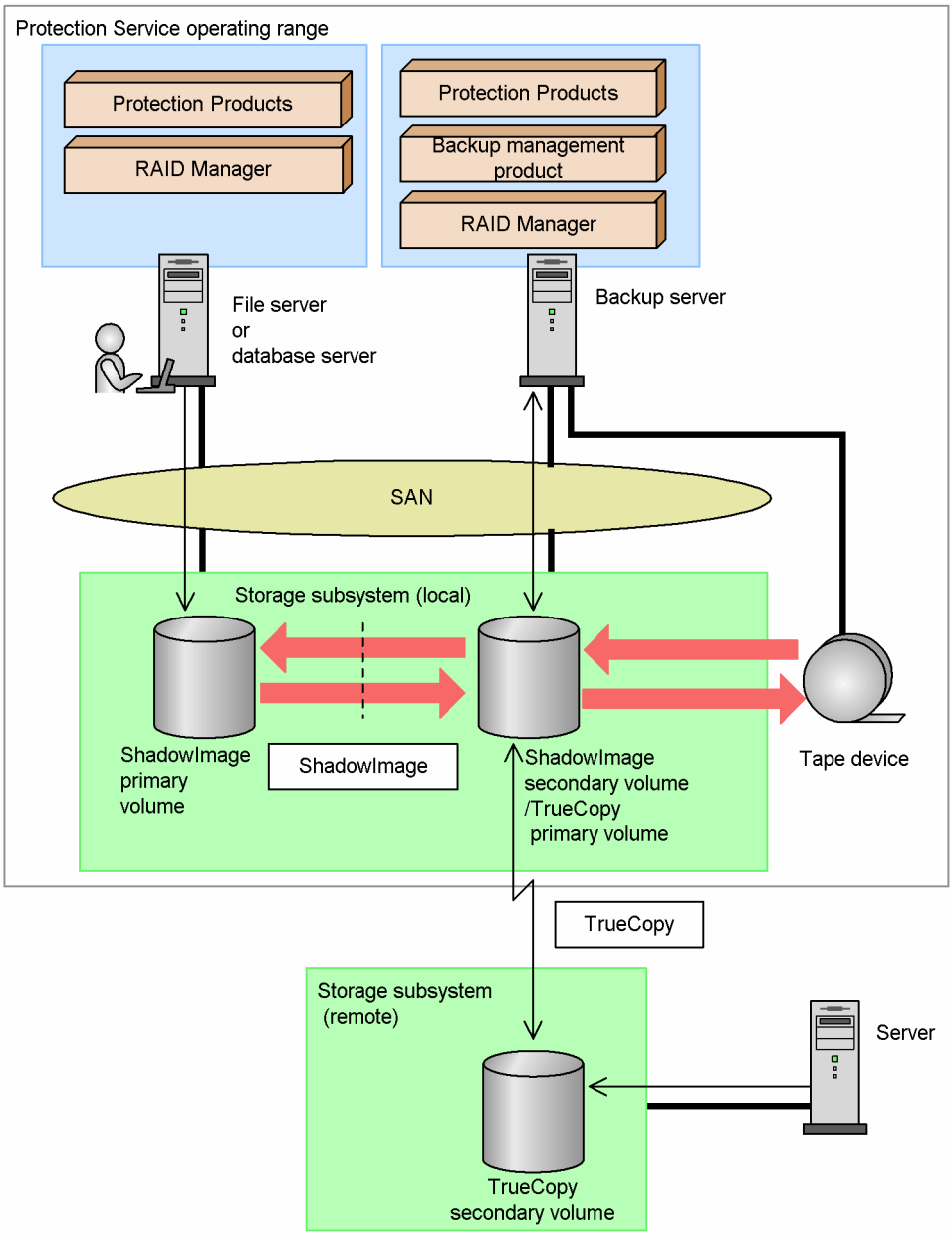


Figure 2.17 When the Secondary Volume for SI and the Primary Volume for TC Are the Same LDEV

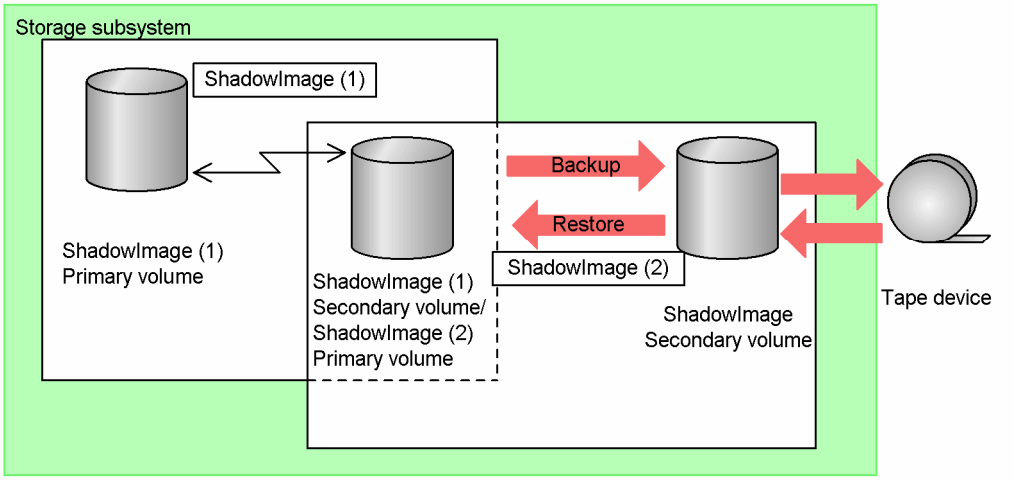


Figure 2.18 ShadowImage Cascade Configuration (Volume to Back up is Both Primary and Secondary)

In a ShadowImage cascade configuration (Figure 2.18), Protection Products programs only support backup and restore operations for the pair volumes of the primary volumes used in the database or the file system.

When using a ShadowImage cascade configuration where the volume subject to backup serves both as the primary volume and secondary volume, set the paired volume status on the secondary volume to `SMPL` and `SSUS`. Similarly, for restoration, set the status to `SMPL`.

- With TrueCopy-Async and UR, pair operations are always performed in units of at least one consistency group. Define the copy groups, and then start the backup and restore operations, considering the following points:
  - Define consistency groups while considering the operations for restoring the databases. Pay extra attention to the definitions if partial restoration of databases backed up at the same time will be performed.

  For SQL Server databases

  Define copy groups to form a single consistency group for each database. However, the databases that are always restored together can be in the same consistency group.

  For Exchange databases

  Define copy groups to form a single consistency group for the data files (`*.edb`, `*.stm`) in each storage group, or for the transaction log files and checkpoint files (`*.log`, `*.chk`) in each storage group.

  However, if multiple storage groups are always restored together, the data files (*.edb, *.stm) or transaction log files and checkpoint files (*.log, *.chk) in the storage groups can be in the same consistency group.

  Also, if a roll forward is not performed during restoration or recovery, data files or transaction log files and checkpoint files can be in the same consistency group.
  - Define each consistency group as a separate RAID Manager group, so that there will be a match with copy groups, and so that the number of groups is not too low or high.

  Figure 2.19 and Figure 2.20 show the relationships between databases, volumes, consistency groups, and RAID Manager groups.

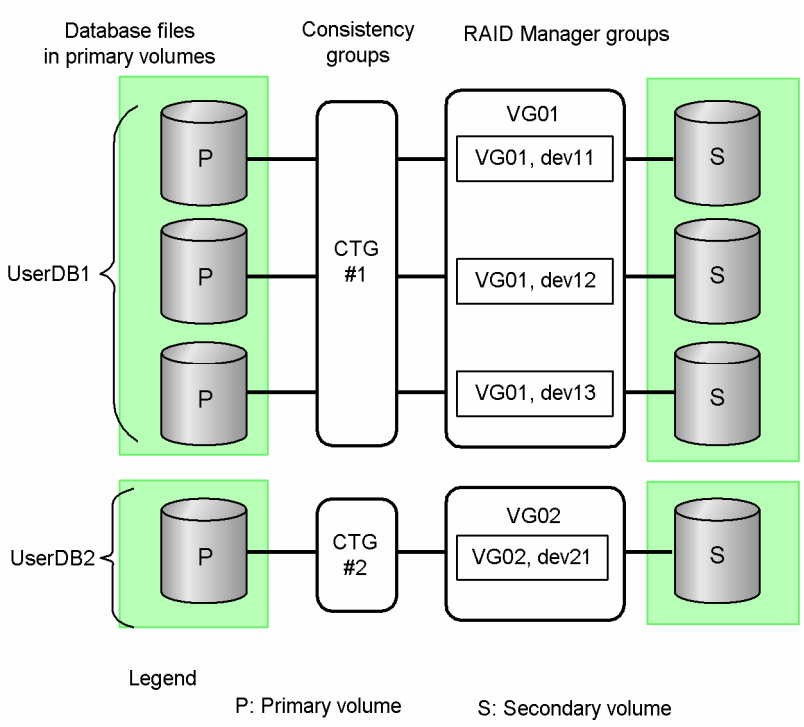


Figure 2.19 TrueCopy-Async/UR Configuration: Restoring UserDB1 and UserDB2 Together or Separately

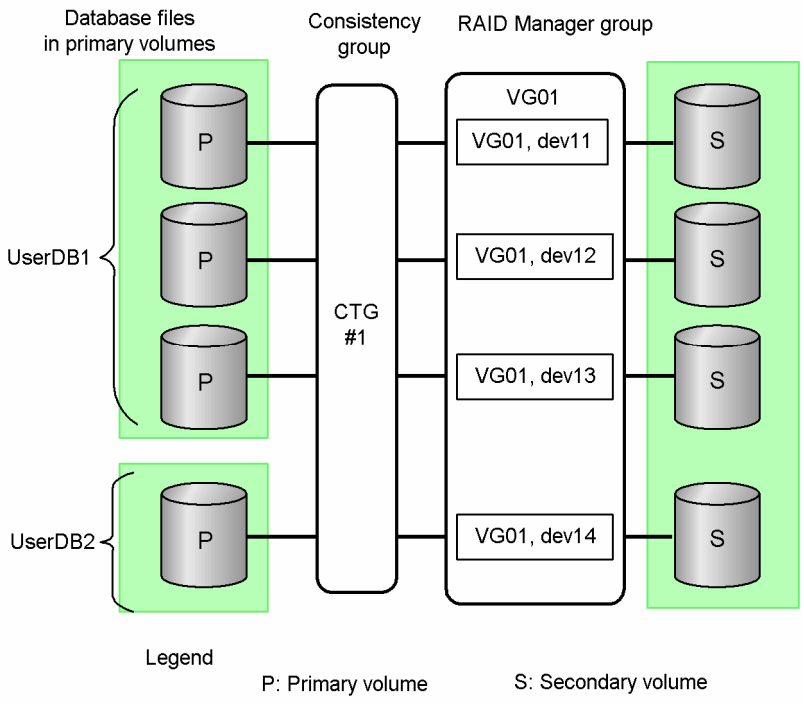


Figure 2.20 TrueCopy-Async/UR Configuration: Restoring UserDB1 and UserDB2 Together

- For information about precautions to take when changing a volume configuration, see section 2.12.3.

- Even if there are pair volumes created in the storage subsystem, you cannot use them unless they have RAID Manager configurations defined on the host.

2.11.2 Requirements for Exchange Database Backup

Protection Manager for Exchange supports the following Exchange database products:

- Exchange 2000 Enterprise Server
- Exchange 2000 Server
- Exchange Server 2003 Enterprise Edition
- Exchange Server 2003 Standard Edition

Table 2.1 lists the types of data subject to backup in Protection Manager for Exchange.

Table 2.1 Types of Data Subject to Backup in Protection Manager for Exchange

Database subject to backup	Files subject to backup	Extension
Exchange Server storage group	Data files(Exchange Server Information Store)	*.edb *.stm
	Transaction log files	*.log
	Checkpoint files	*.chk

Protection Manager for Exchange is backed up and restored in units of physical volumes, so the configuration of objects of an Exchange database has the following conditions:

- The files to be backed up must all be placed on a RAID volume defined as a pair.
- Deploy the storage group taking into consideration that backup is conducted in units of physical volumes. If you deploy multiple storage groups in the same physical volume, you need to back up or restore these storage groups as one group. When backing up or restoring a storage group separately, deploy the storage group on a separate physical volume.
- When you place an Exchange database that is subject to backup on the volume, note the following:
  - You cannot put the transaction log file (`*.log`) on a volume where a data file is placed.
  - For online backup and VSS backup, data files (`*.edb, *.stm`) and checkpoint files (`*.chk`) cannot be allocated to the same volume.
- Do not use the following characters when setting a name for a storage group: `= ; \ /`

- Because the following operations change information pertaining to the Exchange Server on the Active Directory, if you perform any of these operations you will no longer be able to restore backup data obtained before these operations (that is, you will no longer be able to use the `drmexgrestore` or `EX_DRM_EXG_RESTORE` command):
  - Modification of the paths of the Information Store, log, and system files
  - Addition or deletion of storage groups
  - Modification of the storage group name
  - Addition of an Information Store to a storage group
  - Deletion of an Information Store from a storage group

  If you have performed any of the above operations, you will need to back up the active directory and the Exchange database again. To perform a backup for the active directory, follow the instructions given in documentation provided by Microsoft.
- Performing any of the following operations changes the database name, and you will no longer be able to restore and recover backup data obtained before these operations (that is, you will no longer be able to use the `drmexgrestore` or `EX_DRM_EXG_RESTORE` command with the `-recovery` option specified):
  - Repairing Information Store using the `ESEUTIL` utility (`ESEUTIL /p`)
  - Defragmentation using the `ESEUTIL` utility (`ESEUTIL /d`)

  If you have performed any of these operations, you will have to execute the `drmexgbackup` or `EX_DRM_EXG_BACKUP` command and back up the Exchange database again.
- If you use a circular log, the oldest log among the existing log files is deleted when a new log file is created. This means that some logs may disappear at roll-forward. For this reason, when performing roll-forward recovery, do not use a circular log.
- The restoration storage group for Exchange Server 2003 cannot be backed up. The files and folders that make up the restoration storage group must be placed on a different file system from the storage groups to be backed up.
- When using QuickShadow, you cannot back up to tape by using Protection Manager for Exchange. Instead, use another backup product to back up (to tape) an Exchange database on the primary volume.

2.12 Volume Configuration for Backing up or Restoring Data

Table 2.2 shows the volume-configuration units used by Protection Manager when backing up and restoring data.

When using VSS to back up data, you cannot back up file systems on a dynamic disk.

Table 2.2 Volume Configuration When Backing up and Restoring Data

Logical Volume Manager	Disk Management Method	Volume-configuration units used by Protection Manager when backing up and restoring data
LDM	Basic disk	Physical disk
	Dynamic disk[1]	Disk set
VERITAS Volume Manager for Windows[2]	Basic disk	Physical disk
	Dynamic disk	Disk group

[1]Protection Products do not work with a Windows 2000 Standard LDM dynamic disk.

[2]VSS cannot be used.

When using a Windows dynamic disk, you can use the logical volume manager to group one or more physical volumes and use the group as one or more logical volumes. Specifying a logical volume enables users to back up and restore data without consideration for physical volume. The unit for grouping physical volumes and logical volumes is called a disk group.

The minimum unit for data backup and restoration depends on whether the LDM environment or VERITAS Volume Manager for Windows environments is used.

In Protection Manager dynamic disk configuration, a group of one or more dynamic disks and one or more logical volume groups allocated to a dynamic disk is called a *disk set*. In Figure 2.21, part of the physical volume A is allocated to the logical volume A, and the rest of the physical volume A and part of the physical volume B are allocated to the logical volume B. Likewise, the rest of the physical volume B and part of the physical volume C are allocated to the logical volume C, and the rest of the physical volume C is allocated to the logical volume D. In this case, a group of the four logical volumes to which the contents of the three physical volumes are allocated is one disk set (disk set 1). Also, if you allocate all the contents of one physical volume to one logical volume, you can make a one-to-one relationship disk set (disk set 2).

In the VERITAS Volume Manager for Windows environment, when configured, a disk group is recognized as one or more disk sets. In this environment, Protection Manager backs up and restores data using a disk group as the minimum unit.

The following figure shows examples of a disk set configuration and of a relationship between a disk group and a disk set.

Figure 2.21 shows the relationship between a disk group and a disk set.

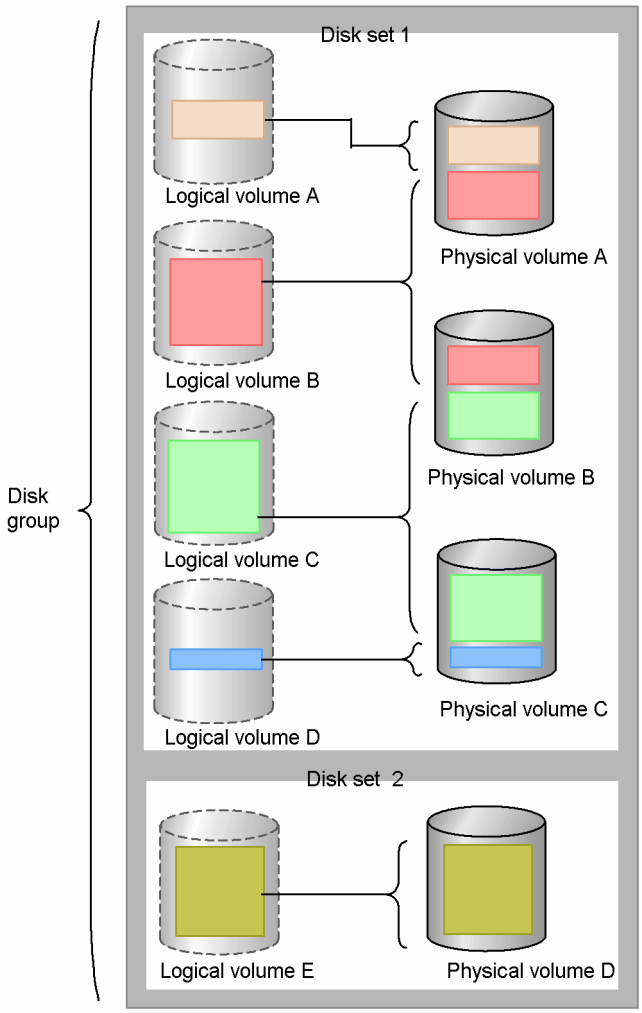


Figure 2.21 Examples of Disk Set Configuration and Relationship Between a Disk Group and a Disk Set

In this manual, unless otherwise noted, LDM disk sets and VxVM disk groups are generically represented as a disk group. Protection Manager can also simultaneously back up a basic disk and a disk group.

When a Windows dynamic disk is being used, Protection Products support both of the following configurations:

- One disk group is used as one logical volume.
- One disk group is divided into multiple logical volumes.

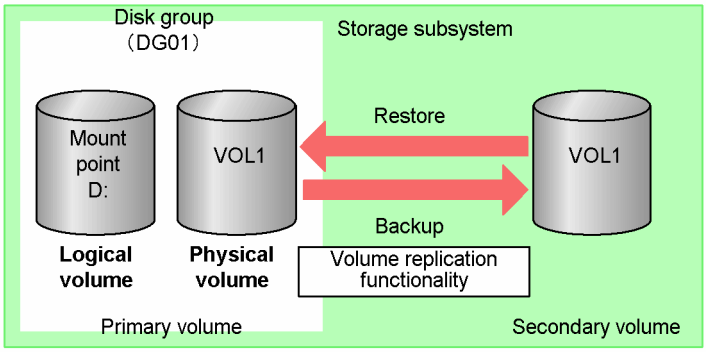
When a Windows basic disk is being used, Protection Products only support a configuration where one physical disk is used as one logical volume (partition).

2.12.1 Examples of Disk Group Configurations Usable by Protection Manager

If you configure a disk group with logical volume manager, Protection Manager can be used only when one or more physical volumes constitute one disk group. You cannot divide one physical group into multiple disk groups.

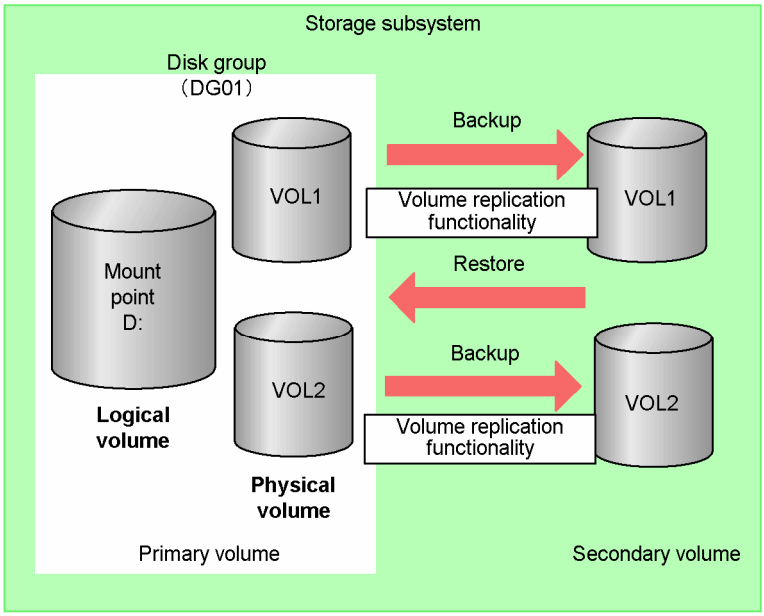
Figure 2.22 shows examples of disk group configurations usable by Protection Manager.

Configuration of one disk group to one physical volume



This figure shows one physical volume (VOL1) comprising one disk group (DG01).

Configuration of one disk group to multiple physical volumes



This figure shows multiple physical volumes (VOL1,VOL2) comprising one disk group (DG01). In this case, the maximum value of the logical volume capacity is the total capacity of VOL1 and VOL2.

Figure 2.22 Examples of Disk Group Configurations for Protection Manager

In the Windows dynamic disk configuration, when multiple disk groups exist in a file system, Protection Manager collectively backs up and restores all the mount points in disk groups. It cannot back up only some drives in a disk group.

Figure 2.23 shows data backup and restoration in a system configuration where multiple disk groups exist in a file system.

Mount point	Logical volume	Disk group	Physical disk	Copy group
D:	DG01\Volume1	DG01	VOL1	Grp1,dev 1
E:	DG01\Volume2		VOL2	Grp1,dev 2
F:	DG01\Volume3			
G:	DG02\Volume1	DG02	VOL3	Grp2,dev 3
			VOL4	Grp2,dev 4

Figure 2.23 Data Backup and Restoration in a System Configuration With Multiple Disk Groups

In this figure, disk groups `DG01` and `DG02`, each of which consists of two physical volumes, are configured. Three logical volumes are created on `DG01` and one logical volume is created on `DG02`. Mount points `D:`, `E:`, and `F:`, are assigned to `DG01` and mount point `G:` assigned to `DG02`.

In Figure 2.24, specifying a mount point directory batch definition file in the `drmfsbackup` command enables Protection Manager to back up data. In this mount point directory batch definition file, the mount points of all the file systems defined in the disk groups are written.

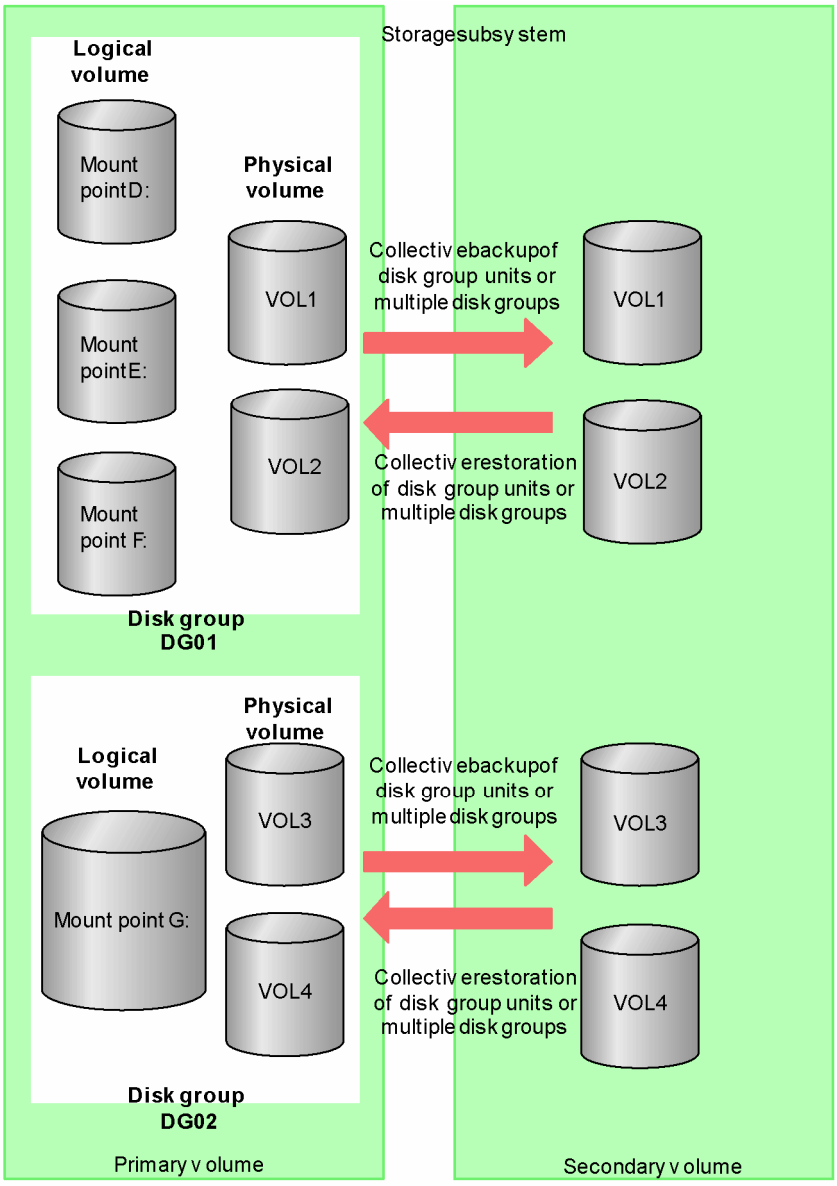


Figure 2.24 Multiple Disk Group Configuration

In this example, the command uses the mount point directory batch definition file mp_dg01.dat to specify the mount points D:, E:, and F: of DG01, and the mount point G:of DG02:

```
PROMPT> drmfsbackup mp_dg01.dat -mode cold
KAVX0001-I The drmfsbackup command will now start.
KAVX0019-I Copying file(s) now...
KAVX0040-I The backup ID has been assigned as follows:
Backup ID = 0000000001
KAVX0002-I The drmfsbackup command will now end.
PROMPT>
```

Specifying DG02 file system (G:) following the -target option of the drmfsrestore command enables Protection Manager to restore only DG02. As shown in the following example, executing the drmfsrestore command without -target option restores both DG01 and DG02 file systems.

Restore only the file system created in disk group DG02 after backing up disk groups DG01 and DG02.

```
PROMPT> drmfsrestore 0000000001 -resync -target G:
KAVX0001-I The drmfsrestore command will now start.
KAVX0019-I Copying file(s) now...
KAVX0002-I The drmfsrestore command will now end.
PROMPT>
```

2.12.2 Precautions for Volume Configuration

Observe the following precautions when configuring Protection Manager:

- You cannot use Protection Manager in a configuration where a physical volume is divided into multiple disk groups using logical volume manager.
- Do not use any functions that let you freely expand or reduce the capacity of a logical volume. If you use such functions, the Protection Manager programs will not perform backup and restoration correctly. This is because the number of physical volumes that make up the logical volume changes according to the change in capacity of the logical volume. The Protection Products programs check whether the configuration of logical volumes has been changed and, if changed, restoration terminates with an error.
- If the volumes to be processed by Protection Manager programs contain an LDEV for which a mode other than normal is specified for the access level by using Data Retention Utility, neither a backup nor a restoration can be performed.
- When executing backup and restoration of file systems and databases, the Protection Manager programs mount and unmount the volume subject to processing. In this situation, set the mount point directory name as follows:
  - Maximum number of characters: The specifiable path length depends on the limitation for the RAID Manager mount or unmount functionality.

    When performing a disk backup, the path length limitation described above applies to the length of the path where the volume for which cold backup operations are to be performed has been mounted.

When backing up to tape, the path length limitation described above applies to the length of the path where the volume for which disk backup operations are to be performed has been mounted. This limitation also applies to the length of the path where the secondary volume for which tape backup operations are to be performed is mounted.

- Usable characters are those which can be used for folder names in Windows (you cannot use null characters).

- One mount point can be specified for each logical volume.
- Allocate database files and file systems that are not subject to backup and restoration so that they are not included in the same disk configuration unit as those that are subject to backup and restoration.
- Allocate database files and file systems that are backed up simultaneously but may be restored separately so that they are not included in the same disk configuration unit as those that are subject to backup and restoration.
- If one disk group contains multiple physical volumes, Protection Manager performs volume duplication operations for backup only by the number of physical volumes.
- When performing online backups or VSS backups, do not define the following as copy groups, and do not specify them as backup targets: directories above a directory mount point, or volumes mounted to a drive. Also, configure directory mount points that are backup targets so that they are on the same directory level.

  When performing a cold backup, acquire a backup by specifying as backup targets all the mount points under a directory that is a backup target.
- Precautions relating to mounting a directory
  - In Windows, do not define the following as copy groups, and do not specify them as backup targets: directories above a directory mount point, or volumes mounted to a drive. Also, configure directory mount points that are backup targets so that they are on the same directory level.

The following shows a configuration example of a mounted directory that becomes a backup target. As shown in Figure 2.25, data files and log files are mounted to a directory and specified as backup targets.

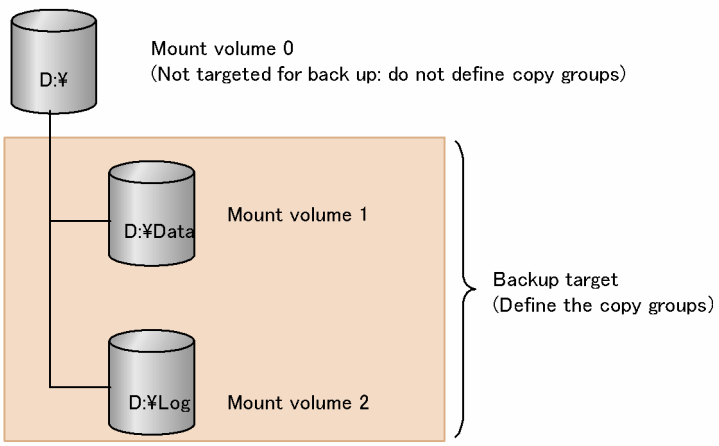


Figure 2.25 Configuration Example of a Mounted Directory that Becomes a Backup Target

Figure 2.26 shows a configuration example of a mounted directory that does not become a backup target. When the configuration places a mount point directory on a directory below a volume, higher-level volumes that contain a volume that was mounted to a directory cannot be specified as a backup target.

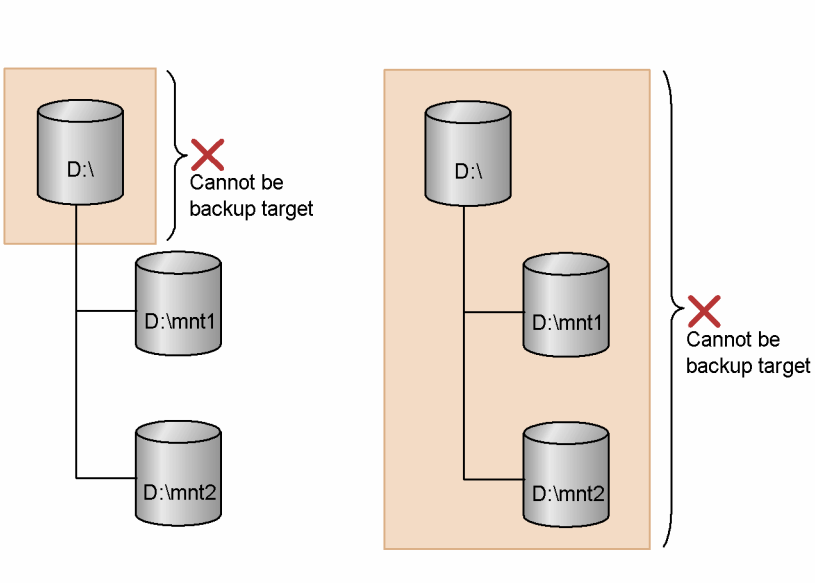


Figure 2.26 Configuration Example of a Mounted Directory that Does not Become a Backup Target

2.12.3 Precautions for Changing a Volume Configuration

If the following operations or changes are performed on a logical volume manager that is a backup operation target, or on a file system or RAID Manager configuration definition file (`horcm`*`n`*`.conf`), then update the dictionary map file before executing backup.

- A new file system was mounted on a directory under the file system.
- A file system was unmounted.
- A file system was moved to another directory and mounted.
- A disk group was deleted or disabled.
- A logical volume allocated to a disk group was added or deleted.
- A physical volume belonging to a disk group is added, deleted, or replaced.
- A change was made in the RAID Manager configuration definition file (`horcm`*`n`*`.conf`), such as the adding or deleting of a pair volume.

Note that if you execute backup without updating the dictionary map file, a system inconsistency might occur. To update the dictionary map file, use the `drmfsdisplay`, `drmsqldisplay`, or `drmexgdisplay` command. For details about how to use these commands, see the *HiCommand® Protection Manager Command Reference* manual.

If you made changes to a configuration involving an update of the dictionary map file, to restore the data backed up before the change to a file server or database server, first restore the configuration for the logical volume manager and file system to the state that was in place at the time the backup was created, and then perform restoration. Note that if you perform restoration incorrectly, a system inconsistency might occur. To review the configuration that was in place at the time the backup was created, use the `drmfscat`, `drmsqlcat`, or `drmexgcat` command. For details about how to use these commands, see the *HiCommand® Protection Manager Command Reference* manual. For details about restoration when a physical volume is replaced, see Appendix E. Replacing Primary and Secondary Volume Disks While Protection Manager Programs Are Running.

2.12.4 Requirements and Precautions for Configuring a Disk Group

When you operate Protection Products in either a disk set configuration managed by Windows Logical Disk Manager (LDM) or a disk group configuration of VERITAS Volume Manager for Windows, the following prerequisites apply:

- The same OS, version revision, and service pack are installed on all the Windows servers subject to backup and restoration by Protection Manager.
- The physical disk subject to backup and restoration does not include a system (OS startup) disk.
- All the physical disks subject to backup and restoration are LUs of a storage subsystem device and for which a pair volume is defined.
- The dynamic volume is a concatenated volume (simple volume or spanned volume).
  - We recommend that one disk group consist of one concatenated volume.
  - Striped volumes, mirrored volumes, and RAID5 volumes are not supported.
- Protection Manager Console can back up and restore file systems only in system configurations where file systems correspond one for one to disk groups.
- You must dynamically recognize the secondary volume when backing up secondary volumes (dynamic disk backup destinations) of multiple generations to tape.

2.12.4.1 Using Dynamic Disks in LDM

- If the database server and file server are dynamic disks, they cannot be configured in a cluster.
- Assign a unique volume label to each dynamic volume in an LDM environment. A volume label can be changed in the volume property dialog box. You can display the volume property dialog box by selecting **Disk Management** in the Computer Management window, right-clicking the target volume, and then clicking **Properties**.
- Note that the mount operation will fail if the same volume label is assigned to multiple volumes within the disk set.
- Enter a character string (up to 31 bytes) for the volume label.
- After a volume label has been changed, always update the dictionary map file.

2.12.4.2 Using Dynamic Disks in VERITAS Volume Manager for Windows

Prerequisites for the VERITAS Volume Manager environment and disk groups are as follow:

- VERITAS Volume Manager with the same version revision and service pack is installed on all the Windows servers subject to backup and restoration by Protection Manager.
- Do not back up and restore data with VERITAS Volume Manager functions because the disk group configurations managed by Protection Manager may be changed.
- Disk group names must be unique in all the systems that back up and restore data. If multiple file servers or database servers are sharing one backup server, the disk group names used by the file server and database server may be duplicated in the backup server. If this occurs, Protection Manager is unable to uniquely recognize the disk groups in which data to be backed up to tape is stored.
- Protection Manager cannot back up and restore primary disk groups.
- Installing VERITAS Volume Manager enables you to use VERITAS Enterprise Administrator instead of Windows disk management. Do not use VERITAS Enterprise Administrator during execution of Protection Manager.

2.12.4.3 Backing up and Restoring Dynamic Disks By Recognizing the Secondary Volume Dynamically

- The LDM environment requires that `diskpart.exe` be installed in the Windows system directory. Windows 2000 Server requires additional installation of Windows resource kits because they do not come standard in the `diskpart.exe` installation.
- In Windows 2000 Server, when Protection Manager dynamically recognizes the secondary volume, the Demount Device Warning dialog box appears.
- When Protection Manager dynamically recognizes the secondary volume, the Demount Device Warning message is output to the event log.

2.12.4.4 Backing up and Restoring Dynamic Disks By Recognizing the Secondary Volume Statically

- Do not rescan disk information with disk management, VERITAS Enterprise Administrator, or the `diskpart` or `vxassist` command. A **Foreign** disk, **Failed** volume, or **Missing** disk appears, making it impossible for Protection Manager to recognize the disk configuration correctly. If there is an application that is using the appropriate disk, an unexpected write error may occur.

  If Protection Manager is unable to recognize the disk configuration correctly, restore the disk configuration using the following procedures. Restoration of application data being executed when disk information is being rescanned is not guaranteed.

  a) Execute Import Foreign Disk Format for the Foreign disk with Disk Management.

  b) Execute Reactivate Volume for the Failed volume. This mounts the file system and restores the original configuration. We recommend that you execute CHKDSK (disk check) for the restored file system.

- If you change a disk configuration on a physical disk (e.g., by volume expansion), update the dictionary map files on the file server or database server, back up the files to a volume once as in initial construction, and then restart the backup server. This enables the server to recognize the secondary volume disk configuration correctly.

2.12.4.5 Installing VERITAS Volume Manager for Windows LDM Environment

Installing VERITAS Volume Manager in a server that operates in the LDM environment makes it possible for Protection Manager to recognize all the disk sets created by LDM as one disk group.

If you want to back up and restore data in units of disk sets as in the LDM configuration, reconfigure disk groups with VERITAS Volume Manager for Windows. In this situation, back up file systems to a tape device, reconfigure disk groups, and then restore the systems.

2.12.4.6 Expanding Dynamic Volumes

- If you expand a dynamic volume, update the dictionary map file.
- When you expand a dynamic volume, we recommend that you do not expand it to a new physical disk that does not belong to the disk set. This expansion will cause multiple disk sets to join. If you change a volume configuration such that multiple disk sets are joined, when restoring pre-join data, you must return the dynamic disks to the basic disks, restore the disk set configuration to the pre-join configuration, and then restore all pre-join disk sets.
- Once you have expanded a dynamic volume, you cannot restore data backed up before the expansion. If you want to restore data backed up before expanding the dynamic volume, return the volume configuration to the configuration that existed before the expansion and then update the dictionary map file.

2.13 Configuring to Back Up Data Using VSS

Protection Manager can use VSS to back up data when the backup is of a file system or an Exchange database. Protection Manager requires two servers (one file server or database server, and one backup server) when it uses VSS to back up data. The file server and database server manage primary volumes. The backup server imports VSS snapshots and verifies the Exchange database. The backup server can also store backup results on tape.

In this configuration, the user executes the backup and restore commands on the file server or on the database server. On the backup server, the Protection Manager service imports VSS snapshots and verifies the Exchange database via the backup and restore commands. For more information on how to store data in a tape, see section 6.3.2 or section 8.3.2.

Figure 2.27 is an example of a system configuration where Protection Manager uses VSS to back up data.

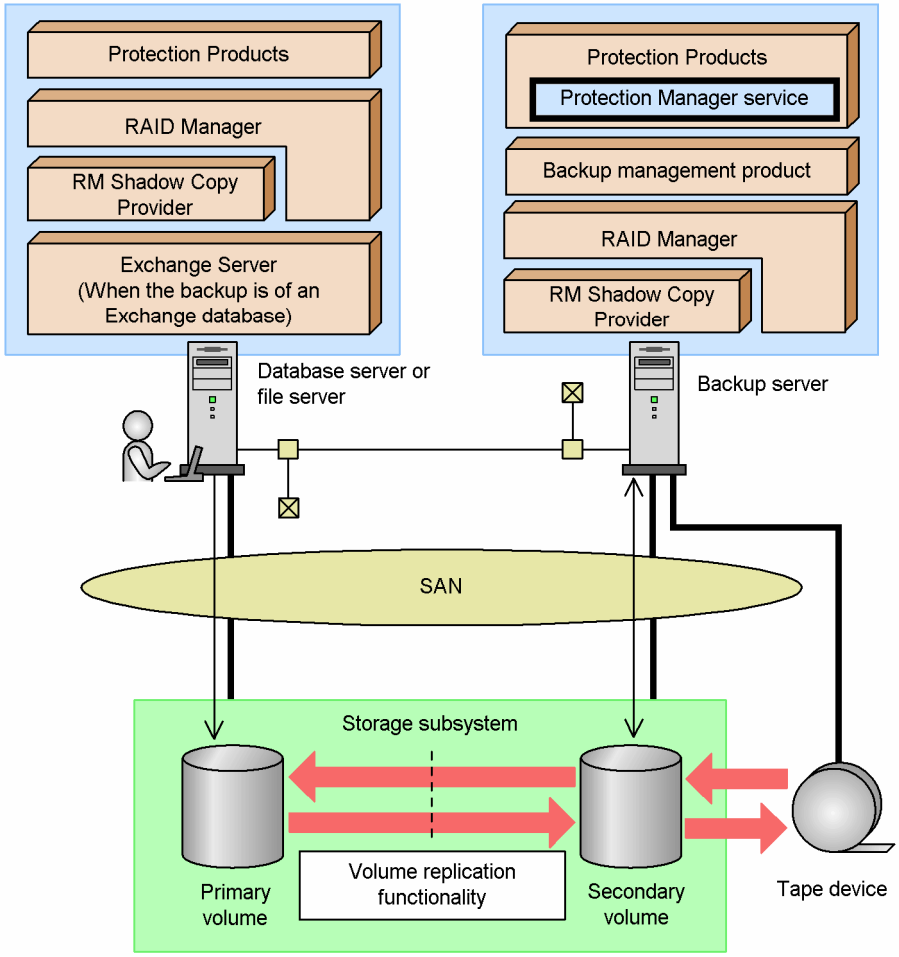


Figure 2.27 Example Configuration with Protection Product Using VSS to Back Up Data

2.13.1 Using VSS With File Systems

This section describes requirements and precautions for using VSS to back up file systems.

- RM Shadow Copy Provider must be installed both in the file server and backup server in advance.
- The backup server is required even if data is not backed up to tape, because the backup server imports VSS snapshots.
- File system write processing is halted for a given length of time (within 10 seconds) during backup. This processing is resumed after backup has been completed.

  Note: With respect to the integrity of the backup data for a file system when VSS is used, when VSS is used to back up a file system, Protection Manager programs obtain a snapshot without using VSS Writer. Although this ensures the integrity of the file system, the integrity of the data for an application that was operating during the backup cannot be ensured.
- Set up the environment of the Protection Manager programs to use VSS. For details on environment settings required to use VSS, see section 4.10.
- All the file systems subject to backup and restoration must be mounted in advance.
- Do not use `LUN#0` as a secondary volume for the backup destination. If the `LUN#0` disk is used as a secondary volume for the backup destination, other disks may not be recognized.
- If the Windows Disk Management window is being displayed on the backup server or on the file server to which no service packs have been applied in Windows Server 2003, close the window before performing the VSS backup and restore. If the command is executed with the window remaining open, the VSS backup or restore might fail.
- Do not mount another volume in a directory on the volume to be backed up. If another volume is mounted, mounting and restoration for the secondary volume may fail.
- If you change the settings of the following files, use the Services window of **Computer Management** in Windows to restart the Protection Manager service:
  - Protection Manager configuration definition file (`init.conf`)
  - RAID Manager-linkage definition file (`DEFAULT.dat`)
- If VERITAS Snapshot Provider (VSP) is installed on the database server or backup server where a VSS backup will be performed, make sure you use version VSP1.04 or later.

  If a VSS backup is executed using version VSP1.03 or earlier, the volume to be backed up might be left with `Read-Only` and `Hidden` attributes. If this happens, the drive will not be allocated at system startup.

  If this occurs, release these attributes as described in Microsoft Knowledge Base Article 840781.

- If a system failure or other problem causes a VSS backup to be interrupted, the volume that is the target of the backup might be left with the `Read-Only` and `Hidden` attributes. If these attributes remain, the drive will not be allocated at system startup. If this occurs, release the attributes as per Microsoft Knowledge Base Article 840781. Note that to perform the operations described in Knowledge Base Article 840781, you need to apply the hotfix described in Microsoft Knowledge Base Article 831112.
- File systems on a dynamic disk cannot be backed up.

2.13.2 Using VSS With Exchange Databases

This section describes requirements and precautions when VSS is used to back up an Exchange database.

- The following database products are subject to backup:
  - Exchange Server 2003 Enterprise Edition
  - Exchange Server 2003 Standard Edition
  - Exchange Server 2007 Enterprise Edition
  - Exchange Server 2007 Standard Edition
- RM Shadow Copy Provider must be installed in advance, on both the database server and the backup server.
- The backup server is required even if data is not backed up to tape. This is because the backup server imports VS snapshots and verifies the Exchange database.
- If VERITAS Snapshot Provider (VSP) is installed on the database server or backup server where a VSS backup will be performed, make sure you use version VSP1.04 or later.

  If a VSS backup is executed using version VSP1.03 or earlier, the volume to be backed up might be left with `Read-Only` and `Hidden` attributes. If this happens, the drive will not be allocated at system startup and Exchange Server will not operate normally.

  If this occurs, release these attributes as described in Microsoft Knowledge Base Article 840781.
- If a system failure or other problem causes a VSS backup to be interrupted, the volume that is the target of the backup might be left with the `Read-Only` and `Hidden` attributes. If these attributes remain, the drive will not be allocated at system startup, and Exchange Server will not operate normally. If this occurs, release the attributes as per Microsoft Knowledge Base Article 840781. Note that, to perform the operations described in that article, you need to apply the hotfix described in Microsoft Knowledge Base Article 831112.
- You can acquire backup or delete transaction logs while Exchange Server is operating. However, you cannot back up data while Exchange Information Store service is shut down.
- During backup, the database write processing of Exchange Server shuts down for a certain period (up to 10 seconds). During that period, the database write processing of Exchange Server (such as sending mail) is suspended. After backup finishes, the database write processing of Exchange Server is resumed.
- Since the number of log files might increase during backup operations by VSS, the backup results might contain a log file that has not been recorded in the catalog.

- Set the environment of the Protection Manager programs to use VSS. For details on environment settings required to use VSS, see section 4.10.
- The Information Stores of the storage group that are to be backed up must all be mounted in advance.
- For storage groups backed up by using version 04-40 or earlier, the Information Stores of the storage groups to be restored must all be mounted in advance. For storage groups backed up using version 05-50, the Information Stores do not need to be mounted in advance in order to be restored.
- If the name of a mounted storage group or Information Store has been changed, unmount all of the Information Stores in the storage group and then remount them.
- Do not use LUN#0 as a secondary volume for the backup destination. If the LUN#0 disk is used as a secondary volume for the backup destination, other disks may not be recognized.
- If the Windows Disk Management window is being displayed on the backup server or on the database server to which no service packs have been applied in Windows Server 2003, close the window before performing the VSS backup and restore. If the command is executed with the window remaining open, the VSS backup or restore might fail.
- When data is backed up, a folder named `DRMVSSMETADATA_`*`backup-ID`* is created in the root directory of the secondary volume. Do not delete this folder, as a backup metadata file necessary for restoration is stored in it. This folder is automatically deleted after data has been restored.
- Do not mount another volume in a directory on the volume to be backed up. If another volume is mounted, mounting and restoration for the secondary volume may fail.
- It may take a long time to execute the backup command for VSS backup, because the VSS import server verifies the backup result database. The time required to verify the database depends on factors such as the database capacity, the backup server's machine performance, and disk performance. Also, an option can be specified in the `drmexgbackup` or `EX_DRM_EXG_BACKUP` command used for completing the backup operation without verifying the database. If you omit the verification, we recommend that you use the `drmexgverify` or `EX_DRM_EXG_VERIFY` command to verify the database before performing restoration or backing up data to tape.

  Note that, although an option is provided for the `drmexgbackup` or `EX_DRM_EXG_BACKUP` command to delete transaction logs after backup without verification, if this option is used when the database is corrupted, roll-forward recovery might not be able to be performed.
- The following are the files verified during database verification when backup is performed by using VSS:
  - Data files in the information store (`*.edb`)
  - Transaction log files that have not been committed (`*.log`)
- Note on approximate periods required for database verification when a backup is performed by using VSS:

  Protection Manager programs use the `ESEUTIL` utility to verify databases. Table 2.6 shows the approximate periods required to verify a database. This table shows the approximate periods under certain conditions, and does not guarantee product performance.

Table 2.3 Approximate Periods Required for Database Verification for Backup by Using VSS

System configuration for the database server and backup server	Capacity of the data file (*.edb)	Number of transaction log files (*.log)	Period of time required for database verification
CPU: Pentium 4 1.5 GHz Memory: 1 GB Storage subsystem: Lightning 9970V	500 MB	2	10 seconds
	2 GB	2	35 seconds

Note: These are approximate periods under certain conditions, and do not guarantee product performance.

After the backup completes, the RAID Manager instance of the backup server is restarted. If the next backup or restore is executed during this restarting process, the command waits for the completion of the restart.

The table above shows the approximate periods under certain conditions, and does not guarantee product performance.

- If you change settings for the following files, restart the Protection Manager service:
  - Protection Manager configuration definition file (`init.conf`)
  - RAID Manager-linkage definition file (`DEFAULT.dat`)

For the storage groups subject to backup and restoration, do not enable circular logs. If a circular log is enabled, you cannot perform backup and restoration by using VSS.

2.14 Requirements and Precautions for Databases Subject to Backup

This section describes requirements and precautions for SQL Server and Exchange databases subject to backup.

2.14.1 Requirements and Precautions for SQL Server Databases

Protection Manager for SQL supports the following SQL Server database products:

- SQL Server 2000 Enterprise Edition
- SQL Server 2000 Standard Edition
- SQL Server 2005 Enterprise Edition
- SQL Server 2005 Standard Edition
- SQL Server 2005 Workgroup Edition

Table 2.4 lists the types of data subject to backup in Protection Manager for SQL. Backed-up databases differ depending on option specifications in the backup command (`the drmsqlbackup or EX_DRM_SQL_BACKUP` command).

Table 2.4 Types of Data Subject to Backup in Protection Manager for SQL

Option specification in backup command	Database subject to backup	Type of file subject to backup	Backup file name	Storage destination for backup files
-system	master	Data file	Same as the backup source file name	Secondary volume
		Transaction log file	Same as the backup source file name	Secondary volume
		Meta file[1]	Varies depending on the VDI metafile storage directory specified with the -drmsqlinit command[2]	
	model	Data file	Same as the backup source file name	Secondary volume
		Transaction log file	Same as the backup source file name	Secondary volume
		Meta file[1]	Varies depending on the VDI metafile storage directory specified with the -drmsqlinit command[2]	
	msdb	Data file	Same as the backup source file name	Secondary volume
		Transaction log file	Same as the backup source file name	Secondary volume
		Meta file[1]	Varies depending on the VDI metafile storage directory specified with the -drmsqlinit command[2]	
	User database	Data file	Same as the backup source file name	Secondary volume
		Transaction log file	Same as the backup source file name	Secondary volume
		Meta file[1]	Varies depending on the VDI metafile storage directory specified with the -drmsqlinit command[2]	
	Distribution database	Data file	Same as the backup source file name	Secondary volume
		Transaction log file	Same as the backup source file name	Secondary volume
		Metafile[1]	Depends on the VDI metafile storage directory specified in the -drmsqlinit command[2]	

Option specification in backup command	Database subject to backup	Type of file subject to backup	Backup file name	Storage destination for backup files
No assignment for -system	User database	Data file	Same as the backup source file name	Secondary volume
		Transaction log file	Same as the backup source file name	Secondary volume
		Meta file[1]	Varies depending on the VDI metafile storage directory specified with the -drmsqlinit command[2]	

[1]This file is created when the backup command executes.

[2]When a VDI metafile storage directory was registered by using the drmsqlinit command, the metafile is stored in the registered directory. The file name is *backup-ID_database-ID*.dmp. When a VDI metafile storage directory was not registered by using the drmsqlinit command, the metafile is stored in the directory that contains the file whose management number (file_id) for SQL Server in the database file is a minimum value. The file name is META_*database-ID*.dmp.

Protection Manager for SQL is backed up and restored in units of physical volumes, so the object configuration of a SQL Server database requires the following:

- Configure the data files that comprise one instance on one disk drive. Do not place data files that comprise multiple instances on one disk drive.
- Store the following directory and file on a physical volume that does not contain data files (*.mdf and *.ndf):
  - Metafile storage directory (except when metafile setting is blank)
  - fBackup data of the transaction log

  Roll-forward recovery processing can be performed if the above directory and file are placed on separate volumes. This is because, even if a restoration is performed, the metadata and transaction log file cannot return to the status they were in when the backup was performed.

  Backup processing for Protection Manager checks whether these requirements are satisfied. If they are not satisfied, backup processing will terminate with an error.
- You can use the following characters when naming a database:
  - ASCII characters
  - Multi-byte characters (one character must be expressed by using 1 or 2 bytes)

  However, do not use the following characters:

  \ / : , ; * ? < > |
- Do not use the name META_*database-ID* (ten digits).dmp for a data file of a database or transaction log file.
- In a cluster environment, specify a user on each node for the owner of a database subject to backup. If failover is performed to a node where the database-owner user does not exist, backup fails because the database owner is unknown.

- Backup data cannot be restored or recovered by specifying the `-undo` option during execution of the `drmsqlinit` command when there is no registered directory for storing the `UNDO` file.
- Backup for the transaction log cannot be executed during execution of the `drmsqlinit` command when there is no registered directory for storing the transaction log backup file.
- If the system databases (master, model, and msdb) are to be backed up, specify, as the output destination for SQL Server error log files, a volume different than the one that contains the system databases. If the system database to be backed up is specified as the destination, when the system database is restored from the secondary volume, the error log files on the SQL Server are also restored. As a result, the contents of the error log files return to the state they were before the backup, and consequently the error log information generated after the backup will be lost.
- Store `tempdb` in a volume other than the user database to be backed up. If it is stored in the user database subject to backup then, when the user database is restored, `tempdb` is overwritten by the data restored from the secondary volume while it is online, and consequently the SQL Server goes into an abnormal state. In this case, restart the SQL Server to recreate `tempdb` and restore the server to a normal state. After restarting the SQL Server, specify a volume other than the user database to be backed up, and then re-execute the restoration. Before re-executing restoration, see section 7.1.10 to determine whether the database can properly be restored.
- The database cannot be restored to SQL Servers with the version different from the one at backup. For example, a database backed up with SQL Server 2000 cannot be restored to SQL Server 2005.
- The database cannot be backed up on a TagmaStore AMS/WMS series by using the `drmsqlbackup` or `EX_DRM_SQL_BACKUP` command if the Copy-on-write SnapShot function is used and the SQL Server metadata is stored on the primary volume.
- In a cluster environment, install an SQL Server instance as a failover cluster.

The following points must be considered if you are using SQL Server 2005:

- Databases that use the snapshot function cannot be backed up and restored by Protection Products.
- The service broker becomes disabled for databases that are restored by using the `drmsqlrestore` command. To enable the service broker, execute the following SQL statement after restoration:

  `ALTER DATABASE [database name` ***database-name***`] SET ENABLE_BROKER`
- If any of the following commands is executed on a database where the database mirroring function is used, execute the command on the principal server:

  `drmsqlbackup`, `drmsqllogbackup`, `drmsqldisplay -refresh`
- If a database that uses the database mirroring function is to be restored, release the database mirroring setting before restoring the database.
- If the system databases (master, model, and msdb) are to be restored in addition to a database that uses the database mirroring function, release the mirroring setting on the database that is to be restored, restore the system databases, and then restore the database that uses the database mirroring function.

2.14.1.1 Allocating Files that Comprise a Database

Allocate the database file backed up by Protection Manager to a volume separate from files not to be backed up.

For Protection Products, only files comprising the database (the data file, index file, and transaction log file) and metafiles are subject to backup. Metafiles are generated when the backup command is executed.

Allocate only the database configuration files subject to the freezing processing in the primary volume.

Allocation of database configuration files is shown in Figure 2.28.

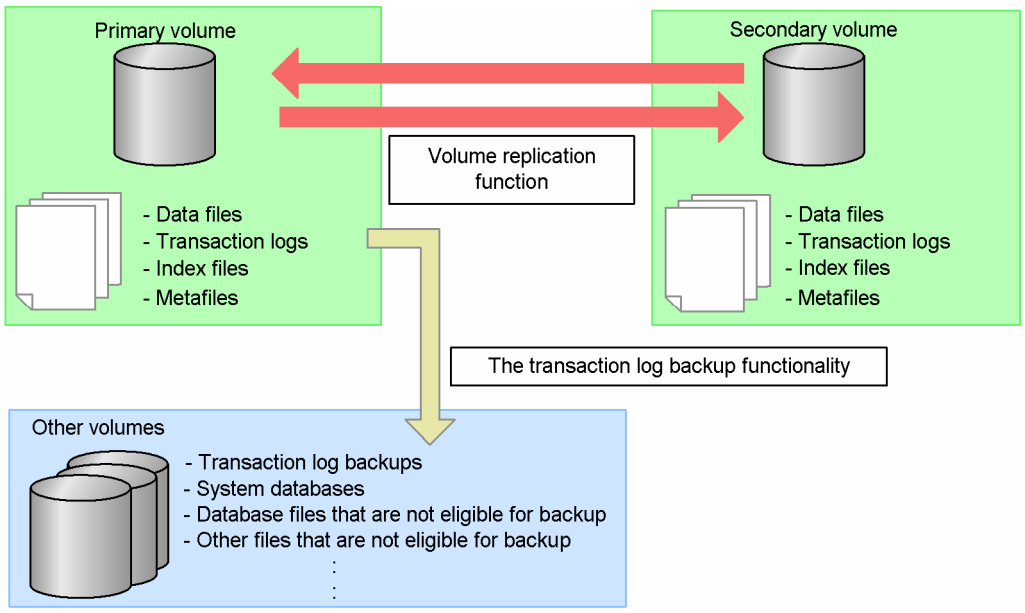


Figure 2.28 Allocation of Database Configuration Files

2.14.1.2 Allocating Multiple Database Configuration Files to One Volume

When multiple database configuration files are allocated to a volume, all databases need to be backed up or restored in a group.

Figure 2.29 shows how backup and restoration are performed in this case.

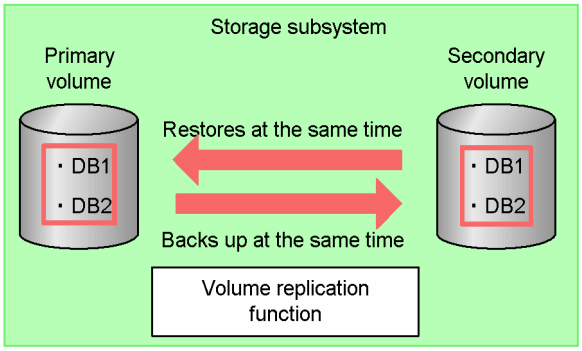


Figure 2.29 When Multiple Database Configuration Files are Allocated to a Volume

2.14.1.3 Allocating Metafiles

A metafile is a file SQL Server outputs during backup processing. Metafiles store information about database composition and are therefore used in restoration processing. For this reason, when backup data is backed up to tape, the metafile must also be backed up to the tape.

If you have specified a VDI metafile storage directory by using the `drmsqlinit` command, when backup data of the secondary volume is backed up to tape, the metafile must also be backed up to the tape (this is not necessary if the metafile is stored in the same directory as the database file).

When you back up to tape, use FTP or a similar method to transfer the metafile created in the metafile storage directory in the database server to the backup server. Transfer the file in the directory using the same name as in the metafile storage directory in the backup server and allocate the same drive characters.

Metafile allocation on the database server and the backup server is shown in Figure 2.30.

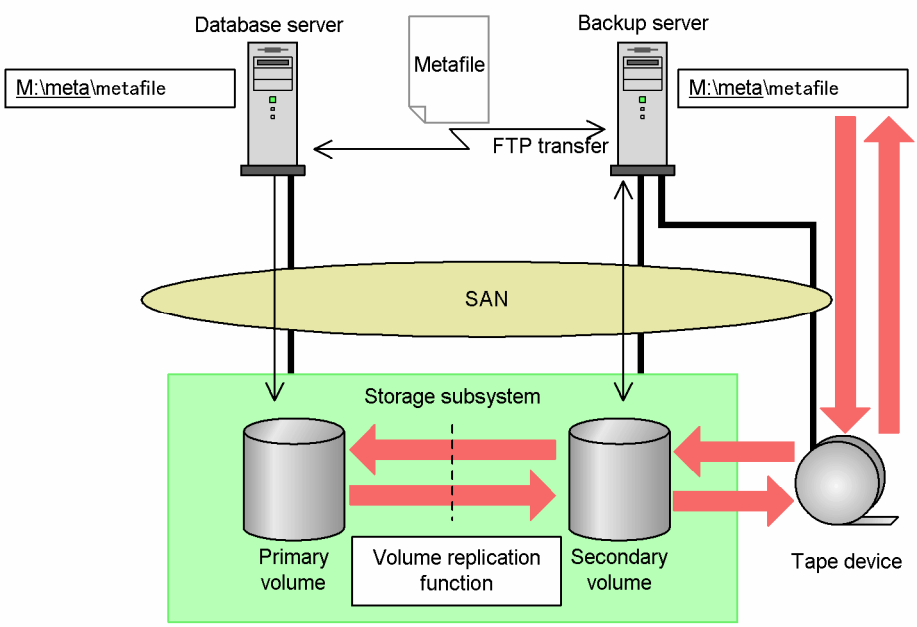


Figure 2.30 Metafile Allocation

In the cluster configuration where the database server uses Microsoft Cluster Service, when a metafile is stored in a volume on the Lightning or Thunder Series, the file can be restored from the secondary volume even when the cluster server fails over.

We recommend that you store metafiles in a volume belonging to the resource group of SQL Server so that, even if the cluster resource of SQL Server fails over, you can use the files by having them fail over at the same time.

Metafile allocation for cluster configuration of the operation standby type (active-passive) is shown in Figure 2.31.

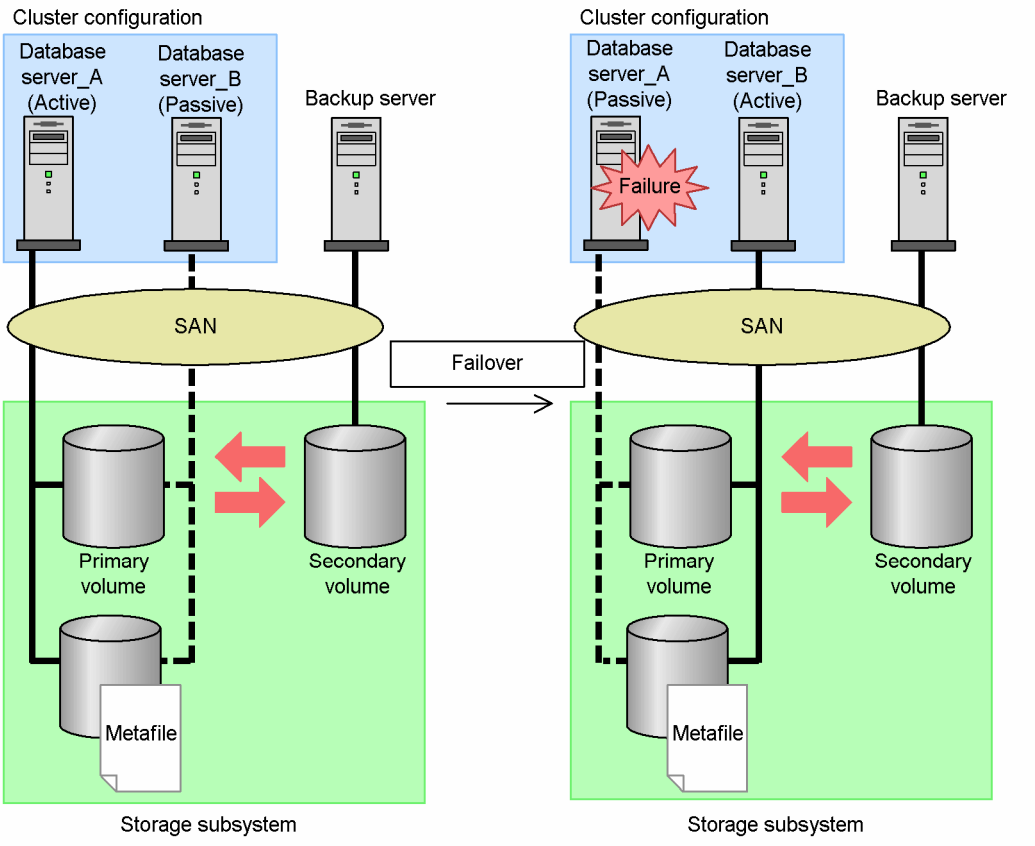


Figure 2.31 Metafile Allocation (in the Cluster Configuration)

2.14.1.4 Using a Dynamic Disk Configuration

- Do not allocate a data file made up of multiple instances to a file system (drive) of a disk group subject to backup and restoration.
- Allocate the following directories and files to a file system (drive) of a disk group different from the one to which data files (`*.mdf` and `*.ndf`) are allocated. If these directories and files are allocated to a file system (drive) of the same disk group, an error will occur when backup is executed.
  - Metafile storage directory (only if a VDI metafile storage directory was specified with the `drmsqlinit` command)
  - Backup data of the transaction log
- To allocate a file that is made up of multiple databases to a file system (drive) of the same disk group, you must collectively back up and restore all the databases. If these requirements are not met when performing Protection Manager backup, backup processing terminates with an error.

- To back up or restore databases individually, allocate database data files to a file system (drive) of a different disk group.
- Store `tempdb` in a file system (drive) in a disk group other than that in which the user database subject to backup is located. If it is stored in a file system (drive) of the same disk group then, when the user database is restored, `tempdb` is overwritten by the data restored from the secondary volume, while it is online, and consequently the SQL Server goes into an abnormal state. In this case, restart the SQL Server to recreate `tempdb` and restore the server to a normal state. After restarting the SQL Server, specify a file system (drive) in a disk group other than that in which the user database subject to backup is located, and then re-execute the restoration. Before re-executing restoration, see section 7.1.10 to determine whether the database can properly be restored.

2.14.1.5 Using a Cluster Configuration

Do not register the service resources of multiple SQL Server instances into a cluster group.

2.14.1.6 Using the VERITAS Cluster Server for Windows Configuration

- The file system that stores registry replication data should not be backed up by Protection Manager for SQL. Make sure that you allocate such a file system to a disk group or physical disk that is not subject to backup.
- Protection Manager programs use VERITAS Cluster Server Enterprise Agent for SQL Server 2000 to manage the SQL Server 2000 services. Do not change the default name for the resource type `SQLServer2000` provided by VERITAS Cluster Server Enterprise Agent for SQL Server 2000. VERITAS Cluster Server Enterprise Agent for SQL Server 2000 is bundled with VERITAS Storage Foundation HA for Windows.

2.14.2 Requirements and Precautions for Exchange Databases

The table below shows the Exchange database products supported by Protection Manager for Exchange.

Table 2.5 Exchange data products supported by Protection Manager for Exchange

Product name	Protection Manager for Exchange (2000/2003 version)	Protection Manager for Exchange 2007
Exchange 2000 Enterprise Server	Yes	--
Exchange 2000 Server	Yes	--
Exchange Server 2003 Enterprise Edition	Yes	--
Exchange Server 2003 Standard Edition	Yes	--
Exchange Server 2007 Enterprise Edition	--	Yes
Exchange Server 2007 Standard Edition	--	Yes

The following are the types of data subject to backup in Protection Manager for Exchange.

Table 2.6 Types of Data Subject to Backup in Protection Manager for Exchange

Database subject to backup	Files subject to backup	Extension
Exchange Server storage group	Data files (Exchange Server Information Store)	*.edb *.stm#
	Transaction log files	*.log
	Checkpoint files	*.chk

Files with this extension do not exist in Exchange Server 2007.

Protection Manager for Exchange is backed up and restored in units of physical volumes, so the configuration of objects of an Exchange database has the following conditions:

- The files to be backed up must all be placed on a RAID volume defined as a pair.
- Deploy the storage group taking into consideration that backup is conducted in units of physical volumes. If you deploy multiple storage groups in the same physical volume, you need to back up or restore these storage groups as one group. When backing up or restoring a storage group separately, deploy the storage group on a separate physical volume.
- When you place an Exchange database that is subject to backup on the volume, note the following:
  - You cannot put the transaction log file (`*.log`) on a volume where a data file is placed.
  - For online backup and VSS backup, data files (`*.edb, *.stm`) and checkpoint files (`*.chk`) cannot be allocated to the same volume.
- Do not use the following characters when setting a name for a storage group:

  `= ; \ /`
- The operations shown below change information pertaining to the Exchange Server on the Active Directory, so if you perform any of these operations you will no longer be able to restore backup data obtained before these operations (that is, you will no longer be able to use the `drmexgrestore` or `EX_DRM_EXG_RESTORE` command):
  - Modification of the paths of the Information Store, log, and system files.
  - Addition or deletion of storage groups.
  - Modification of the storage group name.
  - Addition of an Information Store to a storage group.
  - Deletion of an Information Store from a storage group.

  If you have performed any of the above operations, you will need to back up the Active Directory and the Exchange database again. To perform a backup for the Active Directory, follow the instructions given in documentation provided by Microsoft.

- Performing either of the operations shown below changes the database name, and you will no longer be able to restore and recover backup data obtained before these operations (that is, you will no longer be able to use the `drmexgrestore` or `EX_DRM_EXG_RESTORE` command with the `-recovery` option specified):
  - Repairing Information Store using the `ESEUTIL` utility (`ESEUTIL /p`).
  - Defragmentation using the `ESEUTIL` utility (`ESEUTIL /d`).

  If you have performed either of these operations, you will have to execute the `drmexgbackup` or `EX_DRM_EXG_BACKUP` command, and then back up the Exchange database again.
- You can use a circular log only when using Protection Manager for Exchange (2000/2003 version) to perform a cold backup or online backup. However, if you use a circular log, the oldest log among the existing log files is deleted when a new log file is created. This means that some logs might disappear at roll-forward. For this reason, when performing roll-forward recovery, do not use a circular log.
- The recovery storage groups for Exchange Server 2003 and Exchange Server 2007 cannot be backed up. The files and folders that make up the restoration storage group must be placed on a different file system from the storage groups to be backed up.
- When using QuickShadow, you cannot back up to tape by using Protection Manager for Exchange. So, when using QuickShadow, use another backup product to back up (to tape) an Exchange database that is on the primary volume.
- When using Exchange Server 2007, only VSS backup can be used.
- Out of the high-available functions of Exchange Server 2007, only single copy cluster (SCC) is supported. Neither cluster continuous replication (CCR) nor local continuous replication (LCR) is supported.
- In a cluster environment, when restoring data in a physical node different from that used during backup, an Exchange virtual server that uses the same shared disk as that used during backup must be running. Protection Manager for Exchange cannot restore data for an Exchange virtual server, such as one at a remote site, that uses a disk different from the one used during backup.

2.14.2.1 Restoring Backup Data Obtained Using VSS in Units of Information Stores

You can restore backup data obtained using VSS in units of Information Stores (in units of `*.edb` and `*.stm` data files). Restoring backup data in units of Information Stores makes it possible to quickly restore just the required data files.

To restore backup data in units of Information Stores, you must partition the disk into units of data to be restored. Therefore, take into account the unit of restoration when allocating data files.

Database configurations that allow restoration in units of Information Stores

- Configurations in which only the Information Store data files (`*.edb` and `*.stm`) to be restored individually are stored on separate disks
- Configurations in which only Information Store data files (`*.edb` and `*.stm`) to be restored individually are stored on the same disk

- Configurations in which multiple Information Store data files within the same storage group are stored on the same disk

  You can restore all Information Stores stored on the same disk provided that you specify all the Information Stores. In this case, if you do not specify all Information Stores stored on the same disk, the command will end with an error.

Database configurations that do not allow restoration in units of Information Stores

- When data files (`*.edb` and `*.stm`), transaction log files (`*.log`), and check point files (`*.chk`) in different storage groups are located on the same disk, Information Stores below a storage group that shares the disk cannot be restored individually.

Table 2.7 lists the conditions for allocating files when restoring backup data in units of Information Stores.

Table 2.7 Conditions for Allocating Files When Restoring Backup Data in Units of Information Stores

Restoration target	Files on the same disk						
	No other files	Information Store that is a restoration target		Information Stores in the same storage group		Information Stores in different storage groups	
		*.edb	*.stm	*.edb	*.stm	*.edb	*.stm
*.edb	Can be restored individually.	Not applicable	Can be restored individually.	Can be restored provided all Information Stores on the same disk are specified.	Can be restored provided all Information Stores on the same disk are specified.	Cannot be restored individually.	Cannot be restored individually.
*.stm		Can be restored individually.	Not applicable	Can be restored provided all Information Stores on the same disk are specified.	Can be restored provided all Information Stores on the same disk are specified.	Cannot be restored individually.	Cannot be restored individually.

Figure 2.32 shows a database configuration that allows restoration in units of Information Stores.

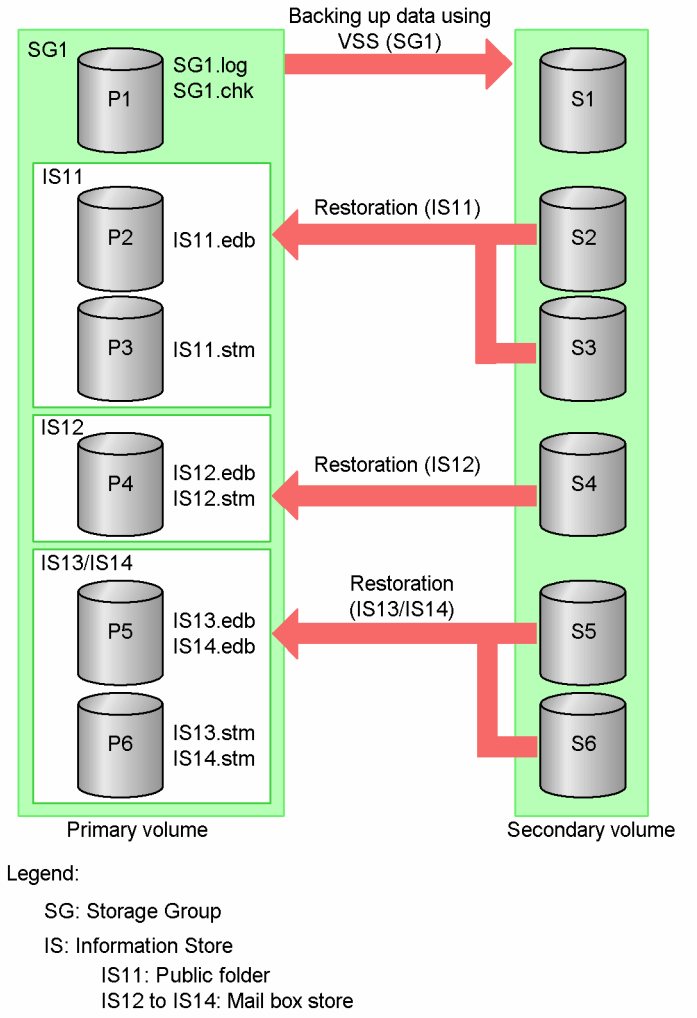


Figure 2.32 Restoration in Units of Information Stores (Backup Target Is an Exchange Database)

In the above example, Information Stores from IS11 to IS14 belong to the storage group SG1. In this case, you can restore backup data in the following units:

- Restore only IS11 individually
- Restore only IS12 individually
- Restore IS13 and IS14 at the same time (you cannot restore just one)

Notes:

– When restoring backup data in units of Information Stores, you must apply (roll forward) the transaction log at the time of restoration. For details on precautions relating to executing commands at restoration, see the description for the `drmexgrestore` command in the *HiCommand Protection Manager Command Reference* manual.

- To restore backup data in units of Information Stores, you need at least three disks for one storage group. If you are using True Copy as the volume duplication function, up to two disks can be backed up for a single storage group. This means that you cannot restore backup data in units of Information Stores.

2.14.2.2 Using a Dynamic Disk Configuration

- If multiple storage groups are allocated to a file system (drive or mount point directory) of the same disk group, you must collectively backup or restore these storage groups.
- To back up or restore storage groups individually, allocate them to a file system (drive or mount point directory) of a different disk group.
- When allocating an Exchange database subject to backup to a file system (drive or mount point directory), note the following points:
  - You cannot allocate the transaction log file (`*.log`) to a file system (drive or mount point directory) of the same disk group to which data files are allocated.
  - For online backup and VSS backup, you cannot allocate the data files (`*.edb`, `*.stm`) and checkpoint file (`*.chk`) to a file system (drive or mount point directory) of the same disk group.
- The recovery storage groups for Exchange Server 2003 and Exchange Server 2007 cannot be backed up. Allocate files and folders that comprise a recovery storage group to a file system (drive or mount point directory) of a disk group different from a storage group subject to backup.

2.14.2.3 Using a Cluster Configuration

- Do not register the instances of multiple Exchange resources into a cluster group.

2.14.2.4 Using the VERITAS Cluster Server for Windows Configuration

- The status of the cluster resource groups must be online during backup. If it is not online, the backup command (`drmexgbackup` or `EX_DRM_EXG_BACKUP`) will end in error.
- The file system that stores registry replication data should not be backed up by Protection Manager for Exchange. Make sure that you allocate such a file system to a disk group or physical disk that is not subject to backup.
- Protection Manager programs use VERITAS Cluster Server Enterprise Agent for Microsoft Exchange to manage the Exchange services and Exchange protocol services. Do not change the default name for the resource type `ExchService` and `ExchProtocol` provided by VERITAS Cluster Server Enterprise Agent for Microsoft Exchange. VERITAS Cluster Server Enterprise Agent for Microsoft Exchange is bundled with VERITAS Storage Foundation HA for Windows.

2.14.2.5 Cluster Configuration Using the Microsoft Cluster Service

- Among the cluster resource groups, the Exchange System Attendant resource and the Exchange Information Store resource must be online during backup. If they are not online, the backup command (`drmexgbackup` or `EX_DRM_EXG_BACKUP`) will result in an error.

Chapter 3 Installing Protection Manager

This chapter describes how to install and uninstall Protection Manager programs.

- Installing Protection Manager Programs (see section 3.1)
- Troubleshooting Installation Errors (see section 3.2)
- Uninstalling Protection Manager Programs (see section 3.3)

3.1 Installing Protection Manager Programs

This section describes the procedures for installing Protection Manager programs. This section describes the procedures for a first-time installation and version-upgrade installation of Protection Manager on the Windows OS.

You can install optional products (Protection Manager for SQL and Protection Manager for Exchange) in any order. For a version-downgrade installation, uninstall the current Protection Manager version and then perform a first-time installation of the desired version. The existing user files are not deleted during uninstallation and also a new installation does not overwrite the existing user files. Therefore, the user files can be inherited.

Before beginning installation, make sure of the following:

- The prerequisite products for Protection Manager programs (section 4.1) have already been installed.
- You are logged in with Administrator permissions.
- All programs are inactive.
- If the Windows firewall functionality is being used, set the firewall to enable the following items:
  - FTP Publishing Service

    Enable this in order to execute FTP transfer extended commands (`EX_DRM_FTP_GET`, `EX_DRM_FTP_PUT`).

    For details, see *http://support.microsoft.com/default.aspx?scid=kb;en;832017*

    Service overview and network port requirements for the Windows Server system.
  - RAID Manager communication port
  - Protection Manager service

    Execute the following command to make the firewall setting:

    Protection-Products-installation-destination`\bin\util\drmaddfwlist.bat`

Install Manager programs are installed in the following order:

1. Protection Manager Copy Controller
2. Protection Manager for SQL or Protection Manager for Exchange

Table 3.1 shows the installation destinations of Protection Manager programs and environment variable setup:

Table 3.1 Installation Destinations of Protection Manager Programs and Environment Variable Setup

Task		File server or database server	Backup server
Install Protection Manager programs	Protection Manager Copy Controller	Needed	Needed
	Protection Manager for SQL (when the backup is of a SQL Server database)	Needed	Not needed

	Protection Manager for Exchange (when the backup is of an Exchange database)	Needed	Not needed

Notes on Installation:

With regard to mount point and installation destination names, make sure that a character string in the system environment variable `PATH` does not exceed 1,023 bytes in length, even after paths to Protection Manager are added. If it exceeds 1,023 bytes, an error will occur.

Protection Manager will add the following path names to the system environment variable `PATH`:

Protection-Products-installation-destination`\bin`

Protection-Products-installation-destination`\lib`

Protection-Products-installation-destination`\script\bin`

The Protection Manager installer outputs installation-processing results to the installation logs. The installation logs are stored in the following locations:

Protection-Manager-installation-destination`\log\drm_inst.log`

Precautions for Changing the Installation Directory:

Keep the following in mind for the string specified for the installation path, when changing the installation directory:

- You cannot use names already reserved by the operating system, such as `CON` or `AUX`.
- You can use alphanumeric characters, space characters, and any of the following characters: `#`, `+`, `-`, `@`, `_`, `(`, `)`, and `.`.

Notes on Installation by Performing a Version Upgrade:

- When you install Protection Products by performing a version upgrade from a version earlier than 3.5, you must convert the data in the dictionary map file.

  Back up the target directory that stores the dictionary map file, and then execute the following command:

  Protection-Manager-installation-directory`\bin\util\drmdbconvert -c`

  If you use the dictionary map with virtual servers, perform data conversion for each virtual server.

- When performing an upgrade installation from a version earlier than 3.5, when installation has finished, check the value of the `PAIR_CREATE` parameter in the RAID Manager-linkage definition file (`DEFAULT.dat`).

  The `PAIR_CREATE` parameter was added to version 3.5, and is used to specify whether to perform backup of the local site or automatic pair generation during resynchronization. Automatic pair generation could be performed in Protection Manager versions earlier than 3.5, but note that in 3.5 or later versions, the default setting disables automatic pair creation. For details about the `PAIR_CREATE` parameter, see section 4.4.6.

3.1.1 Installing Protection Manager Copy Controller the First Time

To install Protection Manager Copy Controller:

1. Insert the installation CD-ROM of Protection Manager Copy Controller into the CD-ROM drive. This example assumes that the `x:` drive is the CD-ROM drive.
2. From the **Start** menu, choose **Run**, enter `x:\Setup.exe` in the **Run** dialog box, and then click the **OK** button. The **Welcome** dialog box appears.
3. Click the **Next** button. A dialog box prompting you to enter user information appears.
4. Enter the user name and company name, and then click the **Next** button. A dialog box prompting you to confirm the installation destination appears.
5. Change the installation destination from the default if required, and then click the **Next** button. You can use alphanumeric characters, space characters, and any of the following characters: `#`, `+`, `-`, `@`, `_`, `(`, `)`, and `.`.

   If you change the installation destination directory, do not specify a directory where another product is installed. If the installation directory is not changed, the following directory is used for the installation directory of Protection Manager Copy Controller:

   `C:\Program Files\HITACHI\DRM`

   When using Windows Server 2003, another installation directory might be used.

   A dialog box prompting you to select the setup type appears.
6. Select the setup type, and then click the **Next** button. There are three types of setup:
   - Standard: Installs the main Protection Manager Copy Controller programs, and the online manual.
   - Compact: Installs only the main Protection Manager Copy Controller programs.
   - Custom: Allows the user to select which components are to be installed.
7. Enter the name of the program folder, and then click the **Next** button. A dialog box prompting you to confirm the start of file copying appears.
8. Make sure that the information you specified is correct, and then click the **Next** button. Installation starts. Wait for a dialog box to appear that reports completion of the installation.

   If RAID Manager is not installed in *system-drive*`\HORCM`, a dialog box prompting you to change the setup file appears.

   The dialog box reporting completion of installation includes a message requesting that the account of the user who executes the Protection Manager service be entered.
9. Click the **Finish** button on the dialog box reporting completion of installation

   The Protection Products installation directory will be added to the `PATH` environment variable.

   A dialog box appears to set the account (user name and password) of the user who executes the Protection Manager service.
10. Enter the information in **User Name** and **Password**, and then click the **Finish** button.

The user that executes the Protection Manager service must have local Administrator permissions.

Notes:

The account of the user who executes the Protection Manager service must have local Administrator permissions and the right to log on as a service.

The specified account is authenticated and, if authentication is successful, the account setup ends. At this time, the startup type is set to **Automatic**.

If the specified account does not have permissions required for the user who executes the service, a dialog box indicating that authentication has failed appears.

If you select **Cancel** in the dialog box used to set the account of the user who executes the Protection Manager service, a dialog box confirming that you want to finish without setting up the account is diaplayed. Selecting **Yes** button in that dialog box sets the account of the user who executes the Protection Manager service to the default local system account, and the startup type to **Disabled**. In this case, select the Protection Manager service in the service window for the Windows administrative tools, change the user to a user who has Administrator permissions, and then change the startup type to **Automatic**. The right to log on as a service is automatically added to the specified user.

The Protection Manager service uses the TCP port number `22300` by default. If this port number is already used, register an unused TCP port number in the Windows `services` file using the name of DRMVSSServer. The port number setting on the file server or database server must be the same as that on the backup server.

When the Protection Manager service is disabled, the following functions cannot be used:

- Functions described in Table 5.19.
- Options that must be specified along with the `-s` option in a command (for example, `-auto_import` and `-svol_check`)

For details about the commands, see the *HiCommand Protection Manager Command Reference*.

During the first-time installation, a value is added to the system environment variable. To apply the system environment variable to the system, restart the system.

3.1.2 Performing an Upgrade Installation of Protection Manager Copy Controller

To install Protection Manager Copy Controller:

1. Insert the installation CD-ROM of Protection Manager Copy Controller into the CD-ROM drive. This example assumes that the `x:` drive is the CD-ROM drive.
2. From the **Start** menu, choose **Run**, enter `x:\Setup.exe` in the **Run** dialog box, and then click the **OK** button. The **Welcome** dialog box appears.
3. Choose **Upgrade**, and then click the **Next** button. The **Select Components** dialog box appears.
4. Select the component you want to upgrade, and then click the **Next** button. Installation starts. Wait for a dialog box reporting completion of installation to appear.

This dialog box includes a message requesting that the account of the user who executes the Protection Manager service be entered.

5. Click the **Finish** button on the dialog box reporting completion of installation.

   A dialog box is displayed to set the account (user name and password) of the user who executes the Protection Manager service.

6. Enter the information in **User Name** and **Password**, and then click the **Finish** button.

   The user that executes the Protection Manager service must have local Administrator permissions.

   Notes: The account of the user who executes the Protection Manager service must have local Administrator permissions and the right to log on as a service.

   The specified account is authenticated and, if authentication is successful, the account setup ends. At this time, the startup type is set to **Automatic**.

   If the specified account does not have permissions required for the user who executes the service, a dialog box indicating that authentication has failed appears.

   If you select **Cancel** in the dialog box used to set the account of the user who executes the Protection Manager service, a dialog box confirming that you want to finish without setting up the account is diaplayed. Selecting **Yes** button in that dialog box sets the account of the user who executes the Protection Manager service to the default local system account, and the startup type to **Disabled**. In this case, select the Protection Manager service in the service window for the Windows administrative tools, change the user to a user who has Administrator permissions, and then change the startup type to **Automatic**. The right to log on as a service is automatically added to the specified user.

   The Protection Manager service uses the TCP port number `22300` by default. If this port number is already used, register an unused TCP port number in the Windows `services` file using the name of DRMVSSServer. The port number setting on the file server or database server must be the same as that on the backup server.

3.1.3 Installing Protection Manager for SQL or Exchange Server the First Time

To install Protection Manager for SQL or Protection Manager for Exchange:

1. Insert the installation CD-ROM of Protection Manager for SQL or Protection Manager for Exchange into the CD-ROM drive. This example assumes that the `x:` drive is the CD-ROM drive.
2. From the **Start** menu, choose **Run**, enter `x:\Setup.exe` in the **Run** dialog box, and then click the **OK** button. The **Welcome** dialog box appears.
3. Click the **Next** button. The **Select the license registration method** dialog box appears.
4. Select the license registration method, and then click the **Next** button.
5. Enter the license key or the name of the license key file, and then click the **Next** button.
6. Make sure that the information you specified is correct, and then click the **Next** button. Installation starts. Wait until a message reporting that installation has finished is displayed.
7. Click the **Finish** button.

To display or update the registered license information, execute the `drmlicense` command. For example, to display Protection Manager for SQL license information, enter the following command:

```
drmlicense MSSQL -l
```

3.1.4 Installing an Upgrade of Protection Manager for SQL or Exchange

To perform an upgrade installation of Protection Manager for SQL or Protection Manager for Exchange:

1. Insert the installation CD-ROM of Protection Manager for SQL or Protection Manager for Exchange into the CD-ROM drive. This example assumes that the `x:` drive is the CD-ROM drive.
2. From the Windows **Start** menu, choose **Run**, enter `x:\Setup.exe` in the **Run** dialog box, and then click the **OK** button. The **Welcome** dialog box appears.
3. Choose **Upgrade**, and then click the **Next** button. Installation starts. Wait until a message reporting that installation has finished is displayed. When upgrading from a Protection Manager program version earlier than 3.5, you need to register a license key by following the dialog box instructions.
4. Click the **Finish** button.

3.2 Troubleshooting Installation Errors

This section describes troubleshooting for Protection Manager installation errors.

3.2.1 When Installation Fails

When an error occurs during Protection Manager installation, determine the cause of the error using the following the procedure:

To determine the cause of the error:

1. Check the installation procedure in the manual, in accordance with the message output by the installer.
2. If the cause of the error cannot be determined even after reviewing the installation procedure, examine the contents of the `drm_inst.log` file. For details on this file, see section 3.2.2.
3. If you still cannot determine the cause of the error, collect the `drm_inst.log` file and contact your support center.

3.2.2 Examining the Contents of the drm_inst.log File

The `drm_inst.log` file is output to the following location:

Protection-Manager-installation-destination\log\drm_inst.log

or

desktop-of-the-user-who-performed-installation\drm_inst.log

The following describes the contents of `drm inst.log`:

- When `LICENSE ERROR:RC=**` is output to `drm_inst.log`, the `**` part of `RC=**` indicates the numerical value explaining the error's cause.

  Review Table 3.2, and take appropriate action.

Table 3.2 Error Cause Output to drm_inst.log

RC	Cause
10, 11	The license has expired.
1 to 9, 20 to 29	The system environment is incorrect.
30 to 39	The entered license key is invalid.
40	The license key file that was entered could not be found.
41 to 43	The license key file that was entered is invalid.
44	The license key that should be entered does not exist in the license key file.

- When `ERROR:`*error-message* is output to `drm_inst.log`

The error message indicates the error cause. Follow the message and take appropriate action.

3.3 Uninstalling Protection Manager Programs

This section describes the procedures for uninstalling Protection Manager programs.

Before beginning uninstallation, make sure that:

- You are logged in with administrator permissions
- All programs are stopped.

Uninstall the Protection Manager programs in the following order. Note that this order differs from the installation order:

1. The following optional Protection Manager programs (not in order):
   - Protection Manager for SQL
   - Protection Manager for Exchange
2. Protection Manager Copy Controller

To delete a Protection Manager program:

1. Uninstall the Protection Manager program.
2. Delete the settings that are related to the Protection Manager program.

Table 3.3 lists the servers that delete Protection Manager programs:

Table 3.3 Servers From Which Protection Manager Programs are Deleted

Task		File server or Database Server	Backup server
Uninstall Protection Manager	Uninstall Protection Manager Copy Controller	Needed	Needed
	Uninstall Protection Manager for SQL (when the backup is of a SQL Server database)	Needed	Not needed
	Uninstall Protection Manager for Exchange (when the backup is of an Exchange database)	Needed	Not needed
Delete the settings related to Protection Manager programs	Delete user-defined files	Needed	Needed

To uninstall Protection Manager for SQL, Protection Manager for Exchange, or Copy Controller:

1. Open the **Add/Remove Programs** Control Panel (**Start→Settings→Control Panel→ Add/Remove Programs**).

2. In the **Currently installed programs** list, select **HiCommand Protection Manager for SQL HiCommand Protection Manager for Exchange**, or **HiCommand Protection Manager Copy Controller**, then click the **Change or Remove Programs** button. The installer dialog box appears.
3. Click the **Remove** button, and then the **Next** button. A dialog box that asks whether you really want to delete the Protection Manager program appears.
4. Click the **OK** button. Uninstallation starts. After uninstallation finishes, a dialog box appears, indicating the completion of maintenance.
5. Click the **Finish** button.
6. (Copy Controller uninstallation only): Delete any user files that were created after installation: for example, definition files and log files.
7. (Coy Controller uninstallation only): Delete the system environment variable. Delete the Protection Manager Copy Controller installation folder from the `PATH` system environment variable. For details on how to change environment variables, see the Windows help.

Chapter 4 Preparing to Use Protection Manager

This chapter defines the environment setup and explains the the flow required for using Protection Manager.

- Setting up an Environment for Protection Manager (see section 4.1)
- Setting up RAID Manager (see section 4.2)
- Setting up Operation of Protection Manager (see section 4.3)
- Setting up Protection Manager Linkage with RAID Manager (see section 4.4)
- Creating a Dictionary Map File (see section 4.5)
- Creating a Dictionary Map File (see section 4.6)
- Specifying Settings for a Cluster Configuration (see section 4.7)
- Creating a Database Configuration Definition File (see section 4.8)
- Specifying Settings Related to Linkage with a SQL Server (see section 4.9)
- Specifying Settings for Using VSS (see section 4.10)
- Setting up Linkage with a Backup Management Product (see section 4.11)
- Creating a List File (see section 4.12)
- Updating a Dictionary Map File (see section 4.13)
- Specifying How to Mount a Secondary Volume (see section 4.14)
- Preparations Required Before Extended Commands (see section 4.15)
- Creating a User Script (see section 4.16)
- Specifying Settings for Sending Mail (see section 4.17)
- Backing up a Protection Manager Operating Environment (see section 4.18)

4.1 Setting up an Environment for Protection Manager

Table 4.1 shows the procedures for setting up an environment for Protection Products and the tasks required for each server.

Sometimes Protection Manager commands need to be executed during environment setup. Administrator permissions are required to execute Protection Manager commands. For details on user permissions for executing commands, see section 5.4.1.

Note: If you omit specifying the parameters for setting up an environment or specify an invalid value (a value that cannot be set), the default values will be used.

Table 4.1 Protection Manager Environment Setup Tasks Needed for Each Kind of Server

Environment Setup Task		File Server or Database Server	Backup Server	File Name	Sections and Documents for Reference
Set up the environment for prerequisite products.[1]	Set up the volume replication functionality of the storage subsystem.	Not needed[2]	Not needed[2]	File setting is not required	Documentation for the storage subsystem
	Set up RAID Manager	Needed	Needed	RAID Manager configuration definition file (`horcmn.conf`)	Section 4.2
	Set up the backup management product.	Not needed	Needed	File setting is not required	Documentation for the backup management product
Set up Protection Manager operations.		Needed	Needed	Protection Manager configuration definition file `init.conf`	Section 4.3
Specify settings for linkage with RAID Manager.		Needed	Needed	RAID Manager-linkage definition file `DEFAULT.dat`	Section 4.4
Create a dictionary map file.		Needed	Needed	File setting is not required	Section 4.5
Create a cluster information configuration definition file.		Needed	Not needed	Cluster information configuration definition file (*`cluster-name`*`.dat`)	Section 4.7
Create a database configuration definition file (when the backup is of a SQL Server database).		Needed	Not needed	Database configuration definition file (*`instance-name`*`.dat`)	Section 4.8
Specify settings related to linkage with a SQL server[#3]	Specify the SQL Server login timeout option Specify the SQL Server automatic recovery time	Needed	Not needed	`sqlinit_`*`instance-name`*`.conf`	Section 4.9
Specify settings for using VSS (when the backup is of a file system or an Exchange database).		Needed	Needed	VSS definition file (vsscom . conf)	Section 4.10

Environment Setup Task	File Server or Database Server	Backup Server	File Name	Sections and Documents for Reference
Set up linkage with a backup management product.	Not needed	Needed	Configuration definition file for linkage with a backup management product (DEFAULT.dat)	Section 4.11
			Configuration definition file for tape backup	
Specify settings for batch operations.	Needed	Needed	Any file	Section 4.12
Update the dictionary map file.	Needed	Needed[4]	File setting is not required	Section 4.13
Perform preparation for executing extended commands.	Needed	Needed	File setting is not required	Section 4.15
Create a user script.	Needed	Not needed	User script file (any file name)	Section 4.16
Specify settings for sending mail.	Needed	Not needed	Mail send definition file (mail.conf)	Section 4.17
Back up the operating environment of Protection Manager.	Needed	Needed	File setting is not required	Section 4.18

[1]To use SQL Server's log-shipping functionality, the source server and distribution-destination servers must have the same environment settings as the database server.

[2]In each storage subsystem, set up the volume replication functionality of the storage subsystem.

[3]Specify these settings when a command has ended with an error (with the error message KAVX1008-E).

[4]This is not required for a backup server. When a primary volume exists in the backup server, the dictionary map file must be updated. When the primary volume does not exist in the backup server, updating of the dictionary map file is not necessary.

4.2 Setting up RAID Manager

To back up data using Protection Manager, you need to create a RAID Manager configuration definition file, `horcmn.conf`, for each instance to match the intended usage. In `horcmn.conf`, *n* is the instance number.

The location of the RAID Manager configuration definition file (`horcmn.conf`) is as follows:

- In Windows 2000 Server: *system-drive*`\WINNT`
- In Windows Server 2003: *system-drive*`\Windows`

Note: Do not specify a path to the configuration definition file in the `HORCM_CONF` environment variable.

In Protection Manager, a pair of a primary volume and secondary volume is managed as a copy group. A copy group is defined by a combination of `dev_group` and `dev_name` specified in a RAID Manager configuration definition file.

Protection Manager programs do not allow duplication of copy group names on the same server. If more than one RAID Manager instance is activated, make sure that the combination of `dev_group` and `dev_name` is unique among all the RAID Manager configuration definition files.

Note: Of the definition items in the configuration definition file (`horcmn.conf`), Protection Products programs do not support the `HORCM_LDEV` section.

To check whether the RAID Manager configuration definition file you created is correct, test whether a RAID Manager instance can be activated. For details on how to activate RAID Manager, see the RAID Manager documentation. After you make sure that the RAID Manager instances can be activated correctly, install the Protection Manager programs.

This section describes how to create configuration definition files for the following operations:

- Back up one primary volume to multiple generations
- Back up multiple primary volumes to a single generation
- Back up multiple primary volumes to multiple generations
- Back up or restore between storage subsystems
- Set up a multi-target or cascade configuration

4.2.1 Backing up One Primary Volume to Multiple Generations

This configuration is used when, for example, all the files and databases to be backed up are stored in one volume and multiple backup generations are obtained at set time intervals.

To back up one primary volume to multiple generations:

1. Define the given primary volume and a secondary volume pair from each generation.
2. Specify 0, 1, or 2 as the `MU#` (pair identifier) of each pair.

Figure 4.1 shows the volume configuration and the configuration definition to back up one primary volume to multiple generations.

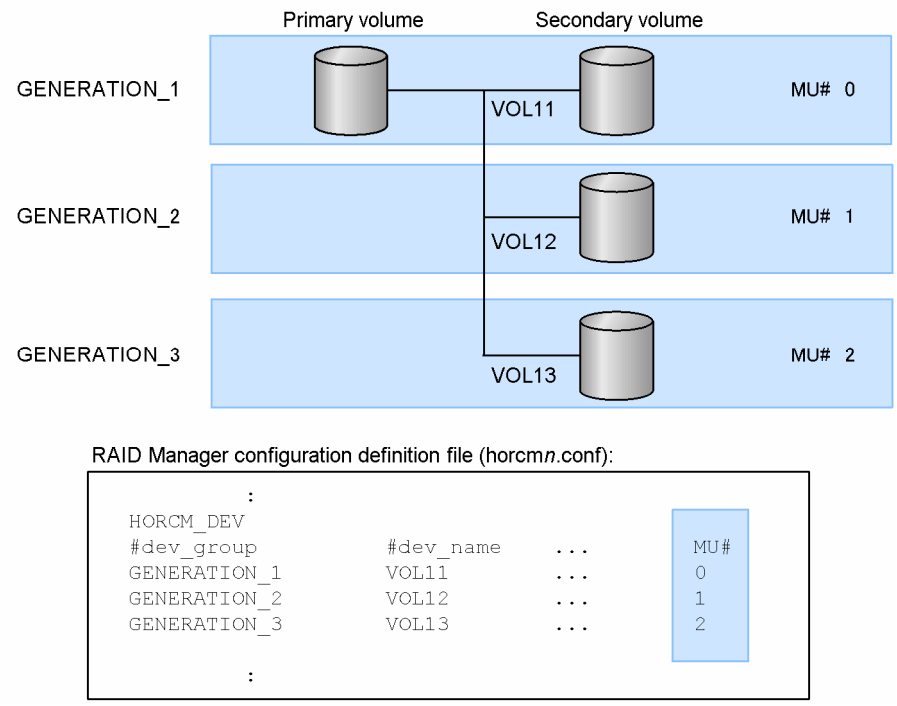


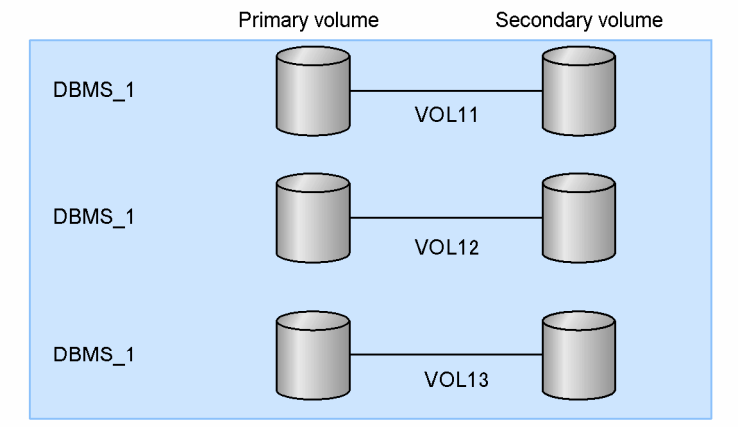
Figure 4.1 Volume Configuration to Back up One Primary Volume to Multiple Generations

4.2.2 Backing up Multiple Primary Volumes to a Single Generation

This configuration is used when, for example, the SQL Server master database and user database are stored in different volumes.

To back up multiple primary volumes to a single generation, specify the same `dev_group` name for all pair volumes to be backed up in a batch.

Figure 4.2 shows the volume configuration and configuration definition to back up multiple primary volumes to a single generation.



RAID Manager configuration definition file (horcm*n*.conf):

```
          :
HORCM_DEV

#dev_group              #dev_name       ...
DBMS_1                   VOL11          ...
DBMS_1                   VOL12          ...
DBMS_1                   VOL13          ...

          :
```

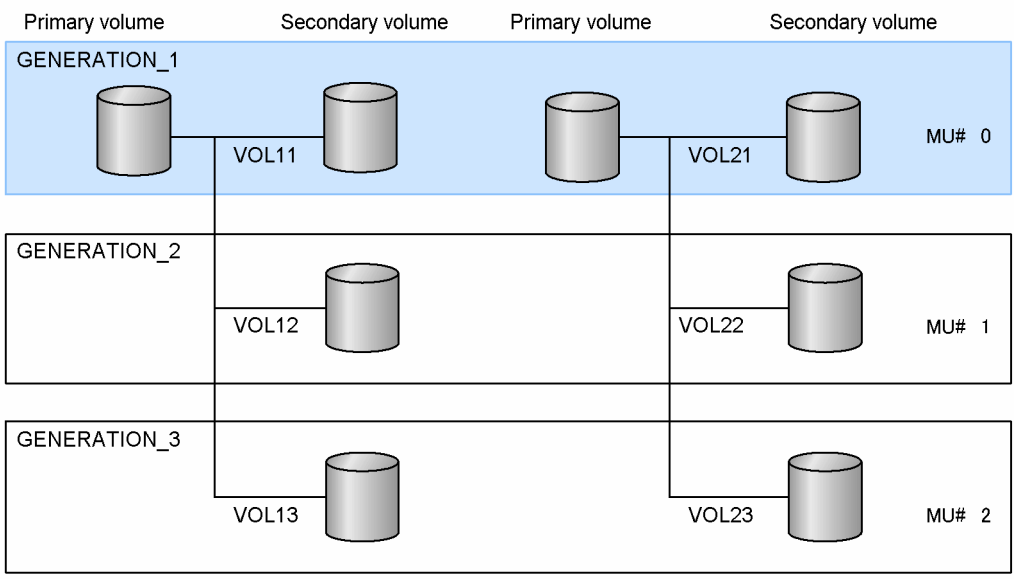
Figure 4.2 Volume Configuration to Back up Multiple Primary Volumes to a Single Generation

4.2.3 Backing up Multiple Primary Volumes to Multiple Generations

This configuration is used when, for example, backups should be taken by the combination of the above two methods, backing up one primary volume to multiple generations and backing up multiple primary volumes to a single generation.

To back up multiple primary volumes to multiple generations, specify the same `dev_group` name for all pair volumes that have the same `MU#`.

Figure 4.3 shows the volume configuration and configuration definition to back up multiple primary volumes to multiple generations.



RAID Manager configuration definition file (horcm*n*.conf):

```
          :
HORCM_DEV
#dev_group          #dev_name     ...        MU#
GENERATION_1        VOL11         ...        0
GENERATION_2        VOL12         ...        1
GENERATION_3        VOL13         ...        2
GENERATION_1        VOL21         ...        0
GENERATION_2        VOL22         ...        1
GENERATION_3        VOL23         ...        2
          :
```

Figure 4.3 Volume Configuration to Back up Multiple Primary Volumes to Multiple Generations

4.2.4 Backup or Restore Between Storage Subsystems

Protection Manager allows you to backup or restore with mixed volume replication functionalities between subsystems such as ShadowImage and TrueCopy or UR. In this case, setup to enable a remote copy is required in the RAID Manager configuration definition file (`horcm`*n*`.conf`). For example, when backing up from multiple primary volumes to multiple secondary volumes simultaneously, the same value needs to be set in `MU#` for each secondary volume in the RAID Manager configuration definition file (`horcm`*n*`.conf`).

Figure 4.4 shows an example of a configuration definition for backup or restore between storage subsystems.

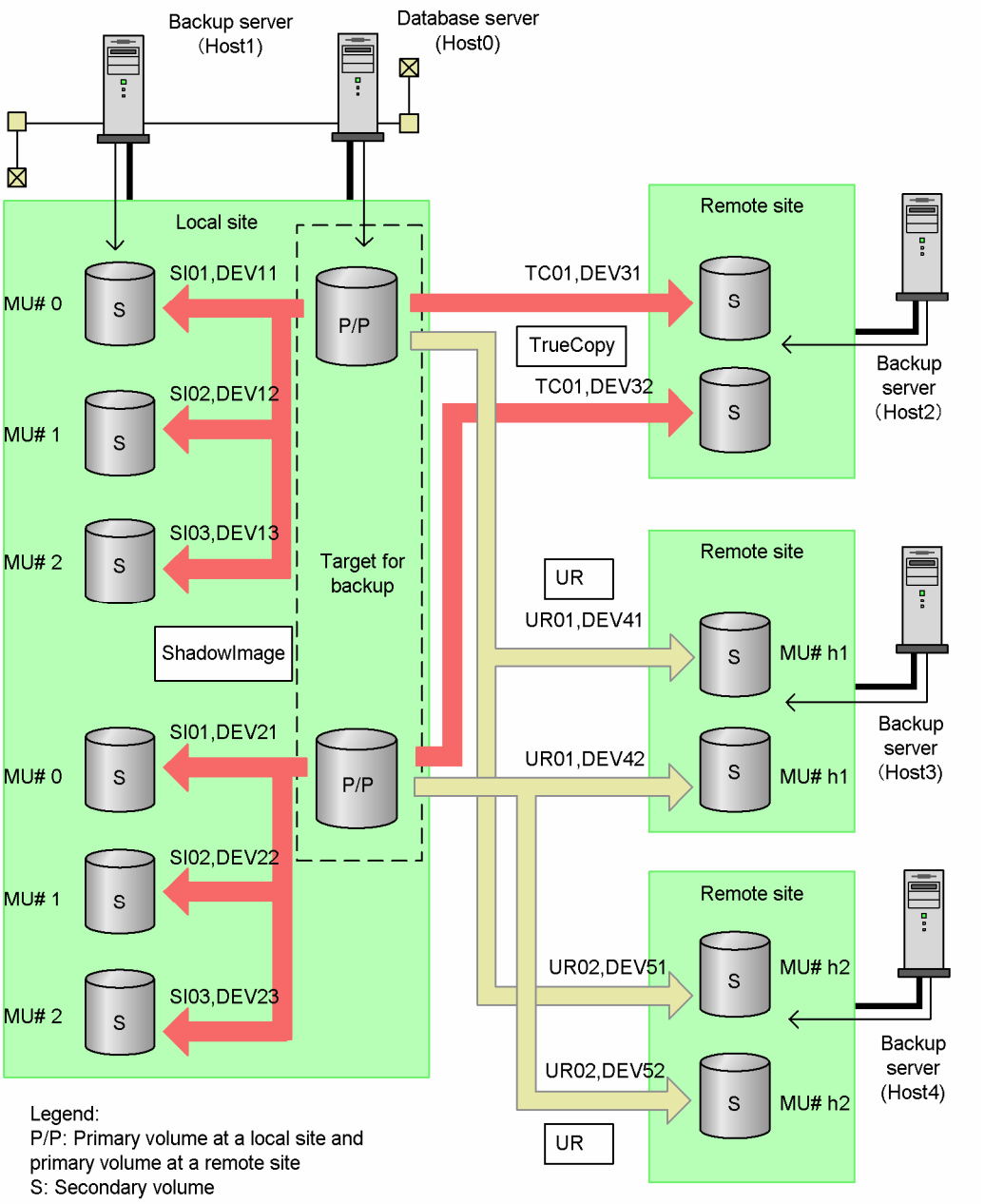


Figure 4.4 Example of a Configuration Definition for Backup or Restore Between Storage Subsystems

The following shows an example of the RAID Manager configuration definition file for a database server `Host0` in the local site in the above configuration example. In this example, multiple secondary volumes are allocated to one primary volume, and a separate configuration definition file is created for each backup target server.

RAID Manager configuration definition file (`horcm0.conf`)

```
HORCM_DEV
##dev_group       dev_name         port#    TargetID   LU#       MU#
SI01              DEV11            CL2-C    8          6         0
SI01              DEV21            CL2-C    8          7         0
SI02              DEV12            CL2-C    8          6         1
SI02              DEV22            CL2-C    8          7         1
SI03              DEV13            CL2-C    8          6         2
SI03              DEV23            CL2-C    8          7         2

HORCM_INST
##dev_group       ip_address             service
SI01              Host1                  12501/udp
SI02              Host1                  12501/udp
SI03              Host1                  12501/udp
```

RAID Manager configuration definition file (`horcm1.conf`)

```
HORCM_DEV
##dev_group       dev_name         port#    TargetID   LU#       MU#
TC01              DEV31            CL2-C    8          6
TC01              DEV32            CL2-C    8          7

HORCM_INST
##dev_group       ip_address             service
TC01              Host2                  12502/udp
```

RAID Manager configuration definition file (`horcm2.conf`)

```
HORCM_DEV
##dev_group       dev_name         port#    TargetID   LU#       MU#
UR01              DEV41            CL2-C    8          6         h1
UR01              DEV42            CL2-C    8          7         h1

HORCM_INST
##dev_group       ip_address             service
UR01              Host3                  12510/udp
```

RAID Manager configuration definition file (`horcm3.conf`)

```
HORCM_DEV
##dev_group       dev_name         port#    TargetID   LU#       MU#
UR02              DEV51            CL2-C    8          6         h2
UR02              DEV52            CL2-C    8          7         h2

HORCM_INST
##dev_group       ip_address             service
UR02              Host4                  12511/udp
```

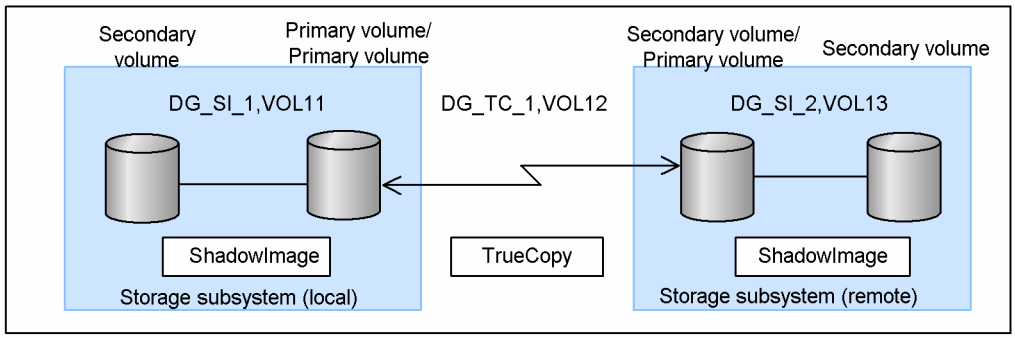
When setup has finished, execute either the `drmfsdisplay` command, the `drmsqldisplay` command, or the `drmexgdisplay` command with the `-refresh` option to register the resource information. This registers the remote site information set up by RAID Manager into Protection Manager. Also, after registering the resource information, execute either the `drmfsdisplay` command, the `drmsqldisplay` command, or the `drmexgdisplay` command with the `-cf` option to confirm that the remote site information is correctly registered into the volume.

4.2.5 Setting up a Multi-Target or Cascade Configuration

In a configuration where one device is paired with more than one volume, every configuration for that device must be defined in the same RAID Manager instance (that is, in the instance that has the same instance number). If the instance numbers differ, configuration information cannot be obtained properly, so backup will not be executable.

In a multi-target or cascade configuration for ShadowImage and TrueCopy, a primary volume in the local site and a primary volume in the remote site are paired up with more than one volume. You need to take this into consideration when specifying the RAID Manager configuration definition file. The following shows an example of coding in a configuration definition file for a multi-target configuration and cascade configuration.

Figure 4.5 shows the configuration definition file for a multi-target configuration and cascade configuration



The RAID Manager configuration definition file (horcm*n*.conf) for the primary and primary volumes in the local site in the multi-target configuration:

```
HORCM_DEV
#dev_group      dev_name
DG_SI_1         VOL11
DG_TC_1         VOL12
```

The RAID Manager configuration definition file (horcm*n*.conf) for the secondary and primary volumes in the remote site in the cascade configuration:

```
HORCM_DEV
#dev_group      dev_name
DG_SI_2         VOL13
DG_TC_1         VOL12
```

Figure 4.5 Configuration Definition for a Multi-Target and Cascade Configurations

4.3 Setting up Protection Manager

Protection Manager operation settings are specified in the Protection Manager configuration definition file (`init.conf`).

The `init.conf` file is stored in the following location:

`Protection-Manager-installation-folder``\conf\init.conf`

If the configuration definition file (`init.conf`) does not exist or contains an error, the commands provided by Protection Manager use default values to complete processing. If you accidentally delete the configuration definition file, you will need to reinstall the Protection Manager programs.

Retry count and interval values in the configuration definition file can be optimized to check the status of the following:

- Cluster resources
- Processes
- Command execution

4.3.1 Setting Up to Check Status of Cluster Resources

When Protection Products programs perform a cold backup or restoration in a cluster configuration, the programs place the cluster resources online or offline. The retry operation checks whether the cluster resource is placed online or offline. To change the retry count and interval for checking the status of the cluster resource, change the parameters listed in

Table 4.2. In a non-cluster environment, these parameters must not be changed.

Table 4.2 Parameters in init.conf

Parameter	Description	Default (specifiable range)
CLU_RETRY_TIME	▪ In this parameter, specify the maximum retry count for checking the status of cluster resources. Generally, specify a value other than 0. When the backup is of an Exchange database, the time to start the service depends on the roll forward processing time. For this reason, when the backup is of an Exchange database, use the default setting. ▪ If you specify 0, the Protection Manager program continues to retry checking until it can check the cluster resource status. For example, if the cluster resource has terminated with an error, Protection Manager will perform retries continuously. ▪ Generally, this parameter should be set to 60. However, if this value (60) is insufficient in a large-scale system configuration, this parameter should be twice the time required for cluster resource switching between online and offline mode in a typical operating environment. ▪ When changing the value of this parameter, make sure that the product of the *CLU_RETRY_TIME-value* and *CLU_RETRY_WAIT-value* is sufficiently larger than the time required to complete cluster resource online/offline switching.	0 (0 to 3600)
CLU_RETRY_WAIT	In this parameter, specify the number of seconds as the retry interval for checking the status of cluster resources.	1 (0 to 3600)

The time required for switching cluster resources online or offline depends on the cluster software used, the number of disks, the disk sizes, the time required to start or stop a service resource, and other factors. Review Table 4.3, and then specify an appropriate value.

Table 4.3 CLU_RETRY_WAIT Value for Each Type of Cluster Software

Cluster software to be used	Description	Recommended value (Units: seconds)
VERITAS Cluster Server	Checking the cluster resource status requires a wait period of about 1 to 2 seconds.	1 to 2
Microsoft Cluster Service (MSCS)	Checking the cluster resource status requires a wait period of about 1 to 2 seconds.	1 to 2

4.3.2 Setting Up to Check Status of a Process

When Protection Products programs perform a cold backup or restoration, the programs activate or deactivate the service of the DBMS product to be backed up or restored. When the Protection Manager programs activate or deactivate the service, the programs check the active/inactive status of the process. You can change the retry count and retry interval for checking the status of a process by using the parameters. In a cluster environment, you do not need to change this parameter because Protection Products do not reference this parameter.

Table 4.4 Process Checking Parameters in init.conf

Parameter	Description	Default (specifiable range)
SVC_RETRY_TIME	▪ In this parameter, specify the maximum retry count for checking the status of a process. Generally, specify a value other than 0. When the backup is of an Exchange database, the time to start the service depends on the roll forward processing time. For this reason, when the backup is of an Exchange database, use the default setting. ▪ If you specify 0, the Protection Manager program continues to retry checking until it can check the process status. For example, if the process has terminated with an error, Protection Manager will perform retries continuously. ▪ When changing the value of this parameter, make sure that the product of the *SVC_RETRY_TIME-value* and *SVC_RETRY_WAIT-value* is sufficiently larger than the time required for the completion of process startup or stopping.	0 (0 to 3600)
SVC_RETRY_WAIT	In this parameter, specify the number of seconds as the retry interval for checking the status of a process. Generally, you do not need to change the value of this parameter.	1 (0 to 3600)

4.3.3 Setting Up to Check Command Execution Retry Status

To change the retry count and interval for retries of checking the status of the command being executed by Protection Manager, change the parameters listed in Table 4.5.

Table 4.5 Retry Parameters in init.conf

Parameter	Description	Default (specifiable range)
COM_RETRY_TIME	▪ Specify the maximum retry count. Generally this value should be set to be other than 0. ▪ If you specify 0, Protection Products continue to retry checking until command status is confirmed. For example, if you enter two commands that cannot be executed at the same time, the command entered later will perform retries continuously until the command entered earlier finishes.	0 (0 to 3600)
COM_RETRY_WAIT	For this parameter, specify the same number of seconds as the retry interval for checking the command status.	10 (0 to 3600)

4.3.4 Setting up the Backup Option (SQL Server Database)

To shorten the time required for backup when the drmsqlbackup command is executed, you can skip the confirmation of the SQL Server database status. To shorten the time required for backup, change the parameter shown in

Table 4.6.

Table 4.6 Backup Option Parameter in init.conf

Parameter	Description	Default (specifiable range)
SQL_QUICK_BACKUP	Set this parameter to skip the confirmation of the SQL Server database status and thereby shorten the time required for backup when the drmsqlbackup command is executed. If 0 is specified, or this parameter is not specified, Protection Manager will check the database status. If you specify 1, Protection Manager will skip the confirmation of the database status. Generally, this parameter should be set to 0.	None (0 or 1)

In addition to changing the parameter, check the following conditions to shorten the time required for backup:

- Make sure that the database to be backed up is online. If the database to be backed up is not online, an error occurs at the time backup operation is performed on a SQL Server database.
- Make sure that the RAID Manager instance that manages the copy group to be operated is running. If the RAID Manager instance is not running, an error occurs when the RAID Manager command is issued.
- Make sure that the previous backup information has been deleted. Use the drmsqlcat command to confirm whether or not the backup information has been deleted. To delete the backup information, use the drmresync command.
- Make sure that the status of the copy group for which operations are to be performed is PVOL_PAIR. Use the RAID Manager pairdisplay command to confirm the status of the copy group.
- If you are using Protection Manager with a TagmaStore USP, TagmaStore NSC, Lightning 9900V, or Lightning 9900, make sure that the Quick Split option is on.

  If the Quick Split option of the storage subsystem is not on, making the SQL Server database static is canceled in the PVOL_COPY status. *Making a database static* means that only disk writing operations are temporarily stopped, without stopping SQL Server database updates. When making the database static is canceled, the backup data of the secondary volume might become corrupted.

4.3.5 Setting the Dictionary Map Synchronization Option (SQL Server Database)

If you specify that the dictionary map should be updated asynchronously, you can shorten the time required for backup command execution. If you want to shorten the time for backup command execution, specify the parameter shown in

Table 4.7.

Table 4.7 Dictionary Map Synchronization Parameter in init.conf

Parameter	Description	Default (specifiable range)
DRM_DB_SYNC	Specifies whether the dictionary map should be updated synchronously (YES) or asynchronously (NO). Generally, this parameter should be set to YES. If you need to specify NO (e.g., when the time required for backup must be reduced for a particular reason), perform adequate checking before starting operation.	YES (YES or NO)

Note: If you specify that the dictionary map file is to be updated asynchronously, you need to periodically back up the dictionary map file because the dictionary map file is susceptible to collapse if an unexpected server shutdown, or similar event, occurs during command execution.

4.3.6 Setting the Recovery Option (SQL Server Database)

You can specify whether to check the SQL Server database integrity when the drmsqlrecover command is executed, using the parameter listed in Table 4.8.

Table 4.8 Recovery Option Parameter in init.conf

Parameter	Description	Default (specifiable range)
SQL_CHECKDB	Specify YES if you want to check database integrity when the drmsqlrecover command is executed. If nothing or a value other than YES is specified, integrity will not be checked.	NO (YES or NO)

Even after execution of the drmsqlrecover command, you can check integrity from SQL Server Enterprise Manager.

If YES is specified for this option, integrity after recovery will be checked using SQL Server's DBCC CHECKDB. Note also that the DBCC CHECKDB execution time heavily depends on the database size, including the number of indexes in the database and the number of lines per table. For further information, see the SQL Server online help.

4.3.7 Setting up Restoration While Cluster Resources Are Online

In a cluster environment using Microsoft Cluster Service (MSCS), you can use the following parameter to specify whether to enable restoration while the cluster resources are in the online state.

Table 4.9 Setting up Restoration Parameter in init.conf

Parameter	Description	Default (specifiable range)
CLU_MSCS_RESTORE	Specify ONLINE if you want to perform restoration while the cluster resources are in the online state when the restore command is executed. If nothing or OFFLINE is specified, restoration will be performed while the cluster resources are in the offline state.	OFFLINE (ONLINE or OFFLINE)

- File system: You can perform restoration while the cluster resources are in the online state, for a volume within a cluster group that has a Microsoft Cluster Service environment. If you are using VERITAS Cluster Server (VCS) as the cluster software, restoration will be performed while the cluster resources are in the offline state.

  Table 4.10 shows the restoration type when the backup is of a file system.

Table 4.10 Cluster Type Parameters and Status in CLU_MSCS_RESTORE

Cluster type	CLU_MSCS_RESTORE	Restore target	Cluster resource status
MSCS	OFFLINE	Can be restored	OFFLINE
	ONLINE	Can be restored	ONLINE
VCS	Not applicable	Can be restored	OFFLINE

- SQL Server Database: You can perform restoration while the cluster resources are in the online state, for a user database in a Microsoft Cluster Service environment. If the restoration target contains a system database (master, model, msdb, or distribution), the SQL Server service is stopped momentarily. This means that restoration is performed while the cluster resources are in the offline state. If you are using VERITAS Cluster Server (VCS) as the cluster software, restoration will be performed while the cluster resources are in the offline state.

  Table 4.11 shows the cluster resource status when the backup is of a SQL Server database.

Table 4.11 SQL Server Database Cluster Resource Status Parameters

Cluster type	CLU_MSCS_RESTORE	Restoration target			Cluster resource status
		System database	User database	System + User database	
MSCS	OFFLINE	Can be restored	Can be restored	Can be restored	OFFLINE
	ONLINE	Can be restored	Can be restored	Can be restored	ONLINE[1]
VCS	Not applicable	Can be restored	Can be restored	Can be restored	OFFLINE

[1]If the database to be restored contains a system database, the ONLINE specification in the CLU_MSCS_RESTORE parameter is ignored and the programs place the cluster resource that manages the instance to be restored in the offline state.

Table 4.12 shows the cluster resource status when -no_resync is specified as the restoration option.

Table 4.12 SQL Server Database Cluster Resource Status Parameters (No Resync Option)

Cluster type	CLU_MSCS_RESTORE	Restoration target			Cluster resource status (cluster control)
		System database	User database	System + User database	
MSCS	OFFLINE	Can be restored	Can be restored	Can be restored	None[1]
	ONLINE	Can be restored	Can be restored	Can be restored	None[1]
VCS	Not applicable	Can be restored	Can be restored	Can be restored	None[1]

[1]If the database to be restored contains a system database, the programs place only the cluster resources of the SQL Server service in the offline state.

- Exchange database

  You can perform restoration while the cluster resources are in the online state, for all backup data obtained in the backup mode for a database that has a Microsoft Cluster Service environment. If you are using VERITAS Cluster Server as the cluster software, restoration will be performed while the cluster resources are in the offline state.

  The following table shows the cluster resource status when the backup is of an Exchange database.

Table 4.13 Cluster Resource Status Parameters for Exchange Database Backups

Cluster type	Backup mode	CLU_MSCS_RESTORE	Restoration target	Cluster resource status
MSCS	VSS	OFFLINE	Can be restored	OFFLINE
		ONLINE	Can be restored	ONLINE
	COLD or ONLINE	OFFLINE	Can be restored	OFFLINE
		ONLINE	Can be restored	ONLINE
VCS	VSS	Not applicable	Cannot be restored	Cannot be restored
	COLD or ONLINE	Not applicable	Can be restored	OFFLINE

4.4 Setting up Protection Manager Linkage with RAID Manager

Do not set HORCMINST or HORCC_MRCF in the system environment variable. Also, do not set HORCM_CONF or HORCMPERM in the system environment variable or the user environment variable.

For the RAID Manager settings, add or change the parameters in the RAID Manager-linkage definition file (DEFAULT.dat). However, do not delete or modify the following parameters since they are used internally by Protection Manager:

- VENDER
- PRODUCT

The RAID Manager-linkage definition file (DEFAULT.dat) is stored in the following location:

Protection-Manager-installation-folder\conf\raid

4.4.1 Specifying Instance Numbers

For the parameters HORCMINST and HORCMINST_AUX, specify the instance number of the RAID Manager instance. The specified instance number must be equal to the instance number used in the RAID Manager configuration definition file name (*n* in horcm*n*.conf) on the server.

Notes:

- When locating RAID Manager instances for the primary volume and secondary volume on the same server, for example, to utilize the volume replication function within the storage subsystem, do the following:
  - Specify both HORCMINST and HORCMINST_AUX.

    See Table 4.14 for instance numbers to be specified for HORCMINST and HORCMINST_AUX.
  - The values for HORCMINST and HORCMINST_AUX are reversed when specifying settings for either file servers and database servers, or for backup servers.
- When locating RAID Manager instances for the primary volume and secondary volume on separate servers, for example, to utilize the volume replication function between storage subsystems, do the following:
  - Specify only HORCMINST (not HORCMINST_AUX).

    See Table 4.14 for instance numbers to be specified for HORCMINST.
  - The RAID Manager instances specified in HORCMINST must be started on each server.
- If a RAID Manager instance which is not located on the server is specified in HORCMINST_AUX, an error message indicating failure to start the RAID Manager instance is output to the application log. If the instance is not associated with the subject of processing, processing of the Protection Manager command will continue despite this error message. Also, the result of the command will not be affected.

- It is recommended that you start the RAID Manager instances specified in HORCMINST and HORCMINST_AUX before executing a Protection Manager command.
- When specifying multiple instance numbers in HORCMINST and HORCMINST_AUX, write each instance number on a separate line, as shown below:

```
HORCMINST=0
HORCMINST_AUX=1
HORCMINST=2
HORCMINST_AUX=3
```

- In a cluster configuration, all servers making up the cluster must share the HORCMINST and HORCMINST_AUX instance numbers in DEFAULT.dat. You can create two identical DEFAULT.dat files, and then place one DEFAULT.dat file on one server and the other DEFAULT.dat file on the other server. Table 4.14 lists parameters in the RAID Manager-linkage definition file DEFAULT.dat file.

Table 4.14 Parameters to Specify an Instance Number in DEFAULT.DAT

Parameter	Description		Default
	File server or database server	Backup server	
HORCMINST	Specify the instance number of the instance that manages the primary volume.	Specify the instance number of the instance that manages the secondary volume.	None*
HORCMINST_AUX	Specify the instance number of the instance that manages the secondary volume.	Specify the instance number of the instance that manages the primary volume.	None

*The HORCMINST parameter is a required parameter. Do not omit this.

Figure 4.6 provides an example of specifying a RAID Manager instance number in the RAID Manager-Linkage definition file (`DEFAULT.dat`).

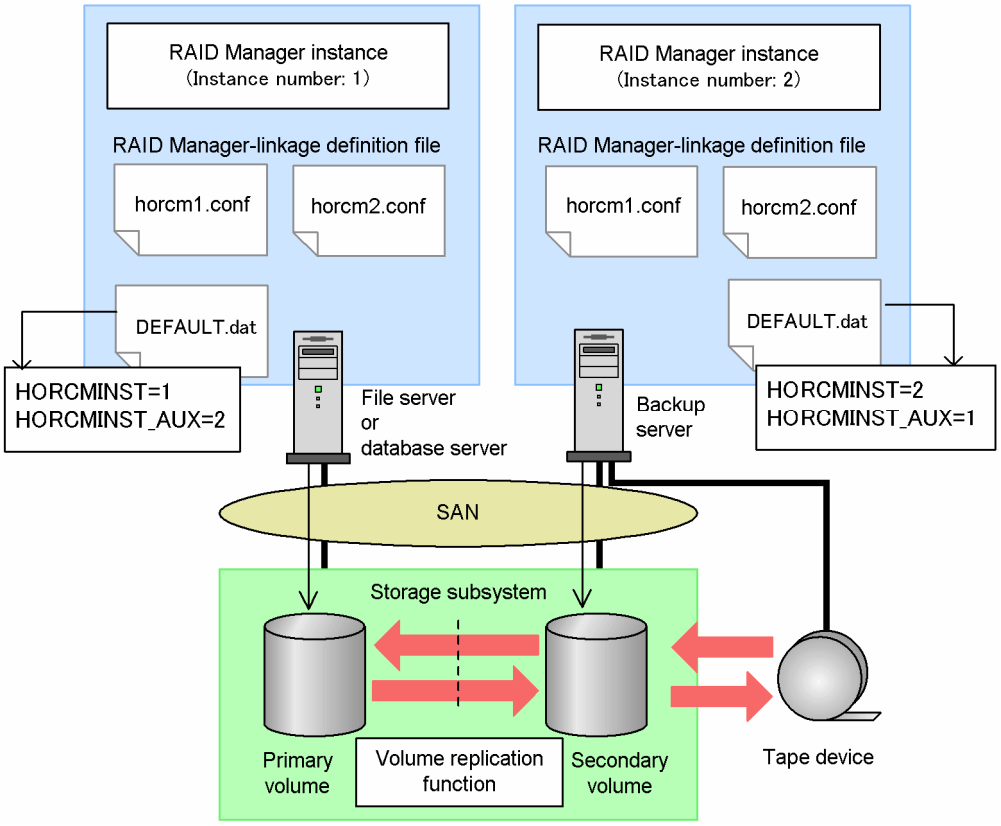


Figure 4.6 Specifying a RAID Manager Instance Number in the Linkage Definition File (DEFAULT.dat)

The following conditions are assumed for this example:

- A RAID Manager instance (instance number: `1`) has already been activated on the file/database server.
- A RAID Manager instance (instance number: `2`) has already been activated on the backup server.
- RAID Manager configuration definition files (`horcm1.conf` and `horcm2.conf`) have already been created on the file/database server and on the backup server.
- In this example, the parameters are specified as follows:
  - In the RAID Manager-linkage definition file (`DEFAULT.dat`) on the file or database server, `1` is assigned to the parameter `HORCMINST` as the instance number of the RAID Manager instance that manages the primary volume and `2` is assigned to the parameter `HORCMINST_AUX` as the instance number of the RAID Manager instance that manages the secondary volume.

– In the RAID Manager-linkage definition file (DEFAULT.dat) on the backup server, 2 is assigned to the parameter HORCMINST as the instance number of the RAID Manager instance that manages the secondary volume and 1 is assigned to the parameter HORCMINST_AUX as the instance number of the RAID Manager instance that manages the primary volume.

4.4.2 Setting up Retry Count and Interval for Checking Pair Status

Protection Manager programs determine whether a backup or restoration has finished by checking the status of the volume pair. By specifying parameters for the retry count and retry interval, you can set the retry count and retry interval for checking the pair status.

If you need to specify the status checking retry count and interval used for backup, restore, and resynchronization for each pair, you can specify them in the copy parameter definition file. For details on the copy parameter definition file, see section 4.4.6.

The pair status checking retry interval should be 1.5 times the estimated pair operation time in a backup system configuration. By default, a Protection Manager programs checks the pair status every five seconds, up to 120 times. This default setting assumes that a backup or restoration will finish in 10 minutes (5 seconds * 120 = 600 seconds). If backup or restoration does not finish within 10 minutes (because, for example, a volume is constantly being updated) the backup or restoration command will result in a timeout error. To prevent this, you must change the default setting. Table 4.15 lists parameters in the DEFAULT.dat file.

Table 4.15 Parameters in DEFAULT.DAT for Retry Count and Interval

Parameter	Description	Default (specifiable range)
RETRY_TIME	In this parameter, specify the maximum retry count for checking the status of a pair. If you specify 0, the Protection Manager program continues to retry checking until it can check the status.	120 (0 to 3600)
RETRY_WAIT	In this parameter, specify the same number of seconds as the retry interval for checking the status of a pair.	5 (0 to 3600)

The smaller the value of the RETRY_WAIT parameter, the larger the CPU load. When changing the RETRY_WAIT value, you must consider a number of conditions, including the storage subsystem to be used, the number of disks to be backed up or restored, the disk sizes, and the difference in data amounts between the primary and secondary volumes.

Table 4.16 provides information for information on setting the appropriate value.

Table 4.16 RETRY_WAIT Parameter Values for Each Storage Subsystem

Storage Subsystem	Description	Recommended value (Units: seconds)
TagmaStore™ USP TagmaStore™ NSC Lightning 9900™ V Series Lightning 9900	For a pair status to change, a wait time of at least one second is needed.	1
TagmaStore™ AMS Thunder 9500™ V Series Thunder 9200™ TagmaStore WMS Series	It takes some time for a pair status to change. A wait time of at least one minute is needed.	60

Depending on the purpose of the processing, you can set the retry count and retry interval for checking the pair status. If you want to set the retry count and retry interval for a specific purpose, add the appropriate parameter in the RAID Manager-linkage definition file DEFAULT.dat. If you do not specify any purpose-specific parameters, the values of RETRY_TIME and RETRY_WAIT are used. Table 4.17 shows the parameters you can add in DEFAULT.dat to specify the retry count and interval for checking the pair status according to the purpose.

Table 4.17 Parameters in DEFAULT.DAT to Specify the Retry Count and Pair Status Check Interval

Parameter	Description	Default (specifiable range)
RESYNC_RETRY_TIME	Specifies the maximum number of retries for checking the completion of pair resynchronization. Specify this parameter to check the pair status to make sure that backup processing or resynchronization finishes (the completion of paircreate[1] or pairresync). If you specify 0, the Protection Manager program continues to retry checking until it succeeds in checking the status. When adding this parameter, make sure that the product of the *RESYNC_RETRY_TIME-value* and *RESYNC_RETRY_WAIT-value* is sufficiently larger than the time required for the completion of pair creation.	Value of RETRY_TIME (from 0 to 3600)

Parameter	Description	Default (specifiable range)
RESYNC_RETRY_WAIT	Specifies the retry interval (in 10 milliseconds) for checking the completion of pair resynchronization. Specify this parameter to check the pair status to make sure that backup processing or resynchronization finishes (the completion of paircreate[1] or pairresync). When adding this parameter, you must consider the following: the number of disks to be backed up or resynchronized, the disk sizes, and the difference in data amounts between the primary and secondary volumes at the time the operation is performed[2]. Generally the required time of the resynchronization processing is larger than that of the splitting processing, specify RESYNC_RETRY_WAIT > SPLIT_RETRY_WAIT for the value.	Value of RETRY_WAIT (from 0 to 360000)
SPLIT_RETRY_TIME	Specifies the maximum number of times a retry can be performed for checking whether pair splitting has finished. Specify this parameter to check the pair status after pairsplit is executed for backup or restoration. If you specify 0, the Protection Manager program continues to retry checking until it succeeds in checking the status. When adding this parameter, make sure that the product of the *SPLIT_RETRY_TIME-value* and *SPLIT_RETRY_WAIT-value* is sufficiently larger than the time required for the completion of pair splitting.	Value of RETRY_TIME (from 0 to 3600)
SPLIT_RETRY_WAIT	Specifies the retry interval for checking whether pair splitting has finished. You can specify a value in units of 10 milliseconds. Specify this parameter to check the pair status after pairsplit is executed for backup or restoration. When adding this parameter, you must consider the following: the number of disks to be backed up or restored, the disk sizes, and the difference in data amount produced during I/O operations in online backup processing[2]. Generally the required time of the resynchronization processing is larger than that of the splitting processing, specify RESYNC_RETRY_WAIT > SPLIT_RETRY_WAIT for the value.	Value of RETRY_WAIT (from 0 to 360000)
RESTORE_RETRY_TIME	Specifies the maximum number of times a retry can be performed for checking whether resynchronization on restoration has finished. If you specify 0, the Protection Manager program continues to retry checking until it succeeds in checking the pair status. When adding this parameter, make sure that the product of the *RESTORE_RETRY_TIME-value* and *RESTORE_RETRY_WAIT-value* is sufficiently larger than the time required for the completion of resynchronization.	Value of RETRY_TIME (from 0 to 3600)
RESTORE_RETRY_WAIT	Specifies the retry interval for checking whether	Value of RETRY_WAIT

Parameter	Description	Default (specifiable range)
	resynchronization on restoration has finished. You can specify a value in units of 10 milliseconds. When adding this parameter, you must consider the following: the number of disks to be restored, the disk sizes, and the difference in data amount between the primary and secondary volumes when at the time the restoration is performed[2].	(from 0 to 360000)
BUSY_RETRY_WAIT	Specifies the retry interval for checking the pair status when the RAID Manager command is busy. You can specify a value in units of 10 milliseconds. Specify this parameter to check the pair status if the RAID Manager command was busy. When adding this parameter, determine the value considering how frequently a RAID Manager command busy factor (such as server overload or traffic concentration on the communication path) will occur and how much time will be required to eliminate the factor. Take adequate care because shortening the retry interval may further increase the RAID Manager load.	Value of RETRY_WAIT (from 0 to 360000)
RESTORE_DELAY_RETRY_TIME	If restoration is performed immediately after a backup, from a generation that is different from the primary volume backed up, some time may be required until the volume becomes ready for resynchronization. Specify the maximum number of retries that will be performed to confirm whether the volume is ready for resynchronization. You need to specify this parameter if you want to check whether the pair status allows execution of pairresync -restore for restoration. If you specify 0, the Protection Manager program continues to retry checking until it can check the pair status. When adding this parameter, make sure that the product of the *RESTORE_DELAY_RETRY_TIME-value* and *RESTORE_DELAY_RETRY_WAIT-value* is sufficiently larger than the time required until synchronization between the primary and secondary volumes is completed on the storage subsystem.	Value of RETRY_TIME (from 0 to 3600)
RESTORE_DELAY_RETRY_WAIT	Specify the interval between retries that will be performed to confirm whether the volume is ready for resynchronization. This value must be specified in 10-millisecond units. You need to specify this parameter if you want to check whether the pair status allows execution of pairresync -restore for restoration. When adding this parameter, you must consider the following: the number of disks to be restored, the disk sizes, and the difference in data amounts between the primary and secondary volumes at the time of the previous backup[2].	Value of RETRY_WAIT (from 0 to 360000)

[1]In the PAIR_CREATE parameter of the RAID Manager-linkage definition file (DEFAULT.dat), specify whether the Protection Manager programs are to automatically execute the paircreate command.

[2]The smaller the retry interval value (RESYNC_RETRY_WAIT, SPLIT_RETRY_WAIT, RESTORE_RETRY_WAIT, BUSY_RETRY_WAIT, or RESTORE_DELAY_RETRY_WAIT), the larger the CPU load. Refer to

Table 4.16 and specify a value appropriate to the storage subsystem to be used.

4.4.3 Specifying a Retry Count and Interval for RAID Mgr Command Busy Status

If a RAID Manager command returns a busy status, the Protection Manager program tries to execute the command again. If you want to change the RAID Manager command retry count and interval, add the parameters listed in Table 4.18.

Table 4.18 Parameters to Specify a Retry Count and Interval at Busy Status

Parameter	Description	Default (specifiable range)
BUSY_RETRY_TIME	Specify the maximum RAID Manager command busy status checking retry count. Specify this parameter to check the pair status if the RAID Manager command returns a busy status.	3 (from 1 to 3600)
BUSY_RETRY_WAIT	Specifies the retry interval for checking the pair status when the RAID Manager command returns a busy status. You can specify a value in units of 10 milliseconds. Specify this parameter to check the pair status if the RAID Manager command returns a busy status. When adding this parameter, you should determine the value after considering how frequently a RAID Manager command busy condition (such as server overload or traffic concentration on the communication path) will be encountered and how much time will be required to eliminate this condition. Take adequate care because decreasing the retry interval may further increase the RAID Manager load.	100 (from 1 to 360000)

4.4.4 Specifying a Retry Count and Interval Appropriate to the Operation Style

For operations such as backing up to a remote site, if you want to specify a different retry time (for retrying to confirm the pair status, which will be used in backup, restore, or resynchronization) at each command execution, you can use the copy parameter definition file. Set the retry time appropriate to the operation by specifying a copy parameter definition file name as an argument when executing the command.

When a retry time for pair status confirmation is specified in a copy parameter definition file, the value is given priority over the one in the RAID Manager-linkage definition file (DEFAULT.dat). The value in the RAID Manager-linkage definition file (DEFAULT.dat) is used when the value is not specified in the copy parameter definition file, or the value in the file is incorrect.

Create copy parameter definition files, using any valid file name, in the following location:

Protection-Manager-installation-directory\conf\raid*arbitrary-file-name*

The coding rules for the copy parameter definition file are:

- Parameter names are case sensitive.
- Write parameters in the form *parameter-name=parameter-value.*
- Use an equals character (=) between a parameter name and a parameter value.
- Specify a parameter value within the valid range.

The values listed in

Table 4.19 can be specified in copy parameter definition files.

Table 4.19 Parameters in Copy Parameter Definition Files

Parameter	Description	Specifiable range
RETRY_TIME	In this parameter, specify the maximum retry count for checking the status of a pair. If you specify 0, the Protection Manager program continues to retry checking until it can check the status.	0 to 3600
RETRY_WAIT	In this parameter, specify the same number of seconds as the retry interval for checking the status of a pair.	0 to 3600
RESYNC_RETRY_TIME	Specifies the maximum number of retries for checking the completion of pair resynchronization.	0 to 3600
RESYNC_RETRY_WAIT	Specifies the retry interval for checking the completion of pair resynchronization (in units of 10 milliseconds).	0 to 360000
SPLIT_RETRY_TIME	Specifies the maximum number of times a retry can be performed for checking whether pair splitting has finished.	0 to 3600
SPLIT_RETRY_WAIT	Specifies the retry interval for checking whether pair splitting has finished. You can specify a value in units of 10 milliseconds.	0 to 360000
RESTORE_RETRY_TIME	Specify the maximum number of times a retry can be performed for checking whether resynchronization on restoration has finished.	0 to 3600
RESTORE_RETRY_WAIT	Specifies the retry interval for checking whether resynchronization on restoration has finished. You can specify a value in units of 10 milliseconds.	0 to 360000
RESTORE_DELAY_RETRY_TIME	Specifies the retry interval for checking whether resynchronization on restoration has finished. If you specify 0, the Protection Manager program continues to retry checking until it can check the pair status.	0 to 3600
RESTORE_DELAY_RETRY_WAIT	Specify the interval between retries that will be performed to confirm whether the volume is ready for resynchronization. This value must be specified in units of 10 milliseconds. You need to specify this parameter if you want to check whether the pair status allows execution of pairresync -restore for restoration.	0 to 360000

For notes on specifying the values and calculation methods of setting values shown in

Table 4.19, see section 4.4.2.

4.4.5 Setting the Track Size for Copying Data

You can change the track size used for copying data that is to be resynchronized at backup or restoration. For example, you may want to change the track size for copying data in the following cases:

- When the priority of I/O for the primary volume is higher than the priority of data copying: Assign a smaller value to the `COPY_SIZE` parameter.
- When the priority of data copying is higher, in order to finish copying faster: Assign a larger value to the `COPY_SIZE` parameter.

If you are using Protection Manager with a TagmaStore USP, TagmaStore NSC, Lightning 9900V, or Lightning 9900, do not specify `15`. If you specify `15`, the highest priority is placed on the Lightning and Thunder internal copy processing; this may affect system operation.

If you are using Protection Products with a Thunder 9200, Thunder 9500V Series, or TagmaStore AMS/WMS Series product, you should consider the difference in data amounts between the primary and secondary volumes and the period of time during which operation can be stopped. If resynchronization from the primary volume to the secondary volume can finish during the period, specify `15`. Otherwise, specify `8` to ensure that backup will be performed after resynchronization from the primary volume to the secondary volume has finished.

For details about the values that can be specified as the track size, see the RAID Manager documentation. Table 4.20 shows the RAID Manager-linkage definition (`DEFAULT.dat`) file parameter for specifying track size.

Table 4.20 Specifying Track Size in DEFAULT.DAT

Parameter	Description	Default (specifiable range)
`COPY_SIZE`	Specifies the track size (the number of volume-type sectors per track) that can be copied in a batch for one copy unit when copying is performed from the primary to secondary volume or from the secondary to primary volume.	RAID Manager's default (1 to 15)

In addition to specifying the `COPY_SIZE` parameter, you can also specify the track size by using other methods. The methods are listed below. When multiple methods are used to specify the track size, the valid method is selected in the order shown:

a) The `-copy_size` option (used only by the `EX_DRM_RESYNC` command)

b) The `DRM_COPY_SIZE` environment variable

c) The `COPY_SIZE` parameter

d) The default value for RAID Manager

When you execute a command, you can use the `DRM_COPY_SIZE` environment variable to change the track size. The change is also applied to the `COPY_SIZE` parameter.

The track size can be changed when you execute the following commands:

Basic commands:

- `drmfsbackup`
- `drmsqlbackup`
- `drmexgbackup`
- `drmfsrestore`
- `drmsqlrestore`
- `drmexgrestore`
- `drmresync`

Extended commands:

- `EX_DRM_FS_BACKUP`
- `EX_DRM_SQL_BACKUP`
- `EX_DRM_EXG_BACKUP`
- `EX_DRM_FS_RESTORE`
- `EX_DRM_SQL_RESTORE`
- `EX_DRM_EXG_RESTORE`
- `EX_DRM_RESYNC`

4.4.6 Specifying Volume Pair Creation

You can specify whether the Protection Manager programs should automatically start the `paircreate` command to create a volume pair when the one of the following commands is executed in the pair volume status of `SMPL`:

- `drmfsbackup`
- `drmsqlbackup`
- `drmexgbackup`
- `drmresync`

Table 4.21 shows the parameter in the `DEFAULT.dat` file for automatic pair creation.

Table 4.21 Parameter in the RAID Manager-Linkage Definition DEFAULT.dat File for Automatic Pair Creation

Parameter	Description	Default (specifiable range)
`PAIR_CREATE`	Specify whether the Protection Products programs should automatically create a pair. If `DISABLE` is specified, the Protection Manager programs will not automatically create a pair. However, a pair is not automatically created when backing up to a secondary volume at a remote site even if `ENABLE` is specified. To automatically create a pair, you need to specify `ENABLE` before updating the dictionary map file.	`DISABLE` (`DISABLE` or `ENABLE`)

When you use Copy-on-write SnapShot in TagmaStore USP/NSC, and pair creation is executed automatically because `PAIR_CREATE=ENABLE` is specified in the RAID Manager-linkage definition file (`DEFAULT.dat`), there must be a pool area whose pool ID is `0`.

4.4.7 Settings for Secondary Volume Dynamic Recognition Functionality

If Protection Manager programs are configured so that backups are taken from the secondary volumes to a tape device, you can set the operating system to dynamically recognize the secondary volumes that connect to the backup server. Setting the operating system to dynamically recognize the secondary volumes prevents the secondary volumes from being accessed via an operation that is not performed using Protection Manager. This helps ensure more reliable backup and restore operations.

Table 4.22 lists the parameter for using the secondary volume dynamic recognition functionality.

Table 4.22 Parameter for Using the Secondary Volume Dynamic Recognition

Parameter	Description	Default (specifiable range)
DEVICE_DETACH	Specify this parameter if you want to make the operating system dynamically recognize volumes on the file/database server or backup server. If you specify DISABLE or omit the specification, dynamic recognition of volumes is disabled. If you specify ENABLE, volumes can be recognized dynamically.	DISABLE (DISABLE or ENABLE)

Note: Of the physical disks connected to the device adapter port, do not use the LUN#0 disk for the primary and secondary volumes if the secondary volume is to be recognized dynamically. If you use the LUN#0 physical disk, the system might not be able to recognize the other physical disks.

4.4.8 Setting up Operation Mode When Automatically Selecting Copy Groups

Usually, Protection Manager programs automatically select the secondary volume used as a backup destination. However, when the programs are copying data to multiple secondary volumes by locking the copy group, one generation might be given different generation numbers. To avoid restoring data with a generation number that does not always refer to the same generation, you can specify that a backup operation will cause an error if a generation number does not always refer to the same generation.

In a remote copy backup session, a generation number (MU#s) that does not always refer to the same generation causes an error regardless of this parameter definition.

Table 4.23 lists the parameter for the operation mode when automatically selecting copy groups.

Table 4.23 Parameter for Operation Mode When Automatically Selecting Copy Groups

Parameter	Description	Default (Specifiable range)
CONFIRM_GENERATION_IDENTICAL	Specify operations if, within a session, a generation number (MU#) does not always refer to the same generation due to automatic selection of the copy group during backup at a local site. Disagreement of pair identifiers causes an error when ENABLE is specified. When DISABLE is specified or this parameter is omitted, a warning message for a generation number that does not always refer to the same generation is output while the backup continues. However, it is possible that the restoration of this backup may fail.	ENABLE[1] (DISABLE or ENABLE)

[1] When the Protection Manager programs have been installed by overwriting the previous version (3.5 or earlier), the CONFIRM_GENERATION_IDENTICAL parameter is not specified in the prior RAID Manager-linkage definition file (DEFAULT.dat). So, the system assumes that the parameter is omitted or DISABLE is specified. To specify ENABLE for the parameter, add the parameter in the RAID Manager-linkage definition file (DEFAULT.dat).

4.4.9 Notes on Starting and Stopping RAID Manager

When a Protection Manager command is executed, a RAID Manager instance is used. The procedures used to start and stop a RAID Manager instance vary depending on whether the RAID Manager instance is active or inactive when the command is executed. Determine which method to use to start the RAID Manager instance, after taking into consideration your operation style.

If RAID Manager is inactive when a Protection Manager command is executed:

- When a command is executed, the command internally starts the RAID Manager instance.
- When the command terminates, the instance that was started by the command also stops.

If RAID Manager is active when a Protection Manager command is executed:

- Protection Manager uses the active RAID Manager instance. Even when the command terminates, the RAID Manager instance does not stop.
- When multiple commands are to be executed in a series, you should start RAID Manager beforehand to eliminate unnecessary repetitions of RAID Manager startup and stopping.

4.5 Specifying the Installation Path for RAID Manager

A parameter in the RAID Manager linkage definition file (DEFAULT.DAT) is used to specify the installation destination for the RAID Manager commands to be started by Protection Products.

If RAID Manager is to be installed on a different drive than Protection Products, specify the installation destination for RAID Manager.

If RAID Manager is to be installed on the same drive as Protection Products, the installation path for RAID Manager is set automatically when Protection Products is installed.

Table 4.24 Parameter for Specifying the Installation Path for RAID Manager

Parameter	Description	Default
INSTALLPATH	If RAID Manager is to be installed on a different drive than Protection Products, specify the installation destination for RAID Manager.	*system-drive*\HORCM

You can set up the system so that if an error occurs during backup command processing, the system modifies the status of the pair in the COPY or PAIR status to the PSUS status before command processing terminates. By specifying this setting, you can avoid the problem of degradation in online I/O performance that results when processing terminates because of an error during online (especially VSS) backup. This setting in the RAID Manager linkage definition file (DEFAULT.DAT) is effective in operations with the Thunder 9500V Series or the TagmaStore AMS/WMS Series.

Table 4.25 Specifying Modification of the Pair Status if an Error Occurs

Parameter	Description	Default
RECOVERY_MODE_ON_BACKUP_ABORTING	When the OS is Windows, specify whether the pair status is to be modified if an error# occurs during backup processing. If SQL Server is used and, in the Protection Manager configuration definition file (init.conf), 1 is set for SQL_QUICK_BACKUP, this parameter is ignored. If you specify ASIS, the pair status is not modified if a backup error occurs. If you specify PAIRSPLIT, the pair status is modified to PSUS if a backup error occurs.	ASIS

#

Indicates an error that occurs after the message KAVX5167-I is output (indicating that the setting for changing the pair status is valid) during backup processing. This setting is invalid if an error occurs before the KAVX5167-I message is output. In this case, the pair status is not modified even if PAIRSPLIT is specified.

4.6 Creating a Dictionary Map File

Protection Manager automatically creates a dictionary map file during installation. If you want to specify a directory and create a dictionary map file in it, or in a cluster configuration, create the dictionary map file using the procedure explained in section 4.6.1.

Note: Unnecessary dictionary map files may remain after the uninstallation of an old version of Protection Manager. If they do, remove the previous dictionary map and create a new dictionary map file.

4.6.1 Creating a Dictionary Map File in a Non-Cluster Configuration or on the Backup Server

In a non-cluster configuration environment or on the backup server, a dictionary map file is automatically created at the following location when the Protection Manager programs are installed:

Protection-Manager-installation-folder\DRM\db

If you use the dictionary map file automatically created during installation, you do not need to create a new dictionary map file.

To specify a directory and create a dictionary map file in it, create the dictionary map file by using the procedure explained below.

To create a dictionary map file in a user-specified directory:

1. Change the locations of the dictionary map file and backup catalog.

   Add the DRM_DB_PATH=*directory-name* entry to the Protection Manager configuration definition file (init.conf). When editing the files, note the following points:

   - Add the entry to the last line of the init.conf file.
   - After *directory-name*, always enter a linefeed code.
   - Specify a disk that is different from the volume to be backed up. Do not specify a disk that is configured as part of the volume pair to be backed up.

   The following shows an example entry: DRM_DB_PATH=H:\PTM

2. Create a dictionary map file and a backup catalog on the disk.

   Execute the drmdbsetup utility, as shown below.

   ```
   PROMPT>
   Protection-Manager-installation-directory\bin\util\drmdbsetup -i
   ```

   When the dictionary map file and backup catalog are created, the following message appears: All data files were created.

4.6.2 Creating a Dictionary Map File in an Active-Passive Cluster Configuration

In an active-passive cluster configuration, you need to create a dictionary map file in a shared directory that can be referenced from both the active and standby servers. Figure 4.7 provides an example of creating a dictionary map file in an active-passive cluster configuration.

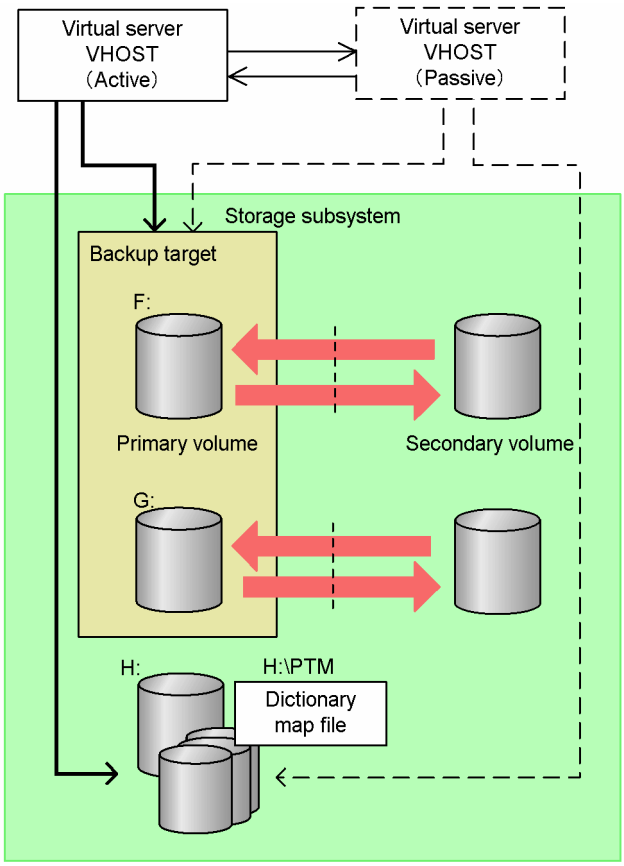


Figure 4.7 Example of Creating a Dictionary Map File in an Active-Passive Cluster Configuration

To create a dictionary map file in an active-passive cluster configuration:

1. Change the locations of the dictionary map file and backup catalog. In the `init.conf` configuration definition files of Protection Manager on the active and standby servers respectively, add the `DRM_DB_PATH=`*`directory-on-the-shared-disk;virtual-server-name`* entry. When editing the files, note the following points:
   - Add the entry to the last line of the `init.conf` file.
   - After *`virtual-server-name`*, always enter a linefeed code.
   - Specify a shared disk that is different from the volume to be backed up. Do not specify a shared disk that is configured as part of a pair volume that is to be backed up.
   - The extended commands automatically create a temporary directory during operations. The location of the temporary directory for extended commands is at the same level as the directory that stores the dictionary map file. To use the extended commands in a cluster configuration, the directory that stores the dictionary map file must be on a shared disk so that the extended commands can share the temporary directory.

   Note: If the OS is Windows and VERITAS Cluster Server is used in the cluster configuration, the virtual server name in `DRM_DB_PATH` must be the same as `VirtualName` of the Lanman resource that exists in the service group to be backed up.

   Note: If the backup target is an Exchange database, specify the name of the Exchange virtual server as the virtual server name in `DRM_DB_PATH`.

   The following shows an example entry in the Protection Manager configuration definition file (`init.conf`): `RM_DB_PATH=H:\PTM;VHOST`

2. Place the dictionary map file and backup catalog on the shared disk. Execute the `drmdbsetup` utility on the active server. You do not need to execute the utility on the standby server by switching the cluster. Before you execute the `drmdbsetup` utility, you must set the virtual server name in the `DRM_HOSTNAME` environment variable.

   Note: If you omit the virtual server name, the system creates a dictionary map file in the default dictionary map file storage directory instead of creating it on the shared disk, which invalidates the cluster settings by Protection Manager.

   If the backup target is an Exchange database, specify the name of the Exchange virtual server as the virtual server name.

   To execute the `drmdbsetup` utility, specify the following commands:

   ```
   PROMPT> set DRM_HOSTNAME=VHOST
   PROMPT>
   Protection-Manager-installation-directory\bin\util\drmdbsetup -i
   ```

   When the dictionary map file and backup catalog are created, the following message appears: `All data files were created.`

3. When the VERITAS Cluster Server is used as a cluster product, execute the `drmclusinit` command to register cluster resource information. For details, see section 4.7.2.

4.6.3 Creating a Dictionary Map File in an Active-Active Cluster Configuration

In an active-active cluster configuration, you need to create a dictionary map file on each virtual server. Note also, that the created dictionary map files must be placed in a shared directory that can be referenced from both servers, in case a failure should occur in either server. Figure 4.8 provides an example of creating a dictionary map file in an active-active cluster configuration.

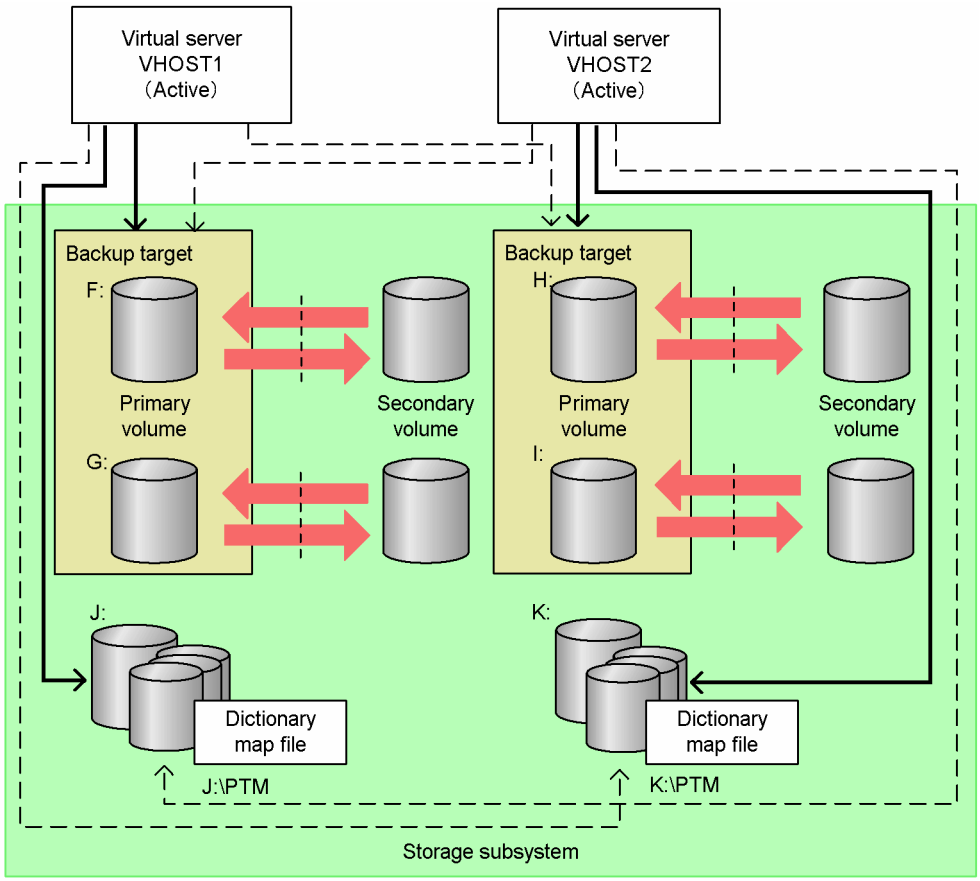


Figure 4.8 Example of Creating a Dictionary Map File in an Active-Active Cluster Configuration

To create a dictionary map file in an active-active cluster configuration:

1. Set the locations of the dictionary map file and backup catalog. In the `init.conf` configuration definition files of Protection Manager on the active and standby servers respectively, add a `DRM_DB_PATH=`*`directory-in-shared-disk;virtual-server-name`* entry for each virtual server. When editing the files, note the following points:
   - Add the entry to the last line of the `init.conf` file.
   - In *`directory-in-shared-disk`*, specify a cluster resource of *`virtual-server-name`*.
   - Specify a shared disk that is different from the volume to be backed up. Do not specify a shared disk that is configured as part of a pair volume that is to be backed up.
   - After *`directory-in-shared-disk;virtual-server-name`*, always enter a linefeed code.
   - The extended commands automatically create a temporary directory during operations. The location of the temporary directory for extended commands is in the same level as the directory that stores the dictionary map file. To use the extended commands in a cluster configuration, the directory that stores the dictionary map file must be on a shared disk so that the extended commands can share the temporary directory.

   Note: If the OS is Windows and VERITAS Cluster Server is used in the cluster configuration, the virtual server name in `DRM_DB_PATH` must be the same as `VirtualName` of the Lanman resource that exists in the service group to be backed up.

   Note: If the backup target is an Exchange database, specify the name of the Exchange virtual server as the virtual server name in `DRM_DB_PATH`. In the configuration definition file (`init.conf`), add `DRM_DB_PATH` once for each Exchange virtual server included in the cluster configuration.

   The following shows an example Protection Manager configuration definition file (`init.conf`):

   ```
   DRM_DB_PATH=J:\PTM;VHOST1

   DRM_DB_PATH=K:\PTM;VHOST2
   ```

2. Create the dictionary map file and backup catalog on each shared disk. At the two servers comprising a cluster configuration, place the virtual server into the `Active` status, and then execute the `drmdbsetup` utility.

   Note: Before you execute the `drmdbsetup` utility, you must set the virtual server name in the `DRM_HOSTNAME` environment variable. If you omit the virtual server name, the system creates a dictionary map file in the default dictionary map file storage directory instead of creating it on the shared disk, which invalidates the cluster settings by Protection Manager.

   If the backup target is an Exchange database, specify the name of the Exchange virtual server as the virtual server name.

To execute the `drmdbsetup` utility, specify the following commands:

```
PROMPT> set DRM_HOSTNAME=VHOST1

PROMPT>
Protection-Manager-installation-folder\bin\util\drmdbsetup -i

PROMPT> set DRM_HOSTNAME=VHOST2

PROMPT>
Protection-Manager-installation-folder\bin\util\drmdbsetup -i
```

When the dictionary map file and backup catalog are created, the following message appears: `All data files were created.`

3. When the VERITAS Cluster Server is used as a cluster product, execute the `drmclusinit` command to register cluster resource information. For details, see section 4.7.2.

4.7 Specifying Settings for a Cluster Configuration

This section describes settings required for a cluster configuration.

A cluster information configuration definition file needs to be created only when VERITAS Cluster Server is used as the cluster software. If you are using Microsoft Cluster Service, you do not need to create a cluster information configuration definition file.

4.7.1 When the File Server or Database Server Is in a Cluster Configuration

In a cluster configuration, you must create storage directories (for example, a dictionary map file storage directory) used by Protection Manager programs on a shared disk.

The following table lists and describes directories that need to be created on a shared disk and the location of explanations on how to create the directories.

Table 4.26 Directories that Must be Crated on a Shared Disk

Condition	Directory name	How to create
Shared disks used by Protection Manager programs (common to all objects subject to backup)	Dictionary map file storage directory	See section 4.5.
	Temporary Directory for Extended Commands	See section 4.15.9.
When backing up a SQL Server database	VDI metafile storage directory	Create the directory when the `drmsqlinit` command is executed. For details, see section 4.8.1.
	Transaction log backup file storage directory	
	UNDO log file storage directory	

To define in the cluster group the shared disks to be used by Protection Manager programs:

- Define a backup target and shared disks in the same cluster group. Normally, this method is used for a definition.

Section 4.7.1.1 explains definitions of each cluster group.

4.7.1.1 When a Backup Target and Shared Disks are in the Same Cluster Group

Define a backup target and shared disks used by Protection Manager programs in the same cluster group.

When the backup is of a file system, define shared disks in the same cluster group as the one monitoring disks to be backed up in the file system.

When the backup is of a SQL Server database, define shared disks in the same cluster group as the one containing instances to be backed up.

When the backup is of an Exchange database, define shared disks in the same cluster group as the one containing storage groups to be backed up.

Figure 4.9 shows the case where the backup target and shared disks are in the same group:

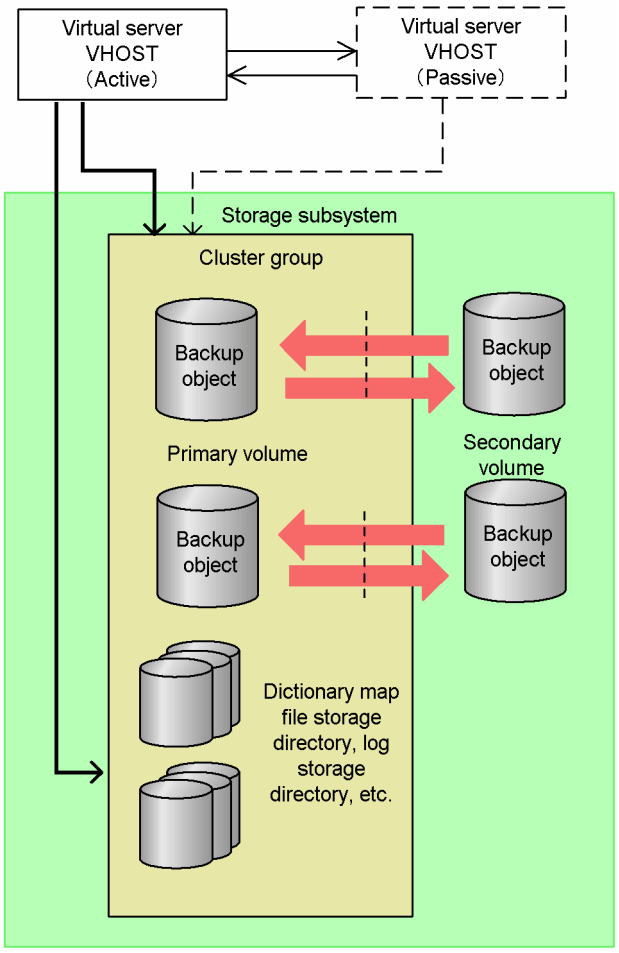


Figure 4.9 Cluster Group Definition When a Backup Target and Shared Disks are in the Same Group

4.7.2 Creating a Cluster Information Configuration Definition File for VERITAS Cluster Server

When you are using VERITAS Cluster Server, execute the `drmclusinit` command on the database or file server to register cluster resource information.

For details on the cluster resource information to be registered, see the description of the `drmclusinit` command in the *HiCommand® Protection Manager Command Reference* manual.

To register cluster resource information:

1. On the database or file server, execute the `drmclusinit` command with the cluster name specified in its argument. In this case, specify VCS as the cluster name because VERITAS Cluster Server is used. `PROMPT> drmclusinit VCS`
2. Register the items listed in Table 4.27 according to the output messages.

Table 4.27 Parameters for the Cluster Product

Entry Name	Explanation
Virtual server name (required)	Enter the virtual server name to be managed by the cluster product.
Resource group name (required)	Enter the service group name.
Name of the instance or application to be backed up (required)	▪ When the backup is of a file system, enter a hyphen (–) ▪ When the backup is of a SQL Server database, enter the SQL instance name ▪ When the backup is of an Exchange Server database, enter the storage group name
Resource type (required)	Enter `D`.
Resource name (required)	Enter the disk resource name. You do not need to specify the service resource. ▪ In a VxVM environment, enter the cluster resource names of the *mount-point* (logical volume)[1] and the *disk-group*[2].
Disk name (required if you have specified a resource type)	Enter the following information corresponding to the resource name: ▪ If you specified the *mount-point*, enter the mount point (e.g., `D:`). ▪ If you specified the *disk-group*, enter the disk group name (e.g., `DG01`).

[1] A resource whose resource type is `MountV`

[2] A resource whose resource type is `VMDg`

3. Confirm that the cluster information configuration definition file (`VCS.dat`) has been created. The cluster information configuration definition file is created in the following location:

```
Protection-Manager-installation-folder\conf\cluster\VCS.dat
```

4. Register the following items according to the output messages:

   To check your registration, execute the `drmclusinit` command with the `-v` option specified, on the database server or file server:

```
PROMPT> drmclusinit -v VCS
```

Notes:

- Execute the `drmclusinit` command before performing backup.
- If you have changed the cluster configuration, execute the `drmclusinit` command to re-specify the cluster information.
- Execute the `drmclusinit` command on both the active server and standby server.
- The virtual server name to be registered by using the `drmclusinit` command must satisfy the following conditions:

  - The specified virtual server name is the same as `VirtualName` in the Lanman resource of VERITAS Cluster Server.

  - The specified virtual server name is the same as that set in the `DRM_DB_PATH` parameter defined in the Protection Products configuration definition file (init.conf).
- The resource information to be registered by using the `drmclusinit` command must exactly match the information registered in the cluster products.
- The mount point to be registered in cluster products as the mount point for the mount point resource (MountV resource) must have the same format as the mount point registered in the dictionary map file.
- Only the following resources are registered by the `drmclusinit` command: Mount point resource (MountV resource) and disk group resource (VMDg resource) created on the physical disks that can be monitored by Protection Manager programs.
- Do not use the `drmclusinit` command to register the following resources:

  Service resource of a SQL Server or Exchange server, Lanman resource (virtual server), RegRep resource (registry replication resource)

  When a dictionary map has been placed on a shared disk and configured as a cluster resource in the same service group, mount point resource (MountV/Mount resource), disk group resource, or physical disk resource (VMDg/DiskRes resource)

4.7.2.1 Fixing Incorrect Registration Information

If the registered information contains an error such as one of the ones described below, Protection Products cannot execute backup or restore processing correctly because it cannot identify the target cluster resource:

- The virtual server name during backup or restoration does not match the virtual server name specified in the `DRM_HOSTNAME` environment variable during backup or restore execution.
- The correspondence between the virtual server name and the resource group name and resource name was registered incorrectly.
- The instance name was registered incorrectly.
- Some cluster resources have not been registered.
- The mount point or disk group name for the resource name was registered incorrectly.

To delete a parameter if a cluster product parameter has been registered incorrectly, execute the following procedure:

```
PROMPT> del <Protection Manager-installation-directory>\conf\cluster\DEFAULT.dat
PROMPT> del <Protection Manager-installation-directory>\conf\cluster\
<cluster-product-name>.dat
```

<*cluster-product-name*> is the argument value specified in the `drmclusinit` command.

4.8 Creating a Database Configuration Definition File

4.8.1 Creating a Database Configuration Definition File for SQL Server Database Backup

Use the `drmsqlinit` command to register the information about SQL Server. Register the following items:

- Storage directory for the VDI (Virtual Device Interface) metafiles
- VDI generation timeout period
- Storage directory for the UNDO log files
- Storage directory for the backup files (for transaction logs)

For details on the SQL Server database information to be registered, see the description of the `drmsqlinit` command in the *HiCommand® Protection Manager Command Reference* manual.

In an environment in which one system has multiple SQL instances, specify a separate directory for each instance when registering the metafile storage direction, the `UNDO` log file storage directory, or the backup file storage directory for transaction logs.

In a cluster configuration, specify a VDI (Virtual Device Interface) metafile storage directory, UNDO log file storage directory, and UNDO log file storage directory on shared disks that can be referenced by both the active server and standby server.

To register SQL Server information:

1. Execute the `drmsqlinit` command with the SQL Server instance name specified in its argument. To use SQL Server's default instance, specify `DEFAULT`.

   ```
   PROMPT> drmsqlinit instance-name
   ```

2. Register the following items according to the output messages:
   - Storage directory for the VDI (Virtual Device Interface) metafiles
   - VDI generation timeout period
   - Storage directory for the UNDO log files
   - Storage directory for the backup files (for transaction logs)

   If you do not specify a VDI metafile storage directory, VDI metafiles are stored in the same location as the SQL Server database data files. To facilitate management of VDI metafiles, we recommend that you do not specify a VDI metafile storage directory.

   You need to confirm that the VDI generation timeout period is appropriate by actually backing up the SQL Server database. In this stage (before operation checking), specify a value greater than your prediction (e.g., 1000 seconds). After operation checking is complete, measure the VDI freeze time and re-specify the timeout period using the `drmsqlinit` command.

For information on how to measure the VDI freeze time during operation checking, see the explanation below.

Both the UNDO log file storage directory and transaction log backup file directory are optional. However, if you do not specify an UNDO log file storage directory, you cannot restore or recover the SQL Server database in read-only mode. Also, if you do not specify a transaction log backup file storage directory, you cannot obtain transaction logs.

3. Confirm that the SQL Server database configuration definition file (*instance-name*.dat) has been created. The SQL Server database configuration definition file is created in the following location:

```
Protection-Manager-installation-folder\conf\MSSQL\instance-name.dat
```

4. Check the registration by executing the drmsqlinit command with the -v option specified:

```
PROMPT> drmsqlinit -v instance-name
```

Notes:

- If your system is in a cluster configuration. or if you use TrueCopy or UR functionality, you need to execute the drmsqlinit command on each server.
- Execute the drmsqlinit command after setting up the SQL Server instance. If you change SQL Server configuration information, such as the metafile storage directory or undo file storage directory, re-execute the drmsqlinit command. If you do not, the drmsqlbackup command will end with an error.
- In the following cases, VDI metafiles must be stored in the same location as SQL Server database data files. Therefore, do not specify a VDI metafile storage directory.
  - When your system is in a cascade configuration
  - When a SQL Server database is backed up between remote sites that use TrueCopy or UR functionality
  - When a SQL Server database is restored to a different SQL instance from that used at backup
- The value for the VDI generation timeout period specified by using the drmsqlinit command should be the duration necessary for the backup or restore process of the SQL Server database.

  If you specify an unsuitable timeout value and backup takes a long time, one of the following problems might occur. Specify a suitable value that allows enough time for both operation and backup.

  - If the timeout value is too low

    The command ends with an error due to a timeout and the database cannot be backed up.

  - If the timeout value is too high

    The client application connected to the SQL Server reaches a timeout before the command ends with an error due to a timeout.

    1. Use the drmsqlinit command to set the number of seconds for the VDI creation timeout to a large value (such as 1000).
    2. Execute the drmsqlbackup command.

3. Check the interval between the following two log messages by viewing the Protection Products log file `drm_output.log`.

   ```
   KAVX1371-I Processing to freeze databases will now start.
   KAVX1372-I Processing to thaw databases will now end.
   ```

4. Re-execute the `drmsqlinit` command to set up the VDI generation timeout value, based on the time obtained in step 3 above. Because the execution time of the `drmsqlbackup` command is affected by the status of system resources, set a timeout value larger than the time obtained in step 3.

4.9 Specifying Settings Related to Linkage with a SQL Server

This section describes the following two types of settings:

- Specifying the SQL Server login timeout option

  Specify this option if the command that operates the SQL Server database terminates with an error due to a SQL Server login request timeout (error message: `KAVX1008-E`, detailed message `DRM-11013`, Code: `0`).

  You can use the `SQL_LOGIN_TIMEOUT` parameter to specify the SQL Server login timeout option in `sqlinit_`*`instance-name`*`.conf`.

  A login request timeout is likely to occur in environments with high network loads, which may result from a large-scale database.

- Specifying the SQL Server automatic recovery time

  Specify this option if the command terminates with an error due to the automatic recovery processing (at SQL Server startup) not being complete during restoration of the SQL Server database that contains the system databases (master, model, and msdb) (error message: `KAVX1008-E`, detailed message `DRM-11011`, code: `5180`).

  You can use the `SQL_AUTORECOVERY_TIME` parameter to specify the SQL Server automatic recovery time in `sqlinit_`*`instance-name`*`.conf`.

  Table 4.28 lists whether the parameter can be specified for each combination of restoration-target databases.

Table 4.28 Relationship Between SQL_AUTORECOVERY_TIME Parameter and Restored Objects

Restored object		SQL Server automatic recovery processing		Parameter specification
System database	User database	Non-cluster	Cluster	
Yes	Not applicable	Performed	Performed	Needed[1]
Yes	Can be restored (all)	Performed	Performed	Not needed[2]
Yes	Can be restored (partial)	Performed	Performed	Needed[1]
Not applicable	Can be restored (all)	Not performed	Performed	Not needed[2]

Restored object		SQL Server automatic recovery processing		Parameter specification
System database	User database	Non-cluster	Cluster	
Not applicable	Can be restored (partial)	Not performed	Performed	Not needed[3]

[1] If the command terminates with an error because the automatic recovery processing at SQL Server startup is not complete during restoration, specify the SQL_AUTORECOVERY_TIME parameter.

[2] You do not need to specify the SQL_AUTORECOVERY_TIME parameter because all user databases are detached during automatic recovery processing.

[3] If it takes time to automatically recover the databases that are not subject to restoration, restoration processing for restoration target databases might not be affected because the system database is not subject to restoration. Therefore, you do not need to specify the SQL_AUTORECOVERY_TIME parameter.

To specify the login timeout option or automatic recovery time:

1. Copy `sqlinit_`*`instance-name`*`.conf`.

- The copy source is:

  `Protection-Manager-installation-folder``/conf/MSSQL/sample/sqlinit.conf`

- The copy destination is:

  `Protection-Manager-installation-folder``/conf/MSSQL/sqlinit_`*`target-instance-name`*`.conf`

2. Specify the parameter in the `sqlinit_`instance-name`.conf` file.

   Write the parameter starting from the far left, without spaces. Delimit the parameter name and timeout value by an equal sign (=), and do not enter space or tab characters before or after the equal sign. Lines beginning with # are assumed to be comment lines. If a parameter is specified two or more times, the earliest value written is effective.

   – To specify the SQL Server login timeout option

     Add the `SQL_LOGIN_TIMEOUT=`***login-timeout-value***.

   – To specify the SQL Server automatic recovery time

     Add the `SQL_AUTORECOVERY_TIME=`***SQL-Server-automatic-recovery-time***.

   Table 4.29 shows the parameter defined in the sqlinit_instance-name.conf file.

Table 4.29 Parameter Defined in the sqlinit_instance-name.conf File

Parameter	Description	Default
SQL_LOGIN_TIMEOUT	In this parameter, specify the same number of seconds as the waiting period for completion of a request to log in to a SQL Server. If this parameter is specified, a login to SQL Server uses the value is specified in the SQL_ATTR_LOGIN_TIMEOUT attribute of the ODBC function SQLSetConnectAttr. If the parameter is not specified, the SQL_ATTR_LOGIN_TIMEOUT attribute is not specified.	None

Note: The value specified for SQL_LOGIN_TIMEOUT must be smaller than the VDI generation timeout period specified for the drmsqlinit command. If you specify a value greater than the VDI generation timeout period, the drmsqlbackup command may encounter a VDI generation timeout when waiting for the processing of the request for login to SQL Server and may terminate with an error.

Table 4.30 shows the parameter defined in the sqlinit_instance-name.conf file.

Table 4.30 Parameter Defined in the sqlinit_instance-name.conf File

Parameter	Description	Default
SQL_AUTORECOVERY_TIME	In this parameter, specify the SQL Server automatic recovery time (in seconds) for the restoration command. If this parameter is set to a value other than 0, the system waits for completion of SQL Server automatic recovery processing for the specified period of time. When you determine a value for SQL_ATTR_LOGIN_TIMEOUT, add enough time so that the value becomes about twice as much as the time from the beginning to the end of automatic recovery of the master database. You can use the SQL Server log file to check the amount of time from the beginning to the end of automatic recovery of the master database.	0

4.10 Specifying Settings for Using VSS

If you want to use VSS to perform an online backup of a file system or an Exchange database, the following settings are required:

- When the backup is of an Exchange database, set up the Protection Manager and RAID Manager environments so that Protection Manager for Exchange can perform cold backups. Moreover, it is necessary to create a pair before backup. When creating a pair, specify the `-m noread` option in the `paircreate` command.
- RM Shadow Copy Provider must be installed in advance on the file server, or both the database server and the backup server. For information about how to install RM Shadow Copy Provider, see the RAID Manager manual.

  Before you install RM Shadow Copy Provider, use **Computer Management** and check that the Windows services are set to the default settings shown in the following table:

Table 4.31 Default Settings for Windows services related to RM Shadow Copy Provider

Service name	Displayed name	Startup type
RpcSs	**Remote Procedure Call (RPC)**	**Automatic**
EventLog	**Event Log**	**Automatic**
DcomLaunch	**DCOM Server Process Launcher**	**Automatic**
SamSs	**Security Accounts Manager**	**Automatic**
Winmgmt	**Windows Management Instrumentation**	**Automatic**
EventSystem	**COM+ Event System**	**Manual**
MSIServer	**Windows Installer**	**Manual**
VSS	**Volume Shadow Copy**	**Manual**
COMSysApp	**COM+ System Application**	**Manual**
MSDTC	**Distributed Transaction Coordinator**	**Manual**

  If you attempt to install RM Shadow Copy Provider when these Windows services are not set to their default settings, the RM Shadow Copy Provider service will not be registered as a Windows service. In such a case, when the backup command is executed when VSS is specified in the `mode` option, an error message will be displayed and processing will terminate with an error.

- The execution of the `raidvchkset` command, described in the RAID Manager manual, is not necessary.
- Edit the VSS definition file on the file server or database server. The VSS definition file (`vsscom.conf`) is stored in the following location. Do not add or delete parameters:

  Protection-Manager-installation-directory\conf

Table 4.32 lists `vsscom.conf` parameters.

Table 4.32 vsscom.conf Parameters

Parameter	Description	Specifiable Values
`BACKUP_SERVER`	Specifies the host name or IP address of the backup server.	1 to 256 one-byte characters
`WRITER_TIMEOUT_RETRY_COUNT`	Specifies the number of times backup is retried if a Writer timeout occurs during backup. If 0 is specified, backup is not retried. The retry count should be increased in the following cases: ▪ Backup operation takes a long time (a large number of disks are backed up at the same time or when ShadowImage is used with a Thunder 9500V or TagmaStore AMS/WMS Series product). ▪ The error message `KAVX5004-E` appears during backup. ▪ For details on the disk configurations in which backup requires a long time for a file system, see section 2.12. For an Exchange database, see section 2.13.2. If file system backups are taking a long time, check the disk configuration details in 2.13.1. If Exchange database backups are taking a long time, check the disk configuration details in 2.13.2.	0 to 99999
`WRITER_TIMEOUT_RETRY_INTERVAL`	Specifies, in seconds, the intervals at which backup is retried when a Writer timeout occurs. Usually, this parameter does not need to be changed.	0 to 99999
`EXG_VERIFY_RETRY_COUNT`	Specifies the number of times backup is retried when a verification error occurs in the Exchange database containing the backup results. If 0 is specified, backup is not retried. Usually, this parameter does not need to be changed. This value is not required when the backup is of a file system.	0 to 99999
`EXG_VERIFY_RETRY_INTERVAL`	Specifies, in seconds, the intervals at which backup is retried when an Exchange database verification error occurs. Usually, this parameter does not need to be changed. This value is not required when the backup is of a file system.	0 to 99999

- When the backup is of an Exchange database, install Exchange Management Console on the backup server. If Exchange Server 2003 is used, also install the `ESEUTIL` command.

  Installing Exchange Management Console

  Install Exchange Management Console on the backup server.

The version of Microsoft Exchange Server to be installed must match that of the Microsoft Exchange Server installed on the database server. If a service pack has been applied to the Microsoft Exchange Server installed on the database server, the service pack must also be applied to the Microsoft Exchange Server installed on the backup server. For details on how to install Exchange Management Console, see the documentation for Exchange Server 2003 or Exchange Server 2007.

Installing the `ESEUTIL` command

After installing Exchange Management Console, on the backup server, copy the following files from *Exchange-Server- install-destination*\bin to *Protection-Products-installation-directory*\bin\util:

-ESE.dll

-ESEUTIL.exe

-EXCHMEM.dll

-EXOSAL.dll

4.10.1 Using VSS When RM Shadow Copy Provider Version is 01-02-03/03 or Later

This section describes the system settings on each server when the version of RM Shadow Copy Provider is 01-02-03/03 or later.

Note that you do not need to change the settings if you have upgraded the version of RM Shadow Copy Provider to 01-02-03/03 or later, and perform the following operations:

- A backup to a single generation
- A backup within a storage subsystem (local backup)
- Set the system environment variables for the file server or database server, as shown in Table 4.33.

Table 4.33 System Environment Variables Required to Use VSS (File Server or Database Server)

Environment variable	Description
VSHTCRMENVF	Specify the following value: *Protection-Products-installation-destination*\conf\vssprv.conf[1]

[1]This file is for internal use of Protection Manager programs. You do not need to specify this file.

- Set system environment variables for the backup server, as shown in Table 4.34.

Table 4.34 System Environment Variables Required to Use VSS (Backup Server)

System Environment Variable	Description
VSHTCHORCMINST_REMOTE	Specifies the instance number of the RAID Manager that manages the secondary volume used in VSS backup. The specified value must be specified in the HORCMINST parameter of the RAID Manager-linkage definition file (DEFAULT.dat).

- The execution of the raidvchkset command, described in the RAID Manager manual, is not necessary.
- After setting the above system environment variables, restart the operating system.

4.10.2 Using VSS When RM Shadow Copy Provider Version is Earlier Than 01-02-03/03

If you use a version of RM Shadow Copy Provider that is earlier than 01-02-03/03 and perform backup operations by using VSS, only the following operations are available:

- A backup to a single generation
- A backup within a storage subsystem (local backup)

If you want to perform the following operations, upgrade the version of RM Shadow Copy Provider to 01-02-03/03 or later, and then set up each server by referring to 4.10.1:

- A backup to multiple generations
- A backup between storage subsystems (remote copy)
- A backup with a name identifying the generation specified
- Set the system environment variables for the file server or database server, as shown in Table 4.35.

Table 4.35 System Environment Variables Required to Use VSS (File Server or Database Server)

System Environment	Variable Description
VSHTCHORCMINST_LOCAL	Specifies the instance number of the RAID Manager that manages the primary volume used in VSS backup. The specified value must be specified in the HORCMINST parameter of the RAID Manager-linkage definition file (DEFAULT.dat).
VSHTCHOMRCF_MUN	Specifies the MU# (pair identifier) of the copy group that uses VSS for backup.

- Set system environment variables for the backup server, as shown in

Table 4.36.

Table 4.36 System Environment Variables Required to Use VSS (Backup Server)

System Environment Variable	Description
VSHTCHORCMINST_REMOTE	Specifies the instance number of the RAID Manager that manages the secondary volume used in VSS backup. The specified value must be specified in the HORCMINST parameter of the RAID Manager-linkage definition file (DEFAULT.dat).

- After setting the above system environment variables, restart the operating system.
- When you use VSS in a cluster environment, define the primary volume to be used in all virtual servers in a single RAID Manager instance, and use the same instance number on all physical nodes.

4.11 Setting up Linkage with a Backup Management Product

This section describes how to set up linkage with a backup management product. You need to register parameters and create a configuration definition file.

Before registering parameters and creating configuration definition files, you must set up each backup management product. Perform the following procedures for the backup management product:

- For NetBackup:
  - Create as many CLASS or POLICY entries as the number of backup jobs to be performed in parallel.
  - Specify a SCHEDULE value for each of the created CLASS or POLICY entries.
  - Set the retention period of the media.
- For Backup Exec:
  - Specify the media, device, and logon information.
  - Set the retention period of the media.

4.11.1 Registering Information for Linkage with a Backup Management Product

Use the drmtapeinit command to register the information for linkage with a backup management product. Register the following items according to the output messages:

- Name of the backup management product
- Number of days during which the backup catalog used for tape backup will be stored

The registered information is stored as a file in the following location:

Protection-Manager-installation-directory\conf\tape\DEFAULT.dat

To register information for linkage with a backup management product:

1. Execute the drmtapeinit command.

```
PROMPT> drmtapeinit
```

2. When the following message appears, enter the name of the backup management product.

```
KAVX0411-I Enter the backup management product name.
```

   Enter one of the following values:

   - If you are using NetBackup: `NBU`
   - If you are using Backup Exec: `BEWS`

3. When the following message appears, enter the number of backup catalog storage days.

```
KAVX0417-I Enter the number of days to store the backup catalog.
```

   For details on the value to be specified, see section 4.11.1.1.

4. Execute the `drmtapeinit` command with the `-v` option specified, in order to confirm that the information has been registered correctly.

   For example:

```
PROMPT> drmtapeinit -v
PROMPT> backup-product-name: NBU
PROMPT> number-of-backup-catalog-storage-days: 10
```

4.11.1.1 Notes on Specifying the Number of Backup Catalog Storage Days

- For the `drmtapeinit` command, the number of days for tape backup catalog storage must be specified so that the following relationship holds:
  - The media protection period specified for the backup management product is equal to or greater than the number of backup catalog storage days specified for the `drmtapeinit` command.
  - If the number of backup catalog storage days is longer than the media protection period specified for the backup management product, restoration becomes impossible when the media protection period expires, even if the backup catalog exists.
- Note that if you specify `0`, the backup catalog will be stored infinitely.
- After the storage days of a backup catalog elapse, the backup catalog will be deleted when one of the following commands is executed: `drmmount`, `drmtapebackup`, `drmtaperestore`, or `drmtapecat` (unless the `-o` or `-delete` option is specified).
- When a file that contains the information registered by the `drmtapeinit` command for linkage with a backup management product becomes unnecessary, or when the file changes, delete the file.

4.11.2 Creating a Configuration Definition File for Tape Backup

Depending on the backup management product you use, rename the provided sample configuration definition file for tape backup as indicated below:

- NetBackup: `NBU.dat`
- Backup Exec: `BEWS.dat`

The sample definition file for tape backup is stored in the following location:

- NetBackup

  Protection-Manager-installation-directory\conf\tape\NBU_sample.dat

- Backup Exec

  Protection-Manager-installation-directory\conf\tape\
  BEWS_sample.dat

4.11.2.1 Configuration Parameters

The parameters in the configuration definition file for tape backup occur in the order indicated in Table 4.37.

Table 4.37 Parameters Defined in the Configuration Definition File for Tape Backup (For NetBackup)

Parameter	Description
INST_PATH	Specify the absolute path which NetBackup is installed (use a maximum of 1,024 bytes).
MOUNT_POINT	Specify the absolute path of the mount point of the volume to be backed up (use a maximum of 255 bytes). If no entry for the mount point to be backed up can be found, values specified for CLASS or POLICY and for SCHEDULE under "MOUNT_POINT=default" will be used. Specify the mount point in the tape backup configuration definition file in the order in which Protection Products automatically mount them during a tape backup. Protection Manager mounts the drives according to the following rules: ▪ If the mount point is to be passed in an argument for tape backup or restoration, the system selects the unoccupied drives from the specified drives and mounts them in alphabetical order. ▪ If the mount point is not specified for tape backup or restoration, the system selects the unoccupied drives from all unoccupied drives and mounts them in alphabetical order. Select either of the rules described above in accordance with the actual operation method for tape backup or restoration, check the unoccupied drives using Windows Explorer or other tools, and specify the unoccupied drives in alphabetical order. When the backup is of a SQL Server database, in addition to the mount point that contains the database to be backed up, you must specify the volume that contains the VDI metafile. For this specification, enter the absolute path of the directory containing the VDI metafile or write default. The VDI metafile storage directory must be specified at the end of MOUNT_POINT and in front of default.
CLASS or POLICY	Specify the CLASS or POLICY name to be used when backing up data that is stored in the path specified for MOUNT_POINT (using 255 bytes or less). You can use either POLICY or CLASS. However, you cannot specify both POLICY and CLASS for the same MOUNT_ POINT.
SCHEDULE	Specify the schedule name used to back up the data stored in the path specified by MOUNT_POINT (use a maximum of 255 bytes).
PARALLEL_COUNT	Specify this when executing the following in parallel: backing up of data to tape, and restoring data from tape. Specify the number of backup processes to be executed in parallel: from 1 to 999. If you specify a non-numeric character, or a number outside this range, execution will be performed sequentially, and not in parallel. Before you can perform backup jobs in parallel, you must create and configure as many CLASS or POLICY entries as the number of parallel backup jobs in NetBackup.

The parameters contained in the configuration files for VERITAS Backup Exec occur in the order shown in Table 4.38

Table 4.38 Parameters Defined in the Configuration File for Tape Backup (for Backup Exec)

Parameter	Description
INST_PATH	Specify the absolute path in which Backup Exec is installed (use a maximum of 1,024 bytes).
MOUNT_POINT	Specify the absolute path of the mount point of the volume to be backed up (use a maximum of 255 bytes). If no entry for the mount point to be backed up can be found, a value specified for SCRIPT_FILE under "MOUNT_POINT=default" will be used. If the value of this parameter is set to default, the value specified by SCRIPT_FILE will be used for all directories. Specify the mount point to be mounted during a tape backup. Specify the mount point in the tape backup configuration definition file in the order in which Protection Products automatically mounts them during a tape backup. Protection Manager mounts the drives according to the following rules: ▪ If the mount point is to be passed in an argument for tape backup and restoration, the system selects the unoccupied drives from the specified drives and mounts them in alphabetical order. ▪ If the mount point is not specified for tape backup or restoration, the system selects the unoccupied drives from all unoccupied drives and mounts them in alphabetical order. Select either of the rules described above in accordance with the actual operation method for tape backup or restoration, check the unoccupied drives using Windows Explorer or other tools, and specify the unoccupied drives in alphabetical order. When the backup is of a SQL Server Database, In addition to the mount point that contains the database to be backed up, you must specify the volume that contains the VDI metafile. For this specification, enter the absolute path of the directory containing the VDI metafile or write default. The VDI metafile storage directory must be specified at the end of MOUNT_POINT and in front of default.
SCRIPT_FILE	Specify the name of the backup job script file to be used to back up the data stored in the path specified by MOUNT_POINT (use a maximum of 1,024 bytes).[1]

[1] For details about backup job script files, see the Backup Exec documentation.

A sample of a backup job script file (bjscript.txt) is included in the installation CD-ROM for Backup Exec.

The following shows an example of the Backup Exec backup job script file used by Protection Manager programs:

```
[General]
JOB_TYPE=Backup
JOB_NAME=tapebackup
[Backup_Options]
OVERWRITE_JOB=YES
[Media_Options]
MEDIA_NAME=DSK000001
MEDIA_SET=TAPEMEDIA_SET
APPEND=YES
APPEND_NO_OVERWRITE=YES
[Miscellaneous]
[Schedule]
RUN_IMMEDIATE=YES
```

```
[W2K]
[Selections]
PATH1=
SUBDIRS1=
```

Parameters other than INST_PATH can be specified multiple times for a mount point. However, note the following:

- Always leave the line before the MOUNT_POINT line blank.
- Specify the configuration definition parameters in the order listed in Table 4.38.
- When entering default ("default" must be lowercase) for MOUNT_POINT, the entry must come after all the other entries.
- Do not add a backslash (\) to the end of the path name entered for MOUNT_POINT.
- Path names entered for MOUNT_POINT are case sensitive.
- If you specify the number of parallel backup jobs for PARALLEL_COUNT, you must create CLASS or POLICY definitions for parallel backup in NetBackup.

  Note: When linkage is established between Protection Manager and Backup Exec, only serial backup using a single drive is allowed. Parallel backup or restoration using multiple tape drives is not allowed.
- Based on the CLASS or POLICY definitions created in NetBackup, specify a mount point for each CLASS or POLICY in the tape backup configuration definition file.

4.11.2.2 Configuration File Examples

- Using NetBackup (NBU.dat):

```
INST_PATH=C:\Program Files\VERITAS\NetBackup

MOUNT_POINT=F:
POLICY=backup2
SCHEDULE=schedule2

MOUNT_POINT=G:
POLICY=backup1
SCHEDULE=schedule1

MOUNT_POINT=default
POLICY=backup1
SCHEDULE=schedule3

PARALLEL_COUNT=2
```

- Using Backup Exec (BEWS.dat):

```
INST_PATH=C:\Program Files\VERITAS\Backup Exec\NT

MOUNT_POINT=E:
SCRIPT_FILE=C:\BEWS\E_Drive.txt

MOUNT_POINT=default
SCRIPT_FILE=C:\BEWS\backup1.txt
```

4.11.2.3 User-Defined Configuration Definition File

When you execute the `drmmediabackup` command or the `drmmediarestore` command with the `-bup_env` option (that is, when specifying a specific configuration definition file for each command), make sure that you create the corresponding configuration definition file in advance.

When you execute the `drmmediabackup` or `drmmediarestore` command without the `-bup_env` option, do not delete the default configuration definition file. The parameters in the default configuration definition file will be used to start up the backup management product.

When specifying a user-define name for the configuration definition file, the following requirements must be fulfilled:

- Maximum number of characters: 255 bytes including the director y name
- Usable characters: Characters valid for file names on Windows systems
- File location: *Protection-Manager-installation-destination*`\conf\tape\`

The following shows an example of creating a user-defined configuration definition file:

If you want to change the configuration definition file each day of the week, create seven types of configuration file as listed below, and define required parameters in each day-of-week file.

- `NBU_MONDAY.dat`
- `NBU_TUESDAY.dat`
- `NBU_WEDNESDAY.dat`
- `NBU_THURSDAY.dat`
- `NBU_FRIDAY.dat`
- `NBU_SATURDAY.dat`
- `NBU_SUNDAY.dat`

4.12 Creating a List File

Certain options of extended or basic commands may require you to specify multiple files, databases, copy groups, or other entities. In these situations, you can assign the option a definition file in which you have defined processing targets. Use of list files is optional; however, they eliminate the need to repeat the same operations.

This section explains how to create the following list files:

- Copy-group list file
- Database list file or mount point directory list file
- Transaction log list file

4.12.1 Creating a Copy-Group List File

A copy-group list file is a text file containing a list of copy groups.

The following explains the format of copy-group list files:

- File name

  You can use alphanumeric characters to specify any name.

- File storage location

  This file can be stored in any location.

  In a cluster environment, however, this file should be stored in a shared directory that can be accessed from both servers.

- Contents of a file

  Specify copy group names, following the rules listed below:

  – Write each copy group name from the leftmost column.

  – Copy group names are case-sensitive.

  – A line beginning with a number sign (#) is handled as a comment line. Each copy group name must be specified on one line.

  The following shows an example of entries in a copy-group list file:

  ```
  VG01,dev01
  VG01,dev02
  ```

  When creating a copy-group list file, see the RAID Manager configuration definition file (`horcm`*n*`.conf`) to check the copy groups to be backed up in batch. If a backup catalog has already been created, you can check this by executing the `drmfscat`, `drmsqlcat`, or `drmexgcat` command.

4.12.2 Creating a Database List File or Mount Point Directory List File

Certain command options require you to specify multiple mount point directories, files, databases, storage groups, or other entities. You can create a definition file (list file) beforehand that contains the definitions of target entities such as files, and then assign this definition file to the command option. Thus, you can specify multiple files, directories, databases, or storage groups at one time.

Use alphanumeric characters to specify the file name.

4.12.2.1 Commands that Support a Target Definition File

You can use a target definition file with the following commands:

- `drmexgbackup`
- `drmexgcat`
- `drmexgdisplay`
- `drmexgrestore`
- `drmfsbackup`
- `drmfscat`
- `drmfsdisplay`
- `drmfsrestore`
- `drmsqlbackup`
- `drmsqlcat`
- `drmsqldisplay`
- `drmsqlrestore`
- `drmsqllogbackup`

4.12.2.2 List File Storage Location

This file can be stored in any location. In a cluster environment, however, it should be stored in a shared directory that can be accessed from both servers.

4.12.2.3 Contents of the List File

The contents of the file must follow these conventions:

- Each parameter (file name, directory name, SQL Server database name, or storage group name) must be specified on one line.
- Lines beginning with # are assumed to be comment lines. However, if the SQL Server database name or storage group name starts with #, the line will be assumed as the SQL Server database name or storage group name, not as comment lines.
- The file names and directory names must be specified using absolute path names.

Example of entries in a target definition file:

```
# Example of specifying files
D:\data1\batch_0001\Tokyo_output_dir
D:\data1\batch_0001\Osaka_output_dir
D:\data1\transact.log
```

4.12.3 Creating a Transaction Log List File

The transaction log list file is used to specify the sequence of applying the transaction log files when performing recovery by using the options of the `drmsqlrecover` command. You create this file when performing recovery by using the transaction logs.

When using the `drmsqlrecovertool` command to recover data, you do not need a transaction log list file.

4.12.3.1 Transaction Log List File Storage Location

This file can be stored in any location. In a cluster environment, however, this file should be stored in a shared directory that can be accessed from both servers.

4.12.3.2 Contents of the Transaction Log List File

The contents of the file must follow these coding rules:

- Enter the name of the database, followed by the name of the transaction log file.
- Enclose the name of the database in square brackets "[" and "]".
- Enter the transaction log files for each database in the order in which they are to be applied.
- Each transaction log file name must be specified on one line.
- Each transaction log file name must be specified using the absolute path name.
- A transaction log file name must be left justified with no leading spaces.
- Lines beginning with # are assumed to be comment lines.

Example of entries in a target definition file:

```
# Protection Manager 4.1
# Log Backup Files
[SQLDB001]
C:\MSSQL\BackupLog\SQLDBLog001.bak
C:\MSSQL\BackupLog\SQLDBLog002.bak
C:\MSSQL\BackupLog\SQLDBLog003.bak

[SQLDB002]
C:\MSSQL\BackupLog\SQLDB2Log001.bak
C:\MSSQL\BackupLog\SQLDB2Log002.bak
C:\MSSQL\BackupLog\SQLDB2Log003.bak
```

4.12.3.3 About Automatic Creation of a Transaction Log List File

You can choose to automatically create a transaction log list file after restoration. If you choose automatic creation, Protection Manager detects the transaction logs required for recovery and write the logs in the order in which they should be recovered. Automatic creation is helpful in preventing errors. The number of errors will normally be smaller than when manual list file creation is chosen.

To specify automatic creation of a transaction log list file, use the following command:

```
PROMPT> drmsqllogbackup instance-name -target database-name -v >
transaction-log-list-file-name (absolute path)
```

For example, if you want to create a transaction log list file in C:\temp\SQLTXLOG.txt when the SQL Server instance name is SQL1 and the database name is DB1, execute the command as follows:

```
PROMPT> drmsqllogbackup SQL1 -target DB1 -v > C:\temp\SQLTXLOG.txt
```

4.13 Updating a Dictionary Map File

After environment setup is complete, the next step is to update the dictionary map file. By updating the dictionary map file, you can confirm that the environment has been set up appropriately.

To update the dictionary map file:

- When the backup is of a file system:

  Execute the `drmfsdisplay` command with the `-refresh` option specified.

- When the backup is of a SQL Server database:

  Execute the `drmsqldisplay` command with the `-refresh` option specified.

- When the backup is of an Exchange database:

  Execute the `drmexgdisplay` command with the `-refresh` option specified.

In the following example, the command updates the dictionary map file:

```
PROMPT> drmfsdisplay -refresh
```

When the command terminates, the following message appears:

```
KAVX0023-I Dictionary map file was updated.
```

When this message appears, confirm that the dictionary map file has been updated by, for example, checking the date of the file.

Notes:

- In a cluster configuration, dictionary map file updating must be performed on the active server. You do not need to perform updating on both servers by switching the cluster.
- In a cluster configuration, before updating the dictionary map file, you need to assign the virtual server name to the environment variable `DRM_HOSTNAME`. If you update the dictionary map file without specifying the virtual server name, the default dictionary map file will be updated instead of the dictionary map file created in the shared directory.
- Usually, you do not need to update the dictionary map file on the backup server. However, if a primary volume exists in the backup server, the dictionary map file must be updated.

Even after starting operations, you need to update the dictionary map file when the operating environment is updated. For details on the timing of updating the dictionary map, see section 5.3.7

4.14 Specifying How to Mount a Secondary Volume

When backing up from a secondary volume to a tape, it is necessary to mount a secondary volume on the backup server.

To mount a secondary volume, the following operations may be required:

- Creating a copy group mount definition file
- Making the OS dynamically recognize the secondary volume

You must create a copy group mount definition file when the backup is of a dynamic disk, and in any of the following situations:

- When VSS is used for the backup operation
- When Protection Manager Console is used
- When mounting the secondary volume by specifying a copy group

You can create a copy group mount definition file even if the configuration is not listed above. If a copy group mount definition file does not exist, Protection Manager programs determine a mount point automatically. Once you create a copy group mount definition file, you can perform the following operations:

- You can mount the secondary volume to a specified mount point.
- If the same number of secondary volumes were provided as the number of multiple generations, by defining a mount point for each generation, you can omit the mount point specification when executing the command for mounting a secondary volume.

There are two methods to make the operating system recognize the secondary volume: one is to make the operating system dynamically recognize the secondary volume, and another is to make the operating system statically recognize the secondary volume.

- Making the operating system dynamically recognize the secondary volume:

  Use the `drmdevctl` command to conceal the secondary volume from the operating system. This enables Protection Manager to mount and unmount the target secondary volume only when performing a backup or restore. When using any of the following configurations, make sure that you use this method. You can use this method in other cases as well.

  - Use this method when there are multiple generations of a secondary volume which is a dynamic disk.
  - There are multiple generations of a secondary volume in a cluster configuration using Microsoft Cluster Service.
  - Use this method when there are multiple generations of a secondary volume, and VSS is being used.

  Note: Of the physical disks connected to the device adapter port, do not use the LUN#0 disk for the primary and secondary volumes if the secondary volume is to be recognized dynamically. If you use the LUN#0 physical disk, the system might not be able to recognize the other physical disks.

- Making the operating system statically recognize the secondary volume:

  Reboot the backup server to make the operating system statically recognize a single generation secondary volume. Protection Manager performs backups and restorations using this secondary volume. This method is applicable when there is a single generation of the secondary volume only.

After starting this operation, it is necessary to delete the copy group mount definition file, and specify the method for mounting a secondary volume after a system initial construction or volume configuration changes.

For details on how a mount point and secondary volume (copy group) are determined when the command for mounting a secondary volume is executed, see section 4.14.3.

4.14.1 Enabling Dynamic Recognition of a Secondary Volume

This section describes the procedure for specifying how to mount a secondary volume by making the operating system dynamically recognize the secondary volume. The following describes cases where a copy group mount definition file is created and cases where the file is not created.

A copy group mount definition file is created in the following location:

```
Protection-Manager-installation-destination\conf\vm\CG_MP.conf
```

To specify how to mount a secondary volume:

1. Conceal all secondary volumes from the backup server by executing the `drmdevctl` command on the backup server.

```
BKServer > drmdevctl -detach
BKServer >
```

2. Use the Windows disk management functionality or a RAID Manager command to confirm that the secondary volumes are concealed. The following example uses the `inqraid` command of RAID Manager:

```
BKServer >echo hd0-1 | inqraid -CLI
DEVICE_FILE     PORT    SERIAL  LDEV CTG  H/M/12  SSID R:Group PRODUCT_ID
Harddisk0       -            -     -   -       -     -       - -
Harddisk1       -            -     -   -       -     -       - -
BKServer >
```

3. Create backup catalogs for all generations by executing the backup command on the file server or database server. Repeat the backup command for each generation.

   When the backup is of a file system, create a mount point directory batch definition file that describes mount points of file systems to be backed up. In this example, *Protection-Manager-installation-folder*\conf\fs\mp_on_dg01.dat is assumed as a mount point directory batch definition file:

```
FSServer > drmfsbackup mp_on_dg01.dat -mode cold
FSServer >
FSServer > drmfsbackup mp_on_dg01.dat -mode cold
FSServer >
FSServer > drmfsbackup mp_on_dg01.dat -mode cold
FSServer >
```

   When the backup is of a SQL Server database:

```
DBServer > drmsqlbackup DEFAULT
DBServer >
DBServer > drmsqlbackup DEFAULT
DBServer >
DBServer > drmsqlbackup DEFAULT
DBServer >
```

   When the backup is of an Exchange database:

```
DBServer > drmexgbackup
DBServer >
DBServer > drmexgbackup
DBServer >
DBServer > drmexgbackup
DBServer >
```

4. To confirm that the backup catalogs are created for all generations, on the file server or database server, execute a command to display backup information (drmfscat command, drmsqlcat command, or drmexgcat command). The number of backup information items should match the number of generations.

   When the backup is of a file system:

```
FSServer > drmfscat mp_on_dg01.dat
INSTANCE: mp_on_dg01.dat
BACKUP-ID:0000000001 BACKUP-MODE: COLD INSTANCE: mp_on_dg01.dat ORIGINAL-ID:0000000001
START-TIME:2004/12/15 10:00:00 END-TIME: 2004/12/15 10:03:00 HOSTNAME:    FILESERV1
:
:
INSTANCE: mp_on_dg01.dat
BACKUP-ID:0000000002 BACKUP-MODE: COLD INSTANCE: mp_on_dg01.dat ORIGINAL-ID:0000000002
START-TIME:2004/12/15 10:10:00 END-TIME: 2004/12/15 10:1300 HOSTNAME:    FILESERV1
:
:
mp_on_dg01.dat
BACKUP-ID:0000000003 BACKUP-MODE: COLD INSTANCE: mp_on_dg01.dat ORIGINAL-ID:0000000003
:
:
```

When the backup is of a SQL Server database:

```
DBServer > drmsqlcat DEFAULT
INSTANCE: DEFAULT
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: DEFAULT ORIGINAL-ID: 0000000001
START-TIME: 2004/12/15 10:00:00 END-TIME: 2004/12/15 10:03:00 HOSTNAME: DB_SVR1
:
:
INSTANCE: DEFAULT
BACKUP-ID: 0000000002 BACKUP-MODE: ON-LINE INSTANCE: DEFAULT ORIGINAL-ID: 0000000002
START-TIME: 2004/12/15 10:10:00 END-TIME: 2004/12/15 10:13:00 HOSTNAME: DB_SVR1
:
:
INSTANCE: DEFAULT
BACKUP-ID: 0000000003 BACKUP-MODE: ON-LINE INSTANCE: DEFAULT ORIGINAL-ID: 0000000003
START-TIME: 2004/12/15 10:20:00 END-TIME: 2004/12/15 10:23:00 HOSTNAME: DB_SVR1
:
:
```

When the backup is of an Exchange database:

```
DBServer > drmexgcat
STORAGEGROUP: STR1
BACKUP-ID:0000000001 BACKUP-MODE: ONLINE STORAGEGROUP: STR1 ORIGINAL-ID:0000000001
START-TIME:2004/12/15 10:00:00 END-TIME: 2004/12/15 10:01:00 HOSTNAME:EXG_SRV1
:
:
STORAGEGROUP: STR1
BACKUP-ID:0000000002 BACKUP-MODE: ONLINE STORAGEGROUP: STR1 ORIGINAL-ID:0000000002
START-TIME:2004/12/15 10:10:00 END-TIME: 2004/12/15 10:11:00 HOSTNAME:EXG_SRV1
:
:
STORAGEGROUP: STR1
BACKUP-ID:0000000003 BACKUP-MODE: ONLINE STORAGEGROUP: STR1 ORIGINAL-ID:0000000003
START-TIME:2004/12/15 10:20:00 END-TIME: 2004/12/15 10:21:00 HOSTNAME:EXG_SRV1
:
:
```

5. Export the backup catalog information for all generations to temporary files by executing the `drmdbexport` command. The temporary files are stored in a temporary directory for extended commands that is on the file server. Repeat the `drmdbexport` command for the number of generations. The following example shows the `drmdbexport` command:

```
FSServer > drmdbexport 0000000001 -f C:\tmp\0000000001.drm
FSServer > drmdbexport 0000000002 -f C:\tmp\0000000002.drm
FSServer > drmdbexport 0000000003 -f C:\tmp\0000000003.drm
```

6. Transfer exported backup catalog information from the file server or database server to the backup server by executing the `ftp` (file transfer protocol) command. In the following example, the FTP root folder is assumed to be `C:\FTP_ROOT`:

```
FSServer> ftp backup-server-name
ftp> Username: (login-name)
ftp> password: (password)
ftp> binary
ftp> put C:\tmp\0000000001.drm
ftp> put C:\tmp\0000000002.drm
ftp> put C:\tmp\0000000003.drm
ftp> quit
FSServer>
```

7. Import temporary files transferred from the file server into backup catalogs on the backup server by executing the `drmdbimport` command on the backup server. Repeat the `drmdbimport` command for each generation. The following example shows the `drmdbimport` command:

```
BKServer > drmdbimport -f C:\FTPROOT\tmp\0000000001.drm
BKServer > drmdbimport -f C:\FTPROOT\tmp\0000000002.drm
BKServer > drmdbimport -f C:\FTPROOT\tmp\0000000003.drm
```

8. Create a copy group mount definition file by executing the `drmmount` command on the backup server with the backup ID and `-conf` option. Concealment of the secondary volumes is canceled during this operation. If you do not create a copy group mount definition file, execute the `drmmount` command without the `-conf` option specified to confirm that a mount operation can be performed. The following example shows the `drmmount` command:

```
BKServer > drmmount 0000000001 -conf
KAVX0001-I The drmmount command will now start.
KAVX0400-I Mounting will now start.
           Mount point = X:
KAVX0401-I Mounting has finished.
           Mount point = X:
KAVX0443-I The copy group mount definition file was updated.
         File name = C:\Program Files\Hitachi\drm\conf\vm\CG_MP.conf
KAVX0002-I The drmmount command will now end.
```

9. Unmount the secondary volume by executing the `drmumount` command. The secondary volume becomes concealed as the `drmumount` command is executed. The following example shows the `drmumount` command.

```
BKServer > drmumount 0000000001
KAVX0001-I The drmumount command will now start.
KAVX0406-I Unmounting will now start.
           Mount point = X:
KAVX0407-I Unmounting has finished.
           Mount point = X:
KAVX0002-I The drmumount command will now end.
```

10. Repeat creating the copy group mount definition file and unmounting the secondary volume for each generation, as executed in Step 8 and 9. In this example, execute those operations, specifying backup IDs `0000000002` and `0000000003` in the argument:

```
BKServer > drmmount 0000000002 -conf
BKServer > drmumount 0000000002
BKServer > drmmount 0000000003 -conf
BKServer > drmumount 0000000003
```

11. Check the contents of the created copy group mount definition file using a text editor. The created copy group mount definition file is stored in the `Protection-Manager-installation-directory\conf\vm`..

    Change the `FS` values (mount point of the mount target) as necessary.

    If you do not create a copy group mount definition file, this step is not necessary.

```
#configuration file for copy-group mounting
[CG_DEF]
COPY-GROUP-NUM=6
COPY-GROUP=Grp1,dev11
BUID=0000000001
DATETIME=20041201123456

COPY-GROUP=Grp1,dev12
BUID=0000000001
DATETIME=20041201123456

COPY-GROUP=Grp2,dev21
BUID=0000000002
DATETIME=20041201123456

COPY-GROUP=Grp2,dev22
BUID=0000000002
DATETIME=20041201123456

COPY-GROUP=Grp3,dev31
BUID=0000000001
DATETIME=20041201123456

COPY-GROUP=Grp3,dev32
BUID=0000000003
DATETIME=20041201123456

[DG_DEF]
DISK-GROUP-NUM=2
BUID=0000000001
DATETIME=20041123225248
COPY-GROUP-NUM=2
COPY-GROUP=Grp1,dev11
COPY-GROUP=Grp1,dev12
P-DG=DG01
S-DG=DG01
FS-NUM=3
LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume1
LABEL=Volume1
FS=X:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume2
LABEL=Volume2
FS=Y:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume3
LABEL=Volume3
FS=Z:

BUID=0000000002
DATETIME=20041201123456200412012345620041201123456
COPY-GROUP-NUM=2
COPY-GROUP=Grp2,dev21
COPY-GROUP=Grp2,dev22
P-DG=DG01
S-DG=DG01
FS-NUM=3
LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume1
LABEL=Volume1
```

```
FS=X:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume2
LABEL=Volume2
FS=Y:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume3
LABEL=Volume3
FS=Z:

BUID=0000000003
DATETIME=20041201123456
COPY-GROUP-NUM=2
COPY-GROUP=Grp3,dev31
COPY-GROUP=Grp3,dev32
P-DG=DG01
S-DG=DG01
FS-NUM=3
LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume1
LABEL=Volume1
FS=X:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume2
LABEL=Volume2
FS=Y:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume3
LABEL=Volume3
FS=Z:
```

12. To perform a backup or restore, execute the `drmdevctl` command to conceal all secondary volumes from the backup server.

4.14.2 Mounting a Secondary Volume for Static Recognition by the OS

This section describes the procedure for specifying how to mount a secondary volume by making the OS statically recognize the secondary volume. The following describes both cases where a copy group mount definition file is created, and cases where the file is not created.

A copy group mount definition file is created in the following location:

Protection-Manager-installation-destination\conf\vm\CG_MP.conf

To specify how to mount a secondary volume:

1. Create a backup catalog by executing the backup command on the file server or database server.

   When the backup is of a file system, create (in advance) a mount point directory batch definition file in which the mount point of the file system to be backed up is described. In this example, *Protection-Manager-installation-destination*\conf\fs\mp_on_dg01.dat is assumed as a mount point directory batch definition file:

```
FSServer > drmfsbackup mp_on_dg01.dat -mode cold
FSServer >
```

When the backup is of a SQL Server database:

```
DBServer > drmsqlbackup DEFAULT
DBServer >
```

When the backup is of an Exchange database:

```
DBServer > drmexgbackup
DBServer >
```

2. Confirm that a backup catalog has been created by executing a command (`drmfscat`, `drmsqlcat`, or `drmexgcat`) to display backup information on the file server or database server.

   When the backup is of a file system:

```
FSServer > drmfscat mp_on_dg01.dat
INSTANCE: mp_on_dg01.dat
BACKUP-ID:0000000001 BACKUP-MODE: COLD INSTANCE: mp_on_dg01.dat ORIGINAL-ID:0000000001
START-TIME:2004/12/15 10:00:00 END-TIME: 2004/12/15 10:03:00 HOSTNAME:    FILESERV1
:
:
```

   When the backup is of a SQL Server database:

```
DBServer > drmsqlcat DEFAULT
INSTANCE: DEFAULT
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: DEFAULT ORIGINAL-ID: 0000000001
START-TIME: 2004/12/15 10:00:00 END-TIME: 2004/12/15 10:03:00 HOSTNAME: DB_SVR1
:
:
```

   When the backup is of an Exchange database:

```
DBServer > drmexgcat
STORAGEGROUP: STR1
BACKUP-ID:0000000001 BACKUP-MODE: ONLINE STORAGEGROUP: STR1 ORIGINAL-ID:0000000001
START-TIME:2004/12/15 10:00:00 END-TIME: 2004/12/15 10:01:00 HOSTNAME:EXG_SRV1
:
:
```

3. Export the backup catalog information to a temporary file by executing the `drmdbexport` command. The temporary file is saved in the temporary directory for extended commands that is on the file server:

```
FSServer > drmdbexport 0000000001 -f C:\tmp\0000000001.drm
```

4. Transfer the temporary file exported from the file server or the database server to the backup server by executing the `ftp` (file transfer protocol) command. In this case, set the FTP route folder `C:\FTP_ROOT` as shown here:

```
FSServer> ftp backup-server-name
ftp> Username: (login-name)
ftp> password: (password)
ftp> binary
ftp> put C:\tmp\0000000001.drm
ftp> quit
FSServer>
```

5. Import the backup catalog information transferred from the file server to the backup catalog of the backup server.

   Import the temporary file transferred from the file server to the backup catalog of the backup server by executing the `drmdbimport` command on the backup server:

```
BKServer > drmdbimport -f C:\FTPROOT\tmp\0000000001.drm
```

6. Restart the backup server in order to make the server recognize disk the information of the primary volume that has been backed up. After the server has restarted, use the Windows disk management functionality to confirm that the dynamic disk is recognized correctly.

   When it is recognized as an online disk, the disk information is recognized correctly.

   When it is recognized as either an offline disk or a disk in a different format, execute the operation to import the disk in a different format. If the failed volume remains after importing the disk in a different format, execute the operation to reactivate the volume.

7. Create a copy group mount definition file by executing the `drmmount` command on the backup server with the backup ID and `-conf` option specified.

   If you do not create a copy group mount definition file, execute the `drmmount` command without the `-conf` option specified to confirm that a mount operation can be performed:

```
BKServer > drmmount 0000000001 -conf
KAVX0001-I The drmmount command will now start.
KAVX0400-I Mounting will now start.
           Mount point = X:
KAVX0401-I Mounting has finished.
           Mount point = X:
KAVX0400-I Mounting will now start.
           Mount point = Y:
KAVX0401-I Mounting has finished.
           Mount point = Y:
KAVX0400-I Mounting will now start.
           Mount point = Z:
KAVX0401-I Mounting has finished.
           Mount point = Z:
KAVX0443-I Updating the copy group mount definition file has finished.
             file-name=C:\Program Files\Hitachi\drm\conf\vm\CG_MP.conf
KAVX0002-I The drmmount command will now end.
```

8. Unmount the secondary volume by executing the `drmumount` command on the backup server:

```
BKServer > drmumount 0000000001
KAVX0001-I The drmumount command will now start.
KAVX0406-I Unmounting will now start.
           Mount point = X:
KAVX0407-I Unmounting has finished.
           Mount point = X:
KAVX0406-I Unmounting will now start.
           Mount point = Y:
KAVX0407-I Unmounting has finished.
           Mount point = Y:
KAVX0406-I Unmounting will now start.
           Mount point = Z:
KAVX0407-I Unmounting has finished.
           Mount point = Z:
KAVX0002-I The drmumount command will now end.
```

9. Use a text editor to open the created copy group mount definition file, and check the contents of the file. Change the FS value (mount point of the mount destination) as necessary. If you do not create a copy group mount definition file, this step is not necessary:

```
#configuration file for copy-group mounting
[CG_DEF]
COPY-GROUP-NUM=6
COPY-GROUP=Grp1,dev11
BUID=0000000001
DATETIME=20041201123456

COPY-GROUP=Grp1,dev12
BUID=0000000001
DATETIME=20041201123456

[DG_DEF]
DISK-GROUP-NUM=2
BUID=0000000001
DATETIME=20041123225248
COPY-GROUP-NUM=2
COPY-GROUP=Grp1,dev11
COPY-GROUP=Grp1,dev12
P-DG=DG01
S-DG=DG01
FS-NUM=3
LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume1
LABEL=Volume1
FS=X:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume2
LABEL=Volume2
FS=Y:

LVM-DEVICE=\Device\HarddiskDmVolumes\Dg01\Volume3
LABEL=Volume3
FS=Z:
```

4.14.3 Determining a Mount Point and Target for Secondary Volume Mounting

This section describes how Protection Manager determines a mount point and secondary volume (copy group) to be backed up when the command for mounting a secondary volume is executed. The commands for mounting a secondary volume are as follows:

Basic command:

The `drmmount` command

Extended commands that contain the `drmmount` command:

- `EX_DRM_CACHE_PURGE`
- `EX_DRM_MOUNT`
- `EX_DRM_TAPE_BACKUP`
- `EX_DRM_TAPE_RESTORE`
- `EX_DRM_EXG_VERIFY`

4.14.3.1 Determining a Mount Point

Table 4.39 describes how Protection Manager determines a mount point when executing the command for mounting a second volume and how to determine a mount point when the command is executed with a backup ID specified in the option.

Table 4.39 Determining a Mount Point When the Command for Mounting a Secondary Volume is Executed With a Backup ID Specified

Is a mount point name specified by the -mount_pt option when executing the command?	Copy group mount definition file	Mount point
Any mount point name is specified or the definition file exists.	Any mount point name is specified or the definition file exists.	A mount point specified in the copy group mount definition file and a mount point name specified in the `-mount_pt` option are used[1]
	No mount point name is specified or the definition file does not exist.	A mount point specified in the `-mount_pt` option is used[2]
No mount point name is specified or the definition file does not exist.	Any mount point name is specified or the definition file exists.	A mount point in the copy group mount definition file is used[3]
	No mount point name is specified or the definition file does not exist.	Protection Manager determine a mount point automatically (Unused drives will be selected.)

[1] When a drive name is specified in the `-mount_pt` option, the system selects the unoccupied drives from the specified drives and mounts them in alphabetical order.

If you specify an absolute path that begins with the drive letter as a mount point directory name in the `-mount_pt` option, the mount point should be `absolute-path-specified-as-a-mount-point-directory-name\mount-point-in-the-copy-group-mount-definition-file`

For example, when `W:\` is specified in the `-mount_pt` option and `FS=D:\` is specified in the copy group mount definition file, the system mounts the unoccupied drives from the `W` drive. When `W:\mnt` is specified in the `-mount_pt` option and `FS=D:\mnt` is specified in the copy group mount definition file, the mount point will be `W:\mnt\D\mnt`.

[2] If a backslash (`\`) is not attached to the end of the specified mount point, it is assumed that a backslash (`\`) exists. For example, the system recognizes `D:` is the same drive as `D:\`. `D:\MOUNTDIR` is the same directory as `D:\MOUNTDIR\`.

[3] The system uses a mount point specified in `FS=`*`mountpoint`* that corresponds to a copy group to be backed up, in the copy group mount definition file.

Table 4.40 explains the process when the command is executed with a copy group specified in the option.

Table 4.40 Determining a Mount Point When the Command for Mounting a Secondary Volume is Executed

Is a mount point name specified by the -mount_pt option when executing the command?	Disk type	Copy group mount definition file	Mount point
Any mount point name is specified or the definition file exists.	Basic disk	Any mount point name is specified or the definition file exists.	A mount point specified in the copy group mount definition file[1]
		No mount point name is specified or the definition file does not exist.	A mount point name specified by the -mount_pt option[2]
	Dynamic disk	Any mount point name is specified or the definition file exists.	A mount point in the copy group mount definition file and mount point name specified by the -mount_pt option[3]
		No mount point name is specified or the definition file does not exist.	Terminate in error
No mount point name is specified or the definition file does not exist.	Basic disk	Any mount point name is specified or the definition file exists.	A mount point in the copy group mount definition file[1]
		No mount point name is specified or the definition file does not exist.	A mount point automatically determined by Protection Manager (Drives that are not in use are selected.)
	Dynamic disk	Any mount point name is specified or the definition file exists.	A mount point in the copy group mount definition file[3]
		No mount point name is specified or the definition file does not exist.	A mount point automatically determined by Protection Manager (Drives that are not in use are selected.)

[1] The system uses a mount point specified in FS=*mountpoint* that corresponds to a copy group to be backed up, in the copy group mount definition file.

[2] If a backslash (\) is not attached to the end of the specified mount point, it is assumed that a backslash (\) exists. For example, the system recognizes D: is the same drive as D:\. D:\MOUNTDIR is the same directory as D:\MOUNTDIR\.

[3] When a drive name is specified in the -mount_pt option, the system selects the unoccupied drives from the specified drives and mounts them in alphabetical order.

If you specify an absolute path that begins with the drive letter as a mount point directory name in the -mount_pt option, the mount point should be *absolute-path-specified-as-a-mount-point-directory-name\mount-point-in-the-copy-group-mount-definition-file*

For example, when W:\ is specified in the -mount_pt option and FS=D:\ is specified in the copy group mount definition file, the system mounts the unoccupied drives from the W drive. When W:\mnt is specified in the -mount_pt option and FS=D:\mnt is specified in the copy group mount definition file, the mount point will be W:\mnt\D\mnt.

4.14.3.2 Determining a Mount Target

Table 4.41 describes how Protection Manager determines a secondary volume (copy group) to be backed up when executing the command for mounting a second volume.

Table 4.41 Determining a Mount Target When the Command for Mounting a Secondary Volume is Executed

An option specified at the command execution	Disk type	Copy group mount definition file	Secondary volumes to be backed up (copy group)
Backup ID	Any disk type	Not distinguished	Copy groups that have been recorded in the backup catalog corresponding to the specified backup ID
`-copy_group` *`copy-group-name`*	Basic disk	Not distinguished	Copy groups specified by the `-copy_group` option
	Dynamic disk	The definition file exists.	All copy groups that make up the same disk group (disk set)
		The definition file does not exist.	Terminate in error

4.15 Preparing to Execute Extended Commands

This section explains steps you must perform before executing extended commands. Table 4.42 provides an overview of the preparations required before executing extended commands.

Table 4.42 Extended Command Execution Preparation Tasks Needed for Each Kind of Server

Tasks	File server or database server	Backup server	Section
Check the prerequisite conditions	Required	Required	4.15.1
Set up an FTP service (if you are going to back up to tape).	Not required	Required	4.15.2
Specify how the extended commands are to be started.	Required	Required	4.15.3
Create host environment settings files.	Required	Required	4.15.4
Create operation IDs	Not required	Not required	4.15.5
Create operation definition files (backup of a file system).	Required	Required	4.15.6
Create operation definition files (backup of a SQL Server Database).	Required	Required	4.15.7
Create operation definition files (backup of an Exchange Database).	Required	Required	4.15.8
Create a temporary directory for extended commands.	Required	Required	4.15.9
Create a copy-group list file.	Required	Not required	4.15.10
Set up an FTP service (if you are going to back up to tape).	Not required	Required	4.15.11

4.15.1 Prerequisites for Using Extended Commands

To execute the extended commands, you must have system administrator permissions.

The VBScript version must be 5.1 or 5.6 when extended commands are used.

Note: The returned value is other than 0 when an extended command fails. If you upgrade the VBScript version from version 5.1, a different value is returned when an extended command fails.

Check the following conditions before using extended commands:

- Execution permissions for extended commands
- Automatic execution of extended commands
- Prerequisite version of VBScript
- Deletion of unnecessary files

4.15.1.1 Automatic Execution of Extended Commands

A backup or restoration process that uses extended commands may be performed across more than one server. It is difficult to execute extended commands at multiple physically remote servers. Therefore, you need to perform remote execution of the extended commands. If you want to repeat execution of extended commands, you need to set up automatic execution.

In such a case, install the operation management software that meets the following requirements:

- The software can be started by specifying the file name in VBScript (Microsoft Visual Basic® Scripting Edition) and shell script programs (batch files).
- Operations such as repeat executions of programs and evaluating conditions can automatically be executed, according to the predefined execution order and schedule.

We recommend that you use operation management software to simplify the process of executing extended commands.

4.15.1.2 Deleting Unnecessary Files

Before you can change the extended command settings you specified, the files that have been used and are no longer necessary must be deleted. You need to delete unnecessary files when renaming or discarding an operation ID, or when changing the FTP user.

Deleting Files When Renaming or Discarding an Operation ID:

To rename or discard a created operation ID, you must delete the temporary directory for extended commands that was automatically generated. To delete the temporary directory for extended commands, perform the following steps on the file server or database server:

a) If `1` is assigned to `SET_DRM_HOSTNAME` in the operation definition file, assign the environment variable `DRM_HOSTNAME` the server name assigned to `DB_SERVER_NAME` in the operation definition file. For example, if the server name assigned to `DB_SERVER_NAME` is `VHOST`, specify as follows: `PROMPT> set DRM_HOSTNAME=VHOST`

b) Execute `drmfsdisplay` with the `-v` option specified, and check the directory name displayed after `DB_PATH`:.

c) Delete the *`operation-ID`* directory created in the `script_work` directory in the same level of the directory that you checked in step 2.

Perform the following steps on the backup server:

a) Confirm the name of the temporary directory for extended commands. Check whether the value of `FTP_HOME_DIR` and the value of `\FTP_SUB_DIR` exists.

b) Delete the *`operation-ID`* directory created in the directory that you checked in step 1.

Changing the FTP User:

Before changing the FTP user to be used when the extended command `EX_DRM_FTP_PUT` or `EX_DRM_FTP_GET` is executed, you must delete the files created by the current FTP user. Delete the files that were created in the following directory on the backup server:

`FTP_HOME_DIR-value``\`*`FTP_SUB_DIR-value`*`\`*`operation-ID`*`\BK`

4.15.2 Setting up FTP Service (Tape Backup)

When you execute an extended command, temporary files are generated containing the information needed for backup and restore processing. If you are executing extended commands in a system equipped with a file server or a database server and a backup server, these temporary files are transferred between the servers using the FTP service. Before using the FTP service, set up an FTP server on the backup server.

The IIS FTP service must be installed on the backup server. When installing the IIS FTP service, in the property setting of the FTP site, specify `MS-DOS` for the directory display style.

4.15.3 Specifying How Extended Commands are Started

You use the MS-DOS prompt or operation management software to start extended commands.

If you want to use operation management software to control extended commands, start them by using the program execution definition functionality of the software. When you use this functionality, specify extended command names as the names of programs to be executed.

If you want to start an extended command by using the MS-DOS prompt, take either of the following actions so that `cscript.exe` starts the extended commands:

- Change the host parameter, and then execute the extended commands:

  From the MS-DOS prompt, enter the following command. Once you change the host parameter, you do not need to change it again.

  ```
  cscript //H:Cscript
  ```

- Specify the `cscript` command so that it executes an extended command.

  Execute the `cscript` command in the following format:

  ```
  cscript name-of-the-extended-command-you-want-to-execute
  ```

In the Windows environment default settings, `wscript.exe` starts an extended command and a pop-up dialog box appears. The dialog box displays the execution results of the command. Therefore, note that an extended command cannot execute automatically.

4.15.4 Creating a Host Environment Settings File

Host environment settings files are created depending on the configuration of the system using extended commands. These files must be located on each server. The host environment settings file is created automatically during installation. Check the contents of the file as necessary.

The host environment settings file is created in the following location:

Protection-Manager-installation-folder\script\conf\host.dat

A sample host environment settings file is provided in the following location:

Protection-Manager-installation-folder\script\conf\host.dat.model

Table 4.43 shows the entries of a host environment settings file, the values of them, and the maximum value lengths (in bytes).

Table 4.43 Value and Maximum Value Length of Each Entry in a Host Environment Settings File

Entry name	Explanation	Default (specifiable range)	Omission#
HOST_ROLE	This entry specifies a host type. Specify DB for the file server or the database server, and BK for the backup server.	DB (DB or BK)	N
MAX_LOG_LINES	This entry specifies the maximum number of lines that the log file for scripts can contain. Specify an integer (specifiable range: from 1,000 to 100,000). One extended command outputs approximately 20 lines. Note, however, that when you execute EX_DRM_FTP_PUT or EX_DRM_FTP_GET, approximately 100 lines are output. Based on the above approximate estimate, determine the value of MAX_LOG_LINES.	1000 (1000 to 100000)	N
MSG_OUTPUT	This entry can be specified only in Windows. Specifies whether to output messages from basic commands that are executed internally by an extended command. If NORMAL is specified, messages from basic commands are not output. If DETAIL is specified, messages from basic commands are output. Messages from basic commands are not output to a script log file.	NORMAL (NORMAL or DETAIL)	Y

Legend:

Y: Can be omitted.

N: Cannot be omitted.

For an item that can be omitted, the default value is used for operation.

After creating the host environment settings file, in the created server execute the check tool (EX_DRM_HOST_DEF_CHECK) for the host environment settings file as follows:

On the file or database server:

```
EX_DRM_HOST_DEF_CHECK -db -f host-environment-settings-file
```

On the backup server:

```
EX_DRM_HOST_DEF_CHECK -bk -f host-environment-settings-file
```

If an error is found, revise the contents of the file, and then re-execute the check tool.

4.15.5 Creating Operation IDs

An operation ID is a unique character string given to each resource that is to be a processing target of an extended command. You must define beforehand the information that the extended command will reference during execution in the operation definition file. This operation definition file must be created at a certain location in the "*_operation-ID*.dat" format. By passing the operation ID in an argument to an extended command, you can allow the extended command to reference the contents of the operation definition file.

Before setting up an extended command, determine a name that uniquely identifies the target on any server involved in execution of the extended command. An example of specifying an operation ID is provided here:

"*server-name-of-operation-target*" + "*resource-name-to-be-operated*"

An operation ID can have up to 64 ASCII characters. The operation ID string can contain the following special characters: !, #, $, %, &, ', (,), +, _, –, ^, @, ,, .

The following restrictions are placed on using these special characters:

- You cannot create an operation ID that begins with a hyphen (-)
- You cannot create an operation ID using a period (.) by itself
- If an operation ID contains an ampersand (&) or a caret (^), the operation ID must be enclosed in double quotation marks (")

Operation IDs are not case sensitive. Note the following when using an operation ID:

- You cannot specify the same operation ID for multiple extended commands and execute these commands in parallel. If you do, the extended commands may terminate with an error or the backup catalog that is the target of the extended commands may be damaged.

  However, you can execute an EX_DRM_SQL_TLOG_BACKUP command in parallel that is specifying the same operation ID as another extended command.
- To rename or stop using a created operation ID, you must delete the temporary directory for extended commands that has been generated automatically. For information on how to delete the temporary directory for extended commands, see section 4.15.1.2.

4.15.6 Creating Operation Definition Files for File System Backup

Create an operation definition file according to the configuration of the system using extended commands. After creating the operation definition files, store them in the following directory of each server where the extended commands will be executed:

Protection-Manager-installation-directory\script\conf\

After storing the operation definition file in the stipulated directory, execute the check tool for operation definition files to check the content of the operation definition file and automatically create a temporary directory for extended commands.

4.15.6.1 Deployment of Operation Definition Files

Create two same operation definition files for each target mount point or file. Place one operation definition file on the file server and the other on the backup server. In a cluster configuration, the same operation definition file must be located on all the servers where the cluster resource (mount point or file) to be backed up is defined.

Figure 4.10 shows the locations of operation definition files. In this example, the virtual server VServer_C is running on the file server A (server name: FSServer_A) and the file server B (server name: FSServer_B), which are making up the cluster.

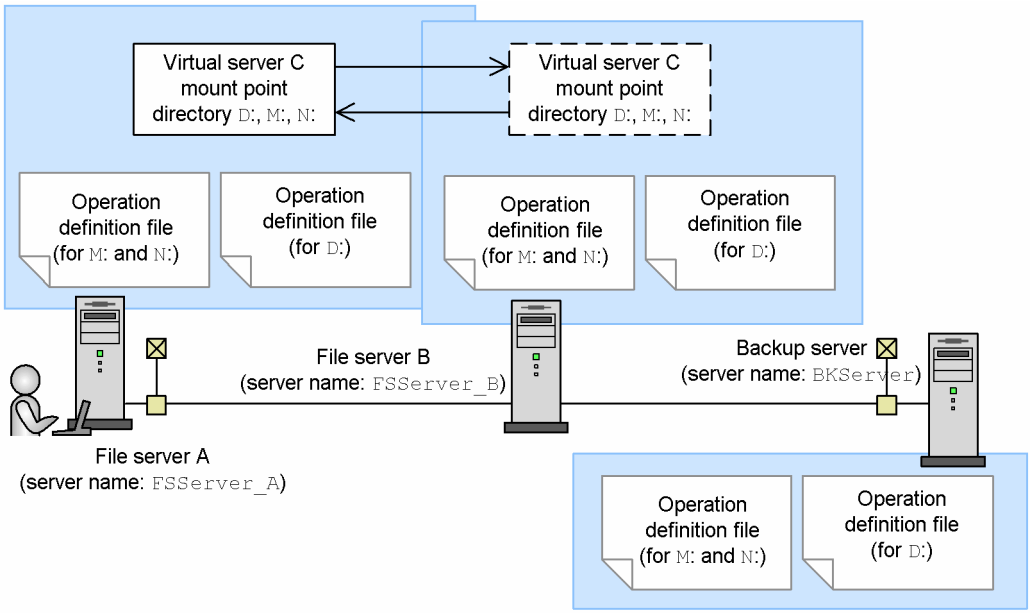


Figure 4.10 Example of Deploying Operation Definition Files (Cluster Configuration, File System Backup)

The three mount points D:, M:, and N: exist on the file server A and file server B, and the virtual server C (server name: VServer_C) has been defined as a cluster resource. The mount points M: and N: are assumed to be set in the mount point directory target definition file app.txt so that the mount point directories are operated at one time.

In this example, on the file server A and the file server B, place the operation definition file about D: and the operation definition file about the mount point directories (M: and N:) specified in the app.txt. On the backup server (server name: BKServer), place the operation definition file about D: and the operation definition file about M: and N:.

4.15.6.2 Format of Operation Definition Files

An operation definition file is created, for each operation ID that uniquely identifies the processing target, with a name in the following format:

Protection-Manager-installation-destination\script\conf_*operation-ID*.dat

The sample file for the operation definition file is stored in the following location:

Protection-Manager-installation-folder\script\sample

Table 4.44 shows the entries of an operation definition file, their contents, and the maximum value length (in bytes). Specification of these entries is required.

Table 4.44 Example: Contents and Maximum Value Length of Each Entry in an Operation Definition File When the Backup is of a File System

Specification item	Information to be specified	Maximum number of characters
BACKUP_OBJECT	▪ Specify a character string identifying the type of the backup object.. Specify FILESYSTEM.	32
DB_SERVER_NAME	▪ Specify the file system name. In a cluster configuration, specify the virtual server name for the cluster resource. This virtual server name must be defined in DRM_DB_PATH=*directory-on-the-shared-disk;virtual-server-name* in the Protection Manager configuration definition file (init.conf).	128
INSTANCE_NAME	Specify the mount point directory name or mount point directory list file name.	128
TARGET_NAME	Specify the file name or directory name to be backed up. This entry is valid only when backup is performed, and cannot be used when restoration is performed. You can specify multiple file names or directories, separated by commas. If you omit this value, backup is executed for each mount point.	1,024 [1]
FTP_HOME_DIR	Specify this value. Specifies the FTP service root directory. This value must be equal to the home directory name of the IIS FTP service on the backup server. You cannot omit this value.	128
FTP_SUB_DIR	Specify this value. Specifies the name of the subdirectory to be created under the FTP service root directory [2]. A temporary directory is created in the subdirectory specified by this setting. The extended commands transfer files to this subdirectory, via FTP. If this value is omitted, a subdirectory named HPtM will be automatically created.	128
SET_DRM_HOSTNAME	This entry specifies the file server configuration. When the file server has a cluster configuration, enter 1. When the file server does not have a cluster configuration, enter 0.	1

[1]Means 1,024 characters (either 1-byte or 2-byte characters), not 1,024 bytes.
The character count does not include the characters in the header `TARGET_NAME=`.
A comma that is used as a separator is also counted as a character.

[2]The specified subdirectory must satisfy the following:

- The subdirectory name must not contain a delimiter character (\).
- The subdirectory name must not be specified as only one period (.) or two periods (..).
- The root directory must not be specified.

4.15.6.3 Example of Creating Operation Definition Files in a Cluster Configuration

The following is an example of creating an operation definition file when the file server is a cluster system. This example assumes the system configuration shown in Figure 4.10.

The following prerequisites apply to this example:

- There are two file servers and a backup server to which a tape device is connected.
- A virtual server `VServer_C` is running on the file server A (server name: `FSServer_A`) and the file server B (server name: `FSServer_B`), which make up the cluster.
- `DRM_DB_PATH=`*`directory-name-on-shared-disk;virtual-server-name`* has been set in the Protection Manager configuration definition file `init.conf`.
- Protection Manager are permitted to create a temporary directory for extended commands in the dictionary map file directory.
- In the disk containing the automatically created temporary directory for extended commands, there is sufficient space to store the following temporary files:
  - Backup ID record file
  - Backup information file
- When a temporary directory for extended commands is created according to the naming conventions, the length of the directory name does not exceed the limit for the operating system.
- The IIS FTP service has been set up in the backup server (server name: `BKServer`) and is running. The home directory of the FTP site is `C:\FTP_ROOT`.
- The FTP subdirectory is `script`.
- File server A is specified as an active server and the file server B is specified as a standby server. The shared disk `G:` is used.
- The three mount points `D:`, `M:`, and `N:` exist on the file server A and file server B, the virtual server C (server name: `VServer_C`) has been defined as a cluster resource.
- The cluster resource in Table 4.45Table 4.45 has been registered in a cluster group `FSCG_1`.

Table 4.45 Cluster Resources in a Cluster Configuration (Backup of File System)

Cluster group name	virtual server	mount point

FSCG_1	VServer_C	D:
FSCG_1	VServer_C	M:,N:

- The mount points M: and N: have been set in the mount point directory target definition file app.txt so that the mount point directories are operated at one time.
- The storage directory for extended command temporary files is created on file server A, file server B, and the backup server (server name: BKServer) (see Table 4.46).

This example creates operation definition files corresponding to the operation IDs listed in Table 4.46.

Table 4.46 Operation IDs for Which an Operation Definition File is Created (Cluster Configuration, File System Backup)

Operation ID	Target file server name	Target mount point or file
Operation_A	VServer_C	Entire D:
Operation_B	VServer_C	Mount point directories specified in app.txt (M: and N:)

To create an operation definition file for a cluster configuration:

1. Create an operation definition file on the active server (file server A), and then set the required items.

   Create the following two files.

   - *Protection-Manager-installation-folder*\script\conf_Operation_A.dat
   - *Protection-Manager-installation-folder*\script\conf_Operation_B.dat

   The following shows an example of entries in the file:

   installation-destination-for-extended-commands\conf_Operation_A.dat.

```
BACKUP_OBJECT=FILESYSTEM
DB_SERVER_NAME=VServer_C
INSTANCE_NAME=D:
TARGET_NAME=
FTP_HOME_DIR=C:\FTP_ROOT
FTP_SUB_DIR=script
SET_DRM_HOSTNAME=1
```

   The following shows an example of entries in the file:

   installation-destination-for-extended-commands\conf_Operation_B.dat.

```
BACKUP_OBJECT=FILESYSTEM
DB_SERVER_NAME=VServer_C
INSTANCE_NAME=app.txt
TARGET_NAME=
FTP_HOME_DIR=C:\FTP_ROOT
FTP_SUB_DIR=script
SET_DRM_HOSTNAME=0
```

2. On the active server (file server A), execute EX_DRM_FS_DEF_CHECK, which is a check tool for operation definition files."EX_DRM_FS_DEF_CHECK" "*operation-ID*" -db

   The operation definition file is checked. In addition, a temporary directory for extended commands is automatically created.

   If an error occurs, correct the contents of the file, and then re-execute the check tool.

   Copy the operation definition files created on the active server (file server A) to the backup server. The copy destination is:

   Protection-Manager-installation-folder\script\conf\

3. On the backup server, execute the EX_DRM_FS_DEF_CHECK, which is a check tool for operation definition files.

   "EX_DRM_FS_DEF_CHECK" "*operation-ID*" -bk

   The operation definition file is checked. In addition, a temporary directory for extended commands is automatically created. If an error occurs, change the backup server configuration depending on the contents of the operation definition file, or change the contents of the operation definition file depending on the backup server configuration. When you change the backup server configuration, re-execute the checking tool. When you change the contents of the operation definition file, place the same file on the file server, and then re-execute the checking tool.

4. Copy the operation definition files created on the active server (file server A) to the standby server (file server B). The copy destination is:

   Protection-Manager-installation-folder\script\conf\

5. On the standby server (file server B), execute EX_DRM_FS_DEF_CHECK, which is a check tool for operation definition files.

   ```
   EX_DRM_FS_DEF_CHECK operation-ID -db
   ```

   The operation definition file is checked. In addition, a temporary directory for extended commands is automatically created.

4.15.7 Creating Operation Definition Files for SQL Server Database Backup

Create an operation definition file according to the configuration of the system using extended commands. After creating the operation definition files, store them in the following directory of each server where the extended commands will be executed:

Protection-Manager-installation-folder\script\conf\

After storing the operation definition file in the stipulated directory, execute the check tool for operation definition files to check the content of the operation definition file and automatically create a temporary directory for extended commands.

4.15.7.1 Deployment of Operation Definition Files

Create two identical operation definition files for the target SQL Server instance. Place one operation definition file on the database server and the other on the backup server. In a cluster configuration, the same operation definition file must be located on all the servers where the cluster resource (mount point or file) to be backed up is defined.

Figure 4.11 shows the locations of operation definition files for backups of a file system.

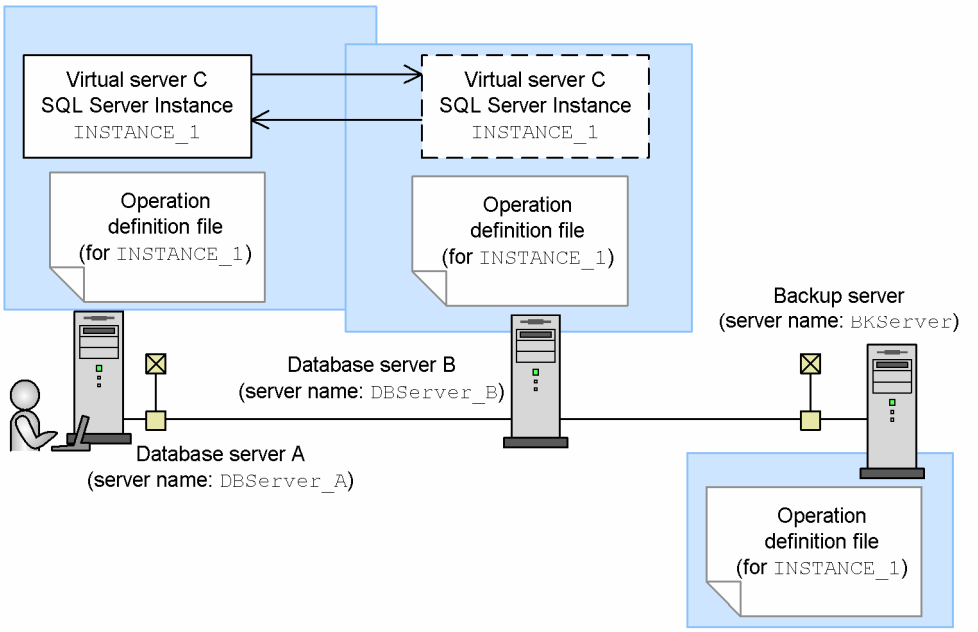


Figure 4.11 Example of Locations of Operation Definition Files When the Backup is of an Exchange Database

In this example, the virtual server (VServer_C) is running on the database server A (FSServer_A) and the database server B (FSServer_B), which make up the cluster.

The SQL Server instance INSTANCE_1 exists on the databases A and B, and the virtual server C has been defined as a cluster resource.

In this example, the operation definition file for the instance INSTANCE_1 is located on the database server A, the database server B, and the backup server (server name: BKServer) respectively.

4.15.7.2 Format of Operation Definition Files

An operation definition file is created, for each operation ID that uniquely identifies the processing target, with a name in the following format:

Protection-Manager-installation-destination\script\conf_*operation-ID*.dat

The sample file for the operation definition file is stored in the following location:

Protection-Manager-installation-folder\script\sample

Table 4.47 shows the entries of an operation definition file, the contents of them, and the maximum value length (in bytes). Specifying the entries is always required.

Table 4.47 Contents and Maximum Value Length of Each Entry in an Operation Definition File When the backup is of a SQL Server Database

Specification item	Information to be specified	Maximum number of characters
BACKUP_OBJECT	When the backup is of a file system: ▪ Specify a character string identifying the type of the backup object. ▪ Specify FILESYSTEM. When the backup is of a SQL Server database: ▪ This entry specifies a character string that indicates the backup object type. Specify MSSQL. When the backup is of an Exchange Server database: ▪ Specify a character string that indicates the type of backup object. Specify MSEXCHANGE.	32
DB_SERVER_NAME	Specify the database server name. In a cluster configuration, specify the name of the virtual server that corresponds to the cluster resource. This virtual server name must be defined in DRM_DB_PATH=*directory-on-the-shared-disk;virtual-server-name* in the Protection Manager configuration definition file (init.conf)	128
INSTANCE_NAME	This entry specifies a SQL Server instance name. If the name contains one or more space characters, enclose the entire name string in double quotation marks (“). When DEFAULT is specified, the SQL Server’s default instance name is used.	128
TARGET_NAME	This entry specifies the names of databases you want to back up. This entry is valid only when backup is performed, and cannot be used when restoration is performed. When you specify more than one name, use a comma (,) to separate each name. If the name contains one or more space characters, enclose the entire name string in double quotation marks (“). If you do not specify this entry, a backup is performed for each instance. When performing a backup that includes the system databases (master, model, and msdb), do not specify TARGET_NAME.	1,024[1]
FTP_HOME_DIR	Specifies the FTP service root directory. This value must be equal to the home directory name of the IIS FTP service on the backup server. You cannot omit this value.	128
FTP_SUB_DIR	Specifies the name of the subdirectory to be created under the FTP service root directory[2]. A temporary directory is created in the subdirectory specified by this setting. The extended commands transfer files to this subdirectory, via FTP. If this value is omitted, a subdirectory named HPtM will be automatically created.	128
SET_DRM_HOSTNAME	This entry specifies the database server configuration. When the file server has a cluster configuration, enter 1. When the file server does not have a cluster configuration, enter 0.	1

[1]Means 1,024 characters (either 1-byte or 2-byte characters), not 1,024 bytes. The character count does not include the characters in the header TARGET_NAME=. A comma that is used as a separator is also counted as a character.

[2]The specified subdirectory must satisfy the following:

- The subdirectory name must not contain a delimiter character (\).
- The subdirectory name must not be specified as only one period (.) or two periods (..).
- The root directory must not be specified.

4.15.7.3 Creating Operation Definition Files

The following gives an example of creating operation definition files for a system that has a cluster configuration.

This example assumes the following conditions:

- The virtual server (server name: VServer_C) is running on the database server A (server name: DBServer_A) and the database server B (server name: DBServer_B), which make up the cluster.
- The SQL Server instance INSTANCE_1 exists on the databases A and B, and the virtual server C (server name: VServer_C) has been defined as a cluster resource.
- There is a cluster group SQLCG_1 on the database server A and the database server B, and the cluster resource in Table 4.48 has been registered.

Table 4.48 Cluster Resources When the Backup is of a SQL Server Database

Cluster group name	Virtual server	SQL Server instance name	Extended command temporary file storage folder
SQLCG_1	VServer_C	INSTANCE_1	G:\tmp\INSTANCE_1

- The IIS FTP service has been set up on the backup server (server name: BKServer) and is running. The home directory of the FTP site is C:\FTP_ROOT.
- The FTP subdirectory is script.
- The database server A is specified as an active server and the database server B is specified as a standby server. The shared disk G: is used.
- The drmsqlinit command is executed for the INSTANCE_1.
- The value has been set in the DRM_DB_PATH=*folder-on-the-shared-disk;virtual-server-name* entry in the Protection Manager environment settings file init.conf.
- The extended command temporary file storage directory and VDI metafile storage directory have been created on database server A, database server B, and the backup server (see Table 4.49).
- The VDI metafile storage directory in the database server A or B is created as shown in Table 4.49.

- Protection Manager are permitted to create a temporary directory for extended commands in the dictionary map file directory.
- In the disk containing the automatically created temporary directory for extended commands, there is sufficient space to store the following temporary files:
  - Backup ID record file
  - Backup information file
  - SQL Server metafile
- When a temporary directory for extended commands is created automatically according to the naming conventions, the length of the directory name does not exceed the limit for the operating system.

Table 4.49 Example of VDI Metafile Storage Directory

Server name	VDI metafile storage directory
DBServer_A DBServer_B	G:\MSSQL\VServer_C\INSTANCE_1\META

In this example, you create an operation definition file for each of the operation IDs listed in Table 4.50.

Table 4.50 Operation IDs for Which to Create Operation Definition Files When the Backup Is of a SQL Server and in a Cluster Configuration

Operation ID	Target database server	Target instance and database
Operation_A	VServer_C	Entire INSTANCE_1

To create operation definition files (in a cluster configuration):

1. Create an operation definition file on the active server (database server A), and set the items you need. Create the file with the following name:

   Protection-Manager-installation-folder\script\conf\Operation_A.dat

   An example of entries in this file is as follows:

```
BACKUP_OBJECT=MSSQL
DB_SERVER_NAME=VServer_C
INSTANCE_NAME=INSTANCE_1
TARGET_NAME=
FTP_HOME_DIR=C:\FTP_ROOT
FTP_SUB_DIR=script
SET_DRM_HOSTNAME=1
```

2. On the active server (database server A), execute EX_DRM_SQL_DEF_CHECK, which is a check tool for operation definition files.

```
EX_DRM_SQL_DEF_CHECK operation-ID -db
```

   The operation definition file is checked. In addition, a temporary directory for extended commands is automatically created. If the tool detected an error, correct the contents of the file and then re-execute the check tool.

3. Copy the operation definition file you created on the active server (database server A) into the backup server. The copy destination is:

   Protection-Manager-installation-folder\script\conf\

4. On the backup server, execute EX_DRM_SQL_DEF_CHECK, which is a check tool for operation definition files.

   EX_DRM_SQL_DEF_CHECK *operation-ID* -bk

   The operation definition file is checked. In addition, a temporary directory for extended commands or VDI metafile storage directory is automatically created. If an error occurs, change the backup server configuration depending on the contents of the operation definition file, or change the contents of the operation definition file depending on the backup server configuration. When you change the backup server configuration, re-execute the check tool. When you change the contents of the operation definition file, place the same file on the database server, and then re-execute the check tool.

5. Copy the operation definition file you created on the active server (database server A) into the standby server (database B). The copy destination is:

   Protection-Manager-installation-folder\script\conf\

6. On the standby server (database server B), execute EX_DRM_SQL_DEF_CHECK, which is a check tool for operation definition files.

   EX_DRM_SQL_DEF_CHECK *operation-ID* -db

4.15.8 Creating Operation Definition Files for Exchange Database Backup

An operation definition file is created depending on the configuration of the system using extended commands. After you create the operation definition files, store them in the following directory of each server where the extended commands will be executed:

Protection-Manager-installation-folder\script\conf\

After storing the operation definition file in the stipulated directory, execute the check tool for operation definition files to check the content of the operation definition file and automatically create a temporary directory for extended commands.

4.15.8.1 Deployment of Operation Definition Files

Create two identical operation definition files for each target mount point, file, instance or storage group. Place one operation definition file on the file server or the database server and the other on the backup server. In a cluster configuration, the same operation definition file must be located on all the servers where the cluster resource (mount point or file) to be backed up is defined.

Figure 4.12 shows the locations of operation definition files.

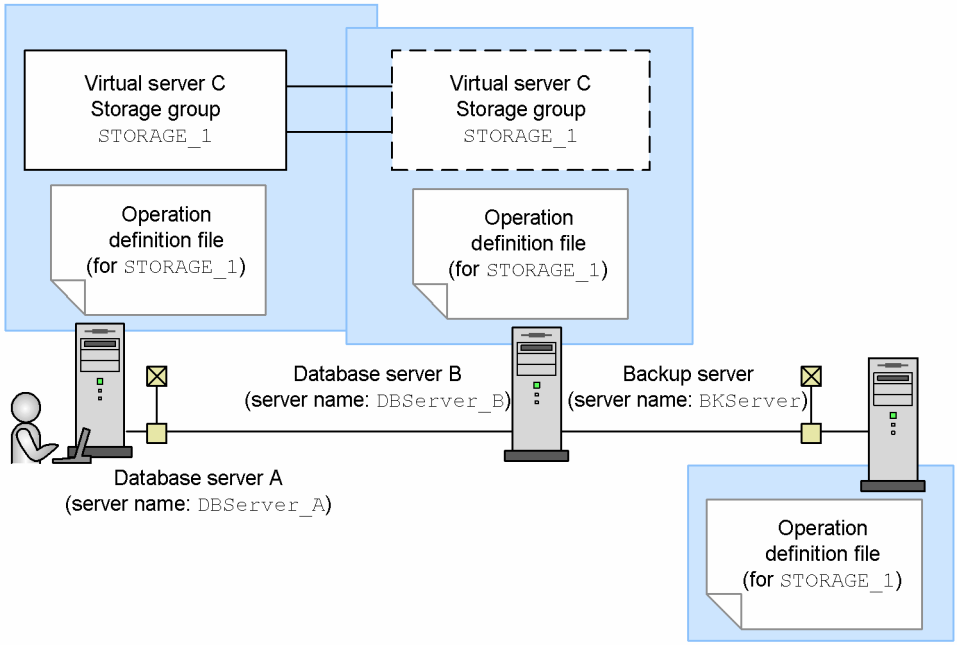


Figure 4.12 Example of Locations of Operation Definition Files (When the Backup is of an Exchange Database)

In this example, the virtual server (server name: VServer_C) is running on the database server A (server name: DBServer_A) and the database server B (server name: DBServer_B), which make up the cluster.

The storage group STORAGE_1 exists on the databases A and B, and the virtual server C has been defined as a cluster resource.

In this example, the operation definition file for the storage group STORAGE_1 is located on the database server A, the database server B, and the backup server (server name: BKServer) respectively.

4.15.8.2 Format of Operation Definition Files

An operation definition file is created, for each operation ID that uniquely identifies the processing target, with a name in the following format:

Protection-Manager-installation-destination\script\conf_*operation-ID*.dat

Protection-Manager-installation-folder\script\conf\

The sample file for the operation definition file is stored in the following location:

Protection-Manager-installation-folder\script\sample

Table 4.51 shows the entries of an operation definition file, their contents, and the maximum value length (in bytes). Specification of these entries is required.

Table 4.51 Specifications for Each Entry in an Operation Definition File When the Backup is of an Exchange Database

Specification Item	Information to be Specified	Maximum Number of Characters
BACKUP_OBJECT	Specify a character string that indicates the type of backup object. Specify MSEXCHANGE.	32
DB_SERVER_NAME	Specify the database server name. In a cluster configuration, specify the name of the virtual server that corresponds to the cluster resource. This virtual server name must be defined in DRM_DB_PATH=*directory-on-the-shared-disk;virtual-server-name* in the Protection Manager configuration definition file (init.conf).[1]	63
INSTANCE_NAME	Enter a hyphen (-).	1
TARGET_NAME	Specify the storage group name to be backed up.[2] You can specify multiple names, delimiting them with a comma. If the name contains one or more space characters, enclose the entire name string in double quotation marks (“). If you omit this value, all storage groups in the server are targeted for backup.	259[3]
FTP_HOME_DIR	Specifies the FTP service root directory. This value must be equal to the home directory name of the IIS FTP service on the backup server. You cannot omit this value.	128
FTP_SUB_DIR	Specifies the name of the subdirectory to be created under the FTP service root directory[4]. A temporary directory is created in the subdirectory specified by this setting. The extended commands transfer files to this subdirectory, via FTP. If this value is omitted, a subdirectory named HPtM will be automatically created.[1]	128
SET_DRM_HOSTNAME	This entry specifies the database server configuration. When the file server has a cluster configuration, enter 1. When the file server does not have a cluster configuration, enter 0.	1

[1] When the backup is of an Exchange database, you can also specify the virtual server name in the -hostname option of the extended command EX_DRM_EXG_BACKUP. If you specify the virtual server name in both DB_SERVER_NAME in the operation definition file and in the -hostname option, the value specified in the -hostname option has priority.

[2] When the backup is of an Exchange database, this specification is effective only if the target Exchange Server is an Exchange 2000 Enterprise Server, Exchange Server 2003 Enterprise Edition, or Exchange Server 2007. If the target Exchange Server is not an Exchange 2000 Enterprise Server, an Exchange Server 2003 Enterprise Edition, or Exchange Server 2007, any items you specify are ineffective. If the server is Exchange Server 2003 Enterprise Edition or Exchange Server 2007, do not specify a recovery storage group.

[3]Means 259 characters (either 1-byte or 2-byte characters), not 259 bytes. The character count does not include the characters in the header TARGET_NAME=. A comma that is used as a separator is also counted as a character.

[4]The specified subdirectory must satisfy the following:

- The subdirectory name must not contain a delimiter character (\).

- The subdirectory name must not be specified as only one period (.) or two periods (..).
- The root directory must not be specified.

4.15.8.3 Example of Creating Operation Definition Files

The following gives an example of creating an operation definition file in a cluster configuration.

The following prerequisites apply to this example:

- The Exchange Server used in the system is an Exchange 2000 Enterprise Server.
- On database A (server name: `DBServer_A`) and database B (server name: `DBServer_B`) making up the cluster, the virtual server C (server name: `VServer_C`) is running.
- The storage group `STORAGE_1` exists on databases A and B, and the virtual server C `VServer_C` has been defined as a cluster resource.
- There is a cluster group `EXGCG_1` on the database server A and the database server B.
- The cluster resource in Table 4.52 has been registered in a cluster group `EXGCG_1`.

Table 4.52 Cluster Resources When the Backup is of an Exchange Database

Cluster group name	Virtual server	Storage group name
EXGCG_1	VServer_C	STORAGE_1

- The IIS FTP service has been set up on the backup server (server name: `BKServer`) and is running. The home directory of the FTP site is `C:\FTP_ROOT`.
- The FTP subdirectory is `script`.
- The database server A is specified as an active server and the database server B is specified as a standby server. The shared disk `G:` is used.
- The value has been set in the `DRM_DB_PATH=`*`folder-on-the-shared-disk;virtual-server-name`* entry in the Protection Manager environment settings file `init.conf`.
- Protection Manager are permitted to create a temporary directory for extended commands in the dictionary map file directory.
- In the disk containing the automatically created temporary directory for extended commands, there is sufficient space to store the following temporary files:
  - Backup ID record file
  - Backup information file
- When a temporary directory for extended commands is created automatically according to the naming conventions, the length of the directory name does not exceed the limit for the OS. This example creates operation definition files corresponding to the operation ID listed in Table 4.53.

Table 4.53 Operation ID for Which a Definition File Is Created on Backup of an Exchange Database in Cluster Configuration

Operation ID	Virtual server	Target storage group
Operation_A	VServer_C	STORAGE_1

To create an operation definition file for a cluster configuration:

1. Create an operation definition file on the active server (database server A) on which the virtual server EVS_C is running and set required items. Create the file with the following name:

   Protection-Manager-installation-folder\script\conf_Operation_A.dat

   An example of specifying this file is:

```
BACKUP_OBJECT=MSEXCHANGE
DB_SERVER_NAME=VServer_C
INSTANCE_NAME=-
TARGET_NAME=
FTP_HOME_DIR=C:\FTP_ROOT
FTP_SUB_DIR=script
SET_DRM_HOSTNAME=1
```

2. On the active server (database server A), execute EX_DRM_EXG_DEF_CHECK, which is a check tool for operation definition files.

   "EX_DRM_EXG_DEF_CHECK" "*operation-ID*" -db

   The operation definition file is checked. In addition, a temporary directory for extended commands is automatically created.

   If an error occurs, correct the contents of the file and execute the check tool again.

3. Copy the operation definition file created on the active server (database A) to the backup server. Copy the file to the following folder:

   Protection-Manager-installation-folder\script\conf\

4. On the backup server, execute the EX_DRM_EXG_DEF_CHECK, which is a check tool for operation definition files.

   EX_DRM_EXG_DEF_CHECK *operation-ID* -bk

   The operation definition file is checked. In addition, a temporary directory for extended commands is automatically created. If an error occurs, change the backup server configuration depending on the contents of the operation definition file, or change the contents of the operation definition file depending on the backup server configuration. When you change the backup server configuration, re-execute the check tool. When you change the contents of the operation definition file, place the same file on the database server, and then re-execute the check tool.

5. Copy the operation definition file you created on the active server (database server A) into the standby server (database B). The copy destination is:

   Protection--Manager-installation-directory\script\conf\

4.15.9 Checking the Temporary Directory for Extended Commands

The temporary directory for extended commands is used when Protection Manager programs execute an extended command. When the check tool for operation definition file is executed, a temporary directory for extended commands is automatically created. It is important to make sure that a temporary directory for extended commands has been correctly created.

The temporary directory for extended commands is automatically created in the following locations:

directory-at-the-same-level-as-the-dictionary-map-file-storage-directory

The temporary directory for extended commands is automatically created according to naming conventions. Table 4.54 shows the conventions:

Table 4.54 Naming Conventions for the Temporary Directory for Extended Commands

File server or database server	Backup server
directory-at-the-same-level-as-the-dictionary-map-file-storage-directory[1]\script_work*operation-ID*\DB	*FTP_HOME_DIR-value**FTP SUB_DIR-value**operation-ID*\BK *FTP_HOME_DIR-value**FTP SUB_DIR-value**operation-ID*\AP

[1] For example, if the dictionary map file storage directory is L:\PTM, the temporary directory name is L:\script_work*operation-ID*\DB.

Temporary files created during execution of the extended command are:

- Backup ID record file
- Backup information file
- SQL Server's metafile (when the backup is of a SQL Server database)

4.15.10 Checking the Copy-Group List File

Certain extended commands allow you to specify a copy-group list file as an option value. A copy-group list file is a file that contains a list of multiple copy groups. When you specify a copy-group list file as an option value, before you start actual operations using the extended command, you need to execute the extended command EX_DRM_CG_DEF_CHECK to determine whether the copy-group list file has been set up correctly.

For information on what is checked in a copy-group list file, see the description on the EX_DRM_CG_DEF_CHECK command in the *HiCommand® Protection Manager Command Reference*.

For information on how to create a copy-group list file, see section 4.12.1.

4.15.11 Checking the FTP Service Before Backing up to Tape

Before extended commands can be executed for backup to a tape, settings for inter-server file transfer must be specified.

Make sure that the following operations are possible when the FTP service is active.

- Logon to the FTP server
- File transmission and reception using FTP for inter-file servers or between the database server and backup server

Make sure that inter-server FTP transmission and reception is possible between the directories indicated in Table 4.55.

Table 4.55 Directories Used for Inter-Server FTP Transmission and Reception

File Server or Database Server	Backup Server
dictionary-map-file-storage-directory\script_work*operation-ID*\DB (the temporary directory for extended commands)	*FTP_HOME_DIR-value**FTP_SUB_DIR-value**operation-ID*\BK (the temporary directory for extended commands)

4.16 Creating a User Script

By specifying a user script when executing the backup command, you can simplify the backup operation because the commands specified in the user script will execute during the backup operation.

You can enter the desired commands in a user script according to the coding rules for scripts. A user script file is a file that contains user scripts.

4.16.1 Overview of User Scripts

When you specify the `-script` option in the backup command, and then execute the backup operation, commands specified in the user script can execute the following processing:

- User pre-processing before the backup command is issued (`PRE_PROC`)
- User post-processing after the backup command has been issued (`POST_PROC`)
- User processing after pair resynchronization (`RESYNC_PROC`)[1]
- User processing after pair splitting (`SPLIT_PROC`)[1]
- User processing before the termination process (`FINISH_PROC`)[1]

[1]This processing is applicable only when you are backing up a SQL Server database.

The standard output and standard error output from the commands executed in the user script are output to the Protection Products standard output log (`drm_output.log`).

4.16.2 Coding Rules for User Scripts

Table 4.56 shows the coding rules for user scripts.

Table 4.56 Coding Rules for User Scripts

Applicable items	Rules
Overall user script	▪ The character code set must be ASCII. ▪ Codes lower than 0x20 cannot be used, except for CR (0x0d), LF (0x0a), and TAB (0x09). ▪ The linefeed code must be LF (0x0a) or CR+LF (0x0d and 0x0a). ▪ The length of a single line must not exceed 8 KB.
Item and section names	▪ Each line must contain only one item. ▪ Item and section names are not case-sensitive. ▪ Item and section names must consist of only one-byte characters. Two-byte characters cannot be used. ▪ Item names must be separated by a one-byte equal sign (=). One-byte space and tab characters can be inserted before and after equal signs.
Item values	▪ The value of an item must be specified after the one-byte equal sign (=) that follows the item name. Do not enter a linefeed code between an item's name and value. ▪ The item value begins after the one-byte equal sign (=) that follows the item name and ends at the linefeed code. ▪ If an item name is specified alone (without a value), an error results.
Comments	▪ A line that begins with a one-byte hash mark (#) is regarded as a comment line. ▪ If a one-byte hash mark (#) is entered anywhere on a line other than at the beginning of the line the part of the line following the hash mark will not be regarded as a comment. ▪ If a hash mark (#) on a line is preceded by no characters other than tab and space characters, the line will be regarded as a comment line.
Blank lines	▪ You can enter blank lines. ▪ A line that consists only of tab and one-byte space characters is regarded as a blank line.

4.16.2.1 Entries in User Scripts

Table 4.57 shows the entries specified in user scripts.

Table 4.57 Entries in User Scripts

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
LOCAL_BACKUP	Specifies whether to perform local backup: ▪ If you want to perform local backup, specify YES. ▪ If you do not want to perform local backup, specify NO. ▪ If local backup is specified, enter the entry before other entries. ▪ Specify the value by using one-byte character. ▪ If the backup target is an SQL Server database, you	Not allowed	Not allowed

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
	can specify YES or NO when you use a user script file by executing a basic command. Specify YES when you use a user script file by using an extended command. In this case, specifying NO results in an error. ▪ If the backup target is not a SQL Server database, specify YES. Even if you specify NO, the value is treated as YES.		
[PRE_PROC][2,11]	User processing section: Indicates the beginning of the "user pre-processing." In this section, define commands to be executed before the backup operation.	Not allowed	Allowed[3]
[RESYNC_PROC][2,9]	User processing section: Indicates the beginning of the "user processing for pair resynchronization." In this section, define commands to be executed as part of user processing for pair resynchronization.	Not allowed	Allowed[3]
[SPLIT_PROC][2,9]	User processing section: Indicates the beginning of the "user processing for pair splitting". In this section, define commands to be executed as part of user processing for pair splitting.	Not allowed	Allowed[3]
[FINISH_PROC][2,9]	User processing section: Indicates the beginning of the "user processing for termination". In this section, define commands to be executed as part of user processing for termination.	Not allowed	Allowed[3]
[POST_PROC][2,11]	User processing section: Indicates the beginning of the "user post-processing." In this section, define commands to be executed after the backup operation.	Not allowed	Allowed[3]
[CMD]	Indicates the beginning of a command definition section. ▪ This item must be specified within a user processing section. ▪ If two or more command definitions are included in your code, they will be executed individually in the order in which they are coded. ▪ When a value other than NOWAIT is specified in TIMEOUT, the next command does not start executing until the executing command finishes or times out (in other words, only one command executes at a time). ▪ When TIMEOUT=NOWAIT is specified, the subsequent command executes without waiting for the preceding command to finish. ▪ The item specified in the command definition section is valid only on the command definition section for the target command. The other command definition sections are not affected.	Allowed[4]	Allowed[3]
CMDLINE[5]	Specifies a command line to be executed. ▪ If you want to specify an option, separate the command name and the option with a one-byte space character. ▪ One command definition section must contain only one command line. ▪ A maximum of 2048 characters can be specified on a command line.[6]	Not allowed	Not allowed

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
	▪ A path or file name containing one or more spaces must be enclosed in double quotation marks ("). ▪ Command and file names must be specified using absolute path names. However, nested double quotation marks (") cannot be specified. ▪ Do not include any environment variable in a command name or command path name (for example, a specification like `%SystemRoot%\notepad.exe` is not allowed). ▪ If you want to use a shell command such as `dir`, execute it as a child process of `cmd.exe`. An example is `C:\WINNT\System32\cmd.exe /c dir`. ▪ To redirect processing, you must specify `C:\WINNT\System32\cmd.exe /c`. ▪ Specify a command that does not conflict with resources used by Protection Products. For example, when executing backup by using the remote copy function with the `-rc` and `-script` options specified, do not use the user script to perform an operation on the same remote copy pair. ▪ Protection Manager commands can be specified only in the `[POST_PROC]` section and in the `[PRE_PROC]` section of the script executed on the backup server. If a Protection Products command is specified in any other section, the command results in an error. ▪ If you specify `.exe`, `.com`, `.cmd`, or `.bat` as the file extension, the command line is executed as is. If you specify an extension other than the above, the command line is executed using the application associated with the extension (file type).#[10] ▪ Do not specify an executable file that will display a window or message and will make the system wait for a response.		
`ENV`[5]	Specifies the environment variables to be applied for execution of the specified command. ▪ Only one environment variable is allowed per line. ▪ The name and value of an environment variable must be separated by a one-byte equal sign (=). ▪ One environment variable definition can consist of up to 2048 characters.[6] ▪ If you omit the value for the environment variable, the environment variable is deleted. For example, the environment variable `ABC` is deleted when you specify `ENV=ABC=`. ▪ The ENV setting is valid only for the target command. Note that the subsequent command does not inherit the setting contents). ▪ Even if you specify an environment variable that contains a percent sign (%), the contents are not extracted. For example, if you specify `ENV=ABC=%PATH%`, the character string `%PATH%` is specified in the environment variable `ABC`.	Allowed[4]	Allowed

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
END_CODE[5]	Specifies the action to be performed in response to the return value of the executed command. Specifiable values are as follows: TERMINATE_NZ (default) Terminates script processing if a non-zero return value returns. IGNORE Continue processing whatever may be returned for a return value. Error threshold value Specify an integer in the range from 0 to 255. If a return value higher than the specified value is returned, script processing terminates. ▪ Specify the value using one-byte characters. ▪ If the executing command times out, processing terminates if TERMINATE_NZ or an error threshold value is specified, and processing continues if IGNORE is specified.	Not allowed	Allowed
TIMEOUT[5]	Specifies the command timeout value. Specifiable values are as follows: Timeout value (in seconds) [8] ▪ The range of specifiable values is from 0 to 86400. ▪ Use one-byte numeric characters only. ▪ No timeout will occur if 0 is specified here. NOWAIT Once a command starts successfully, there is no wait for the command to finish. With this specification, the command return value becomes 0. The standard output and standard error output from the command are not output to the Protection Products standard output log. ▪ The default is 600 (10 minutes).	Not allowed	Allowed
LOCATION[5]	Specifies the execution server of the specified command. Specifiable values are as follows: LOCAL (default) Executes the specified command on the local server. REMOTE Executes the specified command on the backup server specified with the -s option in the backup command. ▪ In the [RESYNC_PROC], [SPLIT_PROC], and [FINISH_PROC] sections, the setting is ignored (the value is fixed to LOCAL).	Not allowed	Allowed
PARENT_STAT[5]	Specifies whether the script can be executed based on the execution status[7] of the parent command (command that calls the script). Specifiable values are as follows: NORMAL (default) Executes the command specified in CMDLINE only if the execution status of the parent command is normal.	Not allowed	Allowed

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
	ERROR Executes the command specified in CMDLINE only if the execution status of the parent command is error. The execution result of the parent command becomes error regardless of the script result. ANY Always executes the command specified in CMDLINE. If the execution status of the parent command is error, the execution result of the parent command becomes error regardless of the script result. ▪ In the [RESYNC_PROC], [SPLIT_PROC], and [FINISH_PROC] sections, the setting is ignored (the value is fixed to NORMAL). ▪ In the [PRE_PROC] section, specify ANY or NORMAL. If you specify ERROR, the commands in the [PRE_PROC] section will not be executed.		

[1]Multiple specification is not allowed" means that multiple entries cannot be specified in the command definition section (CMD).

[2]The [PRE_PROC], [RESYNC_PROC], [SPLIT_PROC], [FINISH_PROC], and [POST_PROC] can be defined in any order.

[3]This item can be omitted if there is no command to be executed within the user processing section.

[4]This item can be used more than once within a user script file.

[5]Within the [CMD] item, CMDLINE, ENV, END_CODE, TIMEOUT, PARENT_STAT, and LOCATION can be defined in any order.

[6]For these characters, one-byte characters and two-byte characters are not distinguished.

[7]Within the same user processing section, the execution status of the parent command does not change even if the result of the preceding command is an error. For error processing of the command specified in CMDLINE, specify a script containing error processing in CMDLINE.

[8]Pay attention to the timeout value setting because if you specify TIMEOUT=0, the parent command will not terminate unless the command specified in CMDLINE terminates.

[9]The [RESYNC_PROC], [SPLIT_PROC], and [FINISH_PROC] specifications are valid only for backing up a SQL Server database. For any other type of database, the sections are ignored even if specified.

[10]Use the Edit File Type window of Explorer to check whether the application file name associated with the extension (file type) contains a space character. If it does, enclose the application file name with double quotation marks (`"`). For example, if the file name is `C:\Program Files\abc\abc.exe %1`, change it to `"C:\Program Files\abc\abc.exe" %1`.

[11]For details about the commands that can be specified, see 4.16.2.4.

4.16.2.2 Script Environment Variables

The table below shows the Protection Products script environment variables that can be referenced by commands in the user post-processing section. These script environment variables can be referenced from the local server and the backup server.

Table 4.58 Script Environment Variables

Environment variable	Description	Remarks
`DRMENV_L_BACKUPID`	Backup ID of local server	This environment variable is valid in the following cases: ▪ The command is being executed in the local server. ▪ The execution status of the parent command is normal.
`DRMENV_R_BACKUPID`	Backup ID of backup server	This environment variable is valid in the following cases: ▪ The command is being executed in the backup server. ▪ The execution status of the parent command is normal. ▪ The `-s` and `-auto_import` options are specified in the backup command.
`DRMENV_COMMENT`	Backup comment	Value specified for the `-comment` option in the backup command
`DRMENV_CMD_STAT`	Command execution status	`NORMAL`: Normal `ERROR`: Error

4.16.2.3 Command Execution Permission

Commands executed in the local server assume the execution permission of the parent command. Commands executed in the backup server assume the execution command of the user who started the Protection Manager service.

4.16.2.4 Commands that can be Specified in User Pre-processing and Post-processing

The table below lists the commands that can be specified in the user pre-processing section (`[PRE_PROC]`) and user post-processing section (`[POST_PROC]`). Only basic commands can be specified. The commands that can be specified differ for file and database servers and backup servers.

Note: Do not specify a command that is not listed. Doing so might prevent the system from running normally.

Table 4.59 Commands That can Be Specified in User Pre-processing and User Post-processing

Command name		[PRE_PROC]		[POST_PROC]	
		LOCATION= LOCAL	LOCATION= REMOTE	LOCATION= LOCAL	LOCATION= REMOTE
File system commands	drmfscat	Y	Y	Y	Y
	drmfsdisplay	Y[1]	N	Y	N
Common commands	drmcgctl	Y	Y	Y	Y
	drmhostinfo	Y	Y	Y	Y
Tape commands	drmmount	N	Y[2]	N	Y
	drmtapecat	N	Y	N	Y
	drmumount	N	Y	N	Y
SQL Server database commands	drmsqlcat	Y	Y	Y	Y
	drmsqldisplay	Y[1]	N	Y[1]	N
	drmsqlrestore	N	N	N	Y[3]
Exchange Server database commands	drmexgcat	Y	Y	Y	Y
	drmexgdisplay	Y[1]	N	Y[1]	N
	drmexgverify	N	N	N	Y

Legend:

Y: Can be specified.

N: Cannot be specified.

LOCATION=LOCAL: File server or database server

LOCATION=REMOTE: Backup server

[1] The -refresh option cannot be used.

[2] A secondary volume that was mounted must be unmounted within [PRE_PROC].

[3] The drmsqlrestore command can be specified only when data is being restored to a different instance. In this case, specify the -no_resync option.

4.16.3 Examples of User Script Usage

Table 4.60 provides references to the sections in which examples of user script usage are explained.

Table 4.60 Examples of User Script Usage

Usage example	Reference
Backing up a primary volume to tape via a secondary volume	6.4 7.4 8.4
Backing up a SQL Server database in a cascade configuration	7.12.2
Backing up a SQL Server database in a multi-target configuration	7.9
Enabling direct usage of a SQL database that was backed up to a secondary volume	7.17

4.17 Specifying Settings for Sending Mail

If you specify settings for sending mail, an e-mail can be sent to pre-registered addresses when an error occurs during processing of the Protection Products backup command.

You specify the settings for sending mail in the mail send definition file `mail.conf` of Protection Products.

`mail.conf` is found at the following location:

- *Protection Manager-installation-directory*`\conf`

`mail.conf` is a text file that has the same format as `init.conf`.

The character codes that can be used in the mail send definition file are limited to ASCII.

Table 4.61 mail.conf Parameters

Parameter	Description	Default (value after installation)
`MAIL_SEND`	Specifies whether or not e-mail is to be sent; if so, specifies the send condition. If you specify `NO`, e-mail is not sent. If you specify `ERROR`, e-mail is sent only when an error occurs. If you specify `ALWAYS`, e-mail is always sent for each execution of the command.	`NO`
`SMTP_SERVER`	Specifies the host name or IP address of the SMTP server to be used for sending the e-mail. Specify as a string of up to 64 one-byte characters.	`mailserver`
`TO`	Specifies an e-mail send destination address. Specify as a string of up to 255 one-byte characters. To send the e-mail to multiple addresses, specify multiple `TO` parameters, each on a separate line (you can specify up to 10 send destinations). Send destinations specified beyond the allowed number (10) are ignored.	`to_address@domain`
`SUBJECT_NORMAL`	Specifies the message subject line for an e-mail sent when the command terminates normally. Specify as a character string of up to 255 bytes.	`Protection Products Information`
`SUBJECT_ERROR`	Specifies the message subject line for an e-mail sent when the command terminates with an error. Specify as a character string of up to 255 bytes.	`Protection Products Error`
`FROM`	Specifies the sender's e-mail address. Specify as a string of up to 255 one-byte characters. This must be the address of an e-mail account that does not require authentication, which is prepared for mail sending.	`from_address@domain`
`TEXT`	Specifies the message text to be displayed in the head	`Protection Products`

Parameter	Description	Default (value after installation)
	of e-mail. Specify as a character string of up to 1024 bytes. Specify \n for a linefeed and \\ for a backslash (\) character (the one-byte \ used as the escape character must be included within the limit of 1024 bytes).	message:

The parameter specification format is explained below:

Each parameter consists of the parameter name, the equal sign (=), and the value to be set. Each parameter begins at the beginning of a line and ends at a linefeed. If space or tab characters are specified before or after the parameter name, the parameter is ignored. Parameter specifications are not case-sensitive.

Japanese can also be used in the subject line and the message text. The message text is set in the TEXT parameter and consists of the physical host name, the virtual server name (value of the DRM_HOSTNAME environment variable), the command line contents, and the command output contents (same contents as in drm_output.log).

An example of an e-mail message is shown below:

```
Protection Manager message:
Hostname=flora1 DRM_HOSTNAME=ExchangeServer1
2005/10/08 19:22:32(00003896)  drmexgbackup -mode vss
2005/10/08 19:22:32(00003896)  KAVX0001-I The drmexgbackup command will now start.
2005/10/08 19:22:33(00003896)  KAVX0256-I A connection to the Protection Manager
service will now be established.
                               Host name = 10.106.136.125
2005/10/08 19:22:33(00003896)  KAVX1600-I Backup processing will now start.
2005/10/08 19:22:58(00003896)  KAVX5108-I Resynchronizing a copy group.
                               Copy group name = EVS2
2005/10/08 19:24:44(00003896)  KAVX5033-E An attempt by the backup server to import the
VSS snapshot has failed.
                               Make sure that VSS has been set up correctly.
2005/10/08 19:27:01(00003896)  KAVX0002-I The drmexgbackup command will now end.
```

User authentication is not performed when e-mail is sent.

4.18 Backing Up a Protection Manager Operating Environment

Backup of the Protection Manager operating environment allows you to recover it in the event of an operating environment error. The Protection Manager operating environment includes directories that contain Protection Manager data and user-created files.

4.18.1 Files that Need to Be Backed Up

Use a tape backup product to back up the following directories and files while Protection Manager is not in operation.

Note: The following files and setting information are not subject to regular backup. Use a different method to back them up. Re-install or reconfigure after a recovery.

- Program files
- License information
- Windows Registry information
- System environment

4.18.1.1 Protection Manager Environment Setting Files and Log Files

After the Protection Manager environment has been set up or changed, back up files in the following directories:

- *Protection-Manager-installation-folder**conf*
- *Protection-Manager-installation-folder*\log[1]
- *Protection-Manager-installation-folder*\script\conf
- *Protection-Manager-installation-folder*\script\log1
- *Protection-Manager-Console-installation-directory*\conf[2]
- *Protection-Manager-Console-installation-directory*\log[2]

[1]Backing up this directory is optional. Back up this dictionary if you want to recover the log file.

[2]Back up if Protection Manager Console is started up locally. However, if it is started up remotely, the following management client files should be backed up.

- *Management-client-user-home-directory*\drm_console\conf
- *Management-client-user-home-directory*\drm_console\log[1]

4.18.1.2 Definition Files Created after the Installation of Protection Manager

Back up definition files that were created after installing Protection Manager. After the Protection Manager environment has been set up or changed, back up definition files.

- Copy-group list file: For details on the file name and location of the copy group list file, see section 4.12.1.
- Database list file or mount point directory list file: For details on the file names and locations of the database list files and mount point directory list files, see section 4.12.2.

4.18.1.3 Protection Products Backup Information

In order to restore data backed up by Protection Manager, the backup catalog and information on the DBMS (SQL Server metafile when the backup is of a SQL Server) are required. Data restoration using backup catalogs cannot be performed if the backup catalog is damaged. Also, data restoration cannot be performed if the DBMS information is damaged.

Therefore, the backup catalog and DBMS information must be protected as the backup information of Protection Manager.

Protect (backup) the backup catalog and DBMS information (SQL Server metafiles) after completing each normal backup.

Protecting Backup Information for Disk Backups

Backup information for disk backups is kept only on the disk. Therefore, if the disk is damaged, restoration of data backed up on the secondary volume cannot be performed. This can be prevented by protecting backup information from the file server or the database server.

In order to protect backup information, store the files to be protected in the temporary directory for extended commands, and then back up the files from the temporary directory onto a media such as tape. Figure 4.13 depicts protecting backup information for disk backups.

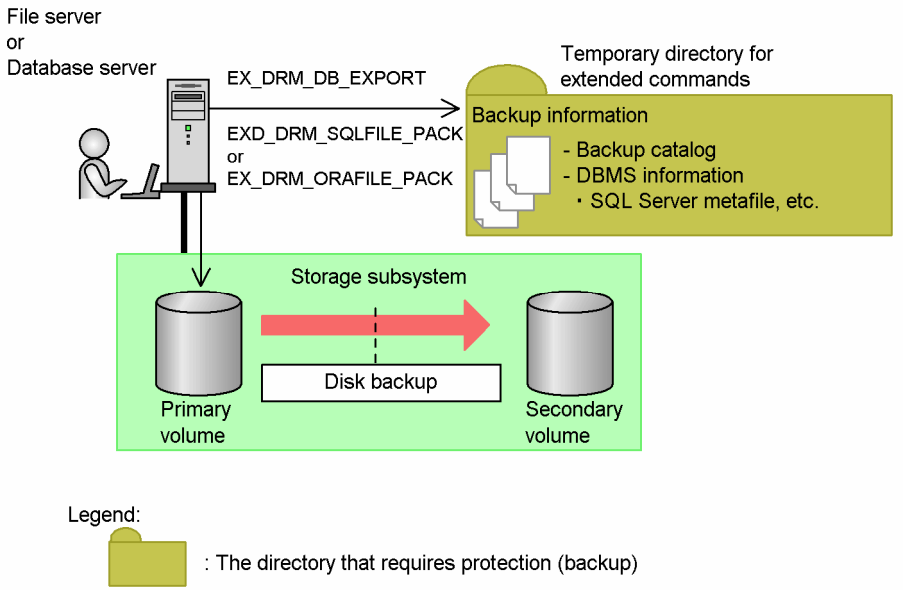


Figure 4.13 Protecting Backup Information for Disk Backups

To back up the backup information, perform the procedure for backing up a file system, SQL Server database, or Exchange database, and then back up the backup information.

Notes

When EX_DRM_DB_EXPORT is executed, one generation of backup information is stored in the temporary directory for extended commands. Only manage the number of secondary volume generations for backup information. Also, make sure that each generation is identifiable.

When the backup is of a file system:

Execute EX_DRM_FS_BACKUP, and then carry out the following procedure on the file server:

1. Execute EX_DRM_DB_EXPORT.

   Backup information is exported and recorded in the backup information file, which is in the temporary directory for extended commands.

2. Back up the files in the temporary directory for extended commands* to a location such as tape.

When the backup is of a SQL Server database:

Execute EX_DRM_SQL_BACKUP, and then carry out the following procedure on the database server:

If the output destination of the SQL Server metafile is configured to be the primary volume where the primary database of the backup target database is stored, it is not necessary to protect the SQL Server metafile. This is because the SQL Server metafile is also protected (backed up) during the backup process.

1. Execute EX_DRM_DB_EXPORT.

   Backup information is recorded in a file in the temporary directory for extended commands.

2. Execute EX_DRM_SQLFILE_PACK.

   DBMS information (SQL Server metafile) is stored in the temporary directory for extended commands.

3. Back up the files in the temporary directory for extended commands* to a location such as tape.

When the backup is of an Exchange database:

Execute EX_DRM_EXG_BACKUP, and then carry out the following procedure on the database server:

1. Execute EX_DRM_DB_EXPORT.

   Backup information is recorded in a file in the temporary directory for extended commands.

2. Back up the files in the temporary directory for extended commands* to a location such as tape.

*For details on the name of the temporary directory for extended commands on the file server or database server, see section 4.15.9.

Protecting Backup Information for Tape Backups

DBMS information (SQL Server metafile) is backed up on a tape for tape backups. However, the backup catalog is not actually backed up on the tape, but is stored on the disk only. Therefore, if the disk is damaged, restoration from the tape using the catalog cannot be performed This can be prevented by protecting the backup catalog from the backup server.

In order to protect the backup catalog, backup the dictionary map file directory including the backup catalog onto a media such as tape.

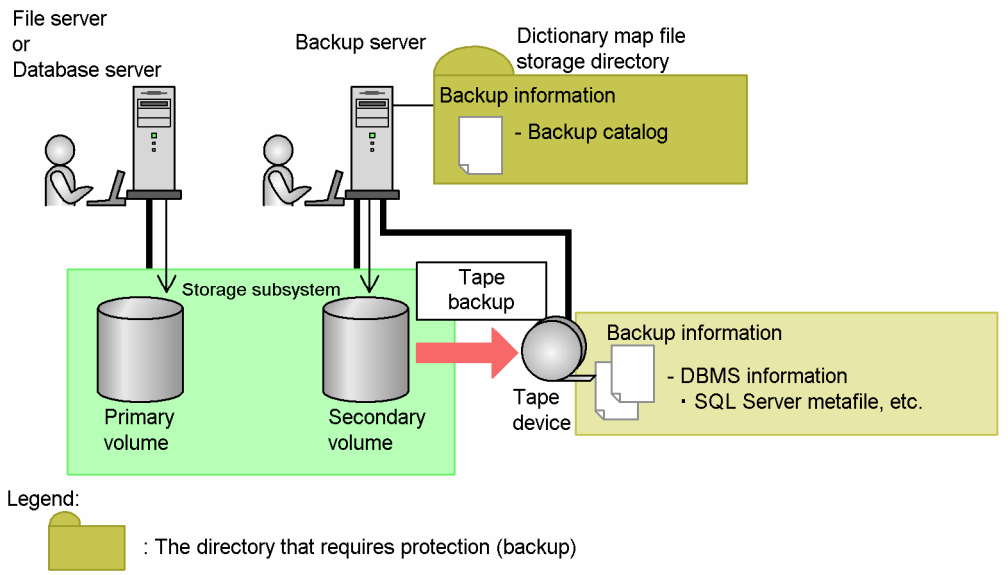


Figure 4.14 Protecting Backup Information for Tape Backups

Back up the tape, and then carry out the following procedure:

Execute `EX_DRM_TAPE_BACKUP`, and then carry out the following procedure on the backup server:

Use a tape backup product to back up all the files in the dictionary map file directory. For details on the dictionary map file directory on the backup server, see section 4.6.1.

4.18.2 Recovering the Operating Environment

In the event of an error in the Protection Manager operating environment, the environment can be recovered by restoring the backup described in section 4.18.1.

An error message is output when an error occurred in the Protection Manager operating environment. Restore necessary files according to the message. For details on errors in a dictionary map file, see section 9.6.

This section explains the following procedures:

- Procedure for recovering the operating environment (common procedure for all servers)
- Procedure for restoring backup information and data (for file, database, and backup servers)

4.18.2.1 Recovering Operating Environment (Common Procedure for All Servers)

To recover the operating environment:

1. Confirm the following items:
   - The Protection Manager programs are installed (including Service Packs)
   - The license is set
   - The Protection Manager programs are not running
2. Overwrite backup files that were protected (backed up) in 4.18.1.1 and 4.18.1.2 to their original location.

Operations from the time of backup can commence after the above processes have finished.

4.18.2.2 Recovering Backup Information (File Server)

This section explains how to recover backup information and restore the file system after restoring the operating environment when the backup information on the file server is damaged.

To recover backup information (file server):

1. From the backup data back up that was described up in section 4.18.1.3, determine the generation and backup information to be restored.
2. Delete all files in the temporary directory for extended commands on the file server.
3. Copy the protected (backed up) backup information into the corresponding temporary directory for extended commands.
4. Execute the following command:

   ```
   EX_DRM_DB_IMPORT
   ```

5. Use the procedure described in this manual to perform restoration.

4.18.2.3 Recovering Backup Information (Database Server)

To restore a file system, SQL Server database, Exchange database when backup information is damaged after the operating environment is recovered, carry out the following procedure on the file server or database server:

To recover backup information (database server):

1. From the backup data that was backed up in 4.18.1.3, decide the generation and backup information to be restored.
2. Delete all files in the temporary directory for extended commands on the database server.
3. Copy the protected (backed up) backup information into the corresponding temporary directory for extended commands.
4. Execute the following command:

   If a SQL Server database is to be restored:

   ```
   EX_DRM_DB_IMPORT
   EX_DRM_SQLFILE_EXTRACT
   ```

   If an Exchange database is to be restored:

   ```
   EX_DRM_DB_IMPORT
   ```

5. Use the procedure described in this manual to perform restoration.

4.18.2.4 Restoring Backup Information (Backup Server)

This section explains how to recover backup information and restore data from a tape after restoring the operating environment when the backup information on the backup server is damaged.

To recover backup information (backup server):

1. Determine the generation of the backup information to be restored among backup information backed up in section 4.18.1.3.
2. Delete all the files in the dictionary map storage directory on the backup server.
3. Copy the protected (backed up) backup information into the dictionary map storage directory on the backup server.
4. Carry out the procedure described in this manual for restoration.

Chapter 5 Using Protection Manager

Operation of Protection Manager differs depending on the object to be backed up.

This chapter provides an outline of and precautions on extended commands and basic commands used when operating Protection Manager. It also gives precautions common to all objects subject to backup .

- Protection Manager Commands (see section 5.1)
- Correspondence Between Extended Commands and Basic Commands (see section 5.2)
- Precautions on Operations (see section 5.3)
- Precautions When Executing Commands (see section 5.4)
- Starting and Stopping the Protection Manager Service (see section 5.5)

5.1 Protection Manager Commands

Protection Manager programs provide two types of commands: extended and basic. When operating Protection Manager programs, you must use extended commands and basic commands appropriately depending on circumstances and operation methods.

This section explains extended commands and basic commands.

5.1.1 Extended Commands

Extended commands are suited to operations for automating backup and restore processes.

Extended commands have the following characteristics:

- Command specifications can be simplified by defining, in advance, the resource information to be processed.
- In a series of commands, backup information can be inherited between commands.
- Backup information can be transferred easily between a database server and backup server.

For processing that cannot be automated because the user's judgment is required, use basic commands instead of extended commands. The following are examples of processing that cannot be automated:

- Environment setup
- Inspection of objects to be restored
- Recovery processing

Extended commands execute basic commands internally. For details about the correspondence between extended commands and basic commands, see section 5.2.

Extended commands are installed in the following locations:

Protection-Manager-installation-directory\script\bin\

5.1.2 Basic Commands

Basic commands provide data protection functions such as backup and restore, while extended commands execute basic commands internally for processing. Therefore, you need to have an understanding of basic commands before you use extended commands. If an error occurs during extended command operation, log information of the basic command being executed in the extended command is output.

5.2 Correspondence Between Extended Commands and Basic Commands

Table 5.1 lists the correspondence between extended commands for backing up and restoring file systems and basic commands for backing up and restoring file systems.

Table 5.1 Correspondence Between Extended Commands and Basic Commands (Used for Backing up and Restoring File Systems)

Extended command	Corresponding basic command	Function
EX_DRM_FS_BACKUP	drmfsbackup	Backs up a file system.
EX_DRM_FS_DEF_CHECK	Corresponding basic command does not exist	Checks the contents of the operation definition file, and automatically generates a temporary directory.
EX_DRM_FS_RESTORE	drmfsrestore	Restores a backed up file system to the primary volume.
Corresponding basic command does not exist	drmfscat	Displays a listing of the backup information for a file system.
Corresponding basic command does not exist	drmfsdisplay	▪ Displays a listing of file system information. ▪ Updates the dictionary map file to the latest status.

Table 5.2 shows the correspondence between extended common commands and basic common commands.

Table 5.2 Correspondence Between Extended Commands and Basic Commands (Common Commands)

Extended command	Corresponding basic command	Function
EX_DRM_BACKUPID_SET	Corresponding command does not exist	Generates a file for recording backup IDs.
EX_DRM_CG_DEF_CHECK	Corresponding command does not exist	Checks the contents of the copy-group batch definition file.
EX_DRM_DB_EXPORT	drmdbexport	Exports backup information to a file.
EX_DRM_DB_IMPORT	drmdbimport	Imports backup information from a file.
EX_DRM_FTP_GET	Corresponding command does not exist	Collects a backup information file from the backup server. If the backup is of a SQL Server database, the metafile is also collected.
EX_DRM_FTP_PUT	Corresponding command does not exist	Transfers a backup information file to the backup server. If the backup is of a SQL Server database, the metafile is also transferred.
EX_DRM_HOST_DEF_CHECK	Corresponding command does not exist	Checks the contents of the host environment setup file.
EX_DRM_RESYNC	drmresync	Resynchronizes a copy group.
Corresponding command does not exist	drmappcat	Displays catalog information on a host
Corresponding command does not exist	drmcgctl	▪ Locks copy groups. ▪ Unlocks copy groups. ▪ Displays a list of copy groups.

Extended command	Corresponding basic command	Function
Corresponding command does not exist	drmclusinit	Registers the parameters of the cluster product.
Corresponding command does not exist	drmdevctl	Enables or disables concealment of a device in the secondary volume.
Corresponding command does not exist	drmhostinfo	Displays a list of host information.
Corresponding command does not exist	drmlicense	Displays, adds, or updates license information.
Corresponding command does not exist	drmvssimport	Starts/stops the VSS import server.

Table 5.3 lists the correspondence between extended tape commands and basic tape commands.

Table 5.3 Correspondence Between Extended Commands and Basic Commands (Tape Commands)

Extended command	Corresponding basic command	Function
EX_DRM_CACHE_PURGE	drmmount drmumount	Clears the secondary volume cache
EX_DRM_MOUNT	drmmount	Mounts a secondary volume.
EX_DRM_TAPE_BACKUP	drmmount drmmediabackup drmtapebackup drmumount	Backs up secondary volume data to tape. If the backup is of a SQL Server database, the metafile is also backed up.
EX_DRM_TAPE_RESTORE	drmmount drmmediarestore drmtaperestore drmumount	Restores data from a tape to the secondary volume.
EX_DRM_UMOUNT	drmumount	Unmounts a secondary volume.
Corresponding command does not exist	drmtapecat	Displays a listing of backup information for a tape.
Corresponding command does not exist	drmtapeinit	Registers the parameters for a backup management product.

Table 5.4 lists the correspondence between extended utility commands and basic utility commands.

Table 5.4 Correspondence Between Extended and Basic Commands (Utility Commands)

Extended command	Corresponding basic command	Function
Corresponding command does not exist	drmdbconvert	Enable the latest version of Protection Manager programs to use backup catalogs of old versions.
Corresponding command does not exist	drmdbsetup	Initializes and deletes the database that stores the dictionary map file containing backup catalog information and mapping information for system resources.

Table 5.5 lists the correspondence between extended and basic commands for Protection Manager for SQL.

Table 5.5 Correspondence Between Extended and Basic Commands (Protection Manager for SQL)

Extended command	Corresponding basic command	Function
EX_DRM_SQL_BACKUP	drmsqlbackup	Backs up a SQL Server database.
EX_DRM_SQL_DEF_CHECK	Corresponding command does not exist	Checks the contents of the operation definition file, and automatically generates a temporary directory.
EX_DRM_SQL_RESTORE	drmsqlrestore	Restores a backed up SQL Server database to the primary volume.
EX_DRM_SQL_TLOG_BACKUP	drmsqllogbackup	Backs up a SQL Server transaction log.
EX_DRM_SQLFILE_EXTRACT	Corresponding command does not exist	Expands the SQL Server metafile into a folder to be backed up to tape.
EX_DRM_SQLFILE_PACK	Corresponding command does not exist	Saves the SQL Server metafile.
Corresponding command does not exist	drmsqlcat	Displays a listing of the backup information for a SQL Server database.
Corresponding command does not exist	drmsqldisplay	▪ Displays a listing of SQL Server database information. ▪ Updates the SQL Server database information in the dictionary map file.
Corresponding command does not exist	drmsqlinit	Registers SQL Server parameters.
Corresponding command does not exist	drmsqlrecover	Recovers a restored SQL Server database.
Corresponding command does not exist	drmsqlrecovertool	Recovers a restored SQL Server database via a GUI.

Table 5.6 lists the correspondence between extended and basic commands for Protection Manager for ExchangeServer.

Table 5.6 Correspondence Between Extended and Basic Commands (Protection Manager for Exchange)

Extended command	Corresponding basic command	Function
EX_DRM_EXG_BACKUP	drmexgbackup	Backs up a storage group.
EX_DRM_EXG_DEF_CHECK	Corresponding command does not exist	Checks the contents of the operation definition file, and automatically generates a temporary directory.
EX_DRM_EXG_RESTORE	drmexgrestore	Restores a storage group.
EX_DRM_EXG_VERIFY	drmmount drmexgverify drmumount	Verifies Exchange database integrity.
Corresponding command does not exist	drmexgcat	Displays a listing of the backup information for a storage group.
Corresponding command does not exist	drmexgdisplay	▪ Displays a listing of storage group information. ▪ Updates the storage group information in the dictionary map file.

5.3 Precautions for Operations

This section explains precautions on operations common to all objects subject to backup. For precautions on individual objects, see the following chapters:

- File system backup (see section 6.1)
- SQL Server database backup (see section 7.1)
- Exchange database backup (see section 8.1)

5.3.1 General Precautions

- When using Protection Manager in a cluster configuration, specify the same instance in the RAID Manager-linkage definition file (`DEFAULT.dat`) as the instance for the active server and standby server.
- Run Protection Manager programs after starting the backup server instance. If the backup server instance is not running, RAID Manager commands will terminate in an error.
- Do not use Protection Manager commands to change the configuration while a RAID Manager command is executing. Also, do not execute RAID Manager commands while Protection Products commands are executing.

  Note: Internally, Protection Products use the command line interface (CLI) function. of RAID Manager. When you are using commands provided by RAID Manager to change the status or the configuration of pair volumes, you must ensure that Protection Products are not operating.
- When you use RAID Manager commands to change the configuration of pair volumes, you must revise the settings so that the configuration matches the conditions for execution of Protection Products commands, and you must also update the dictionary map file.
- If you are rebooting a system that references the secondary volume, the secondary volume might be mounted automatically.

  If this happens, use the disk management utility provided by the OS to manually unmount the automatically mounted secondary volume to be used in backup and restore processing. This operation must be performed before you use Protection Products commands.

5.3.2 Backup and Restore Precautions

The backup information in the backup catalog is managed based on using the copy group as a key. Depending on the backup method, you might not be able to perform restoration, even though the backup data exists on the secondary volume.

The following is an example of two objects[1] stored in separate volumes:

- Database A (copy group: `vg01,obj01`)
- Database B (copy group: `vg01,obj01`)

Use the following backup method to back up the databases:

1. Back up database A and database B concurrently. The backup ID `0000000001` is created.

2. Back up database A only. The backup ID 0000000002 is created.

[1] The objects subject to backup are as follow:

- When the backup is of a file system: mount point
- When the backup is of a SQL Server database: database
- When the backup is of an Exchange Server database: storage group

Figure 5.1 shows the correspondence of the backup catalog and copy group.

Registration information of backup catalogue

Backup ID	Copy group	Backup information
0000000001	vg01, obj01	Object A
	vg01, obj02	Object B
0000000002	vg01, obj01	Object A

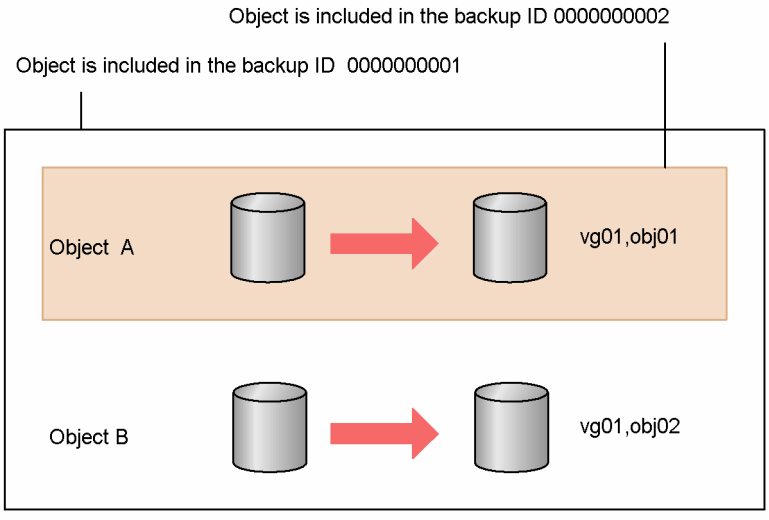


Figure 5.1 Correspondence of Backup Catalog and Copy Group

When you back up database A only (as described in step 2), the backup ID 0000000002 is created, and the old backup information (backup ID: 0000000001) about the copy group VG01,obj01 on database A is deleted from the backup catalog. In other words, since the backup information about database B, which is included in the backup ID 0000000001, is also deleted from the backup catalog, database B cannot be restored even though the backup data exists in the secondary volume. When you restore database B, restore the backup data obtained by using step 1 (backing up database A and database B concurrently) from tape to the secondary volume, and then restore the backup data from the secondary volume to the primary volume.

Note: When using QuickShadow, make sure to only back up a single generation. Backups for multiple generations might fail.

If Protection Products commands are executed on volumes that comprise a disk group managed by VERITAS Volume Manager, the error logs listed below will be output to the event log, but this does not indicate a problem. When Protection Products commands terminate normally, then when the disk group deport processing is executed again, you can verify that it was successful.

- Event ID in the Windows application event log: `10242`
- Event ID in the VERITAS Enterprise Administrator event log: `V-76-58645-10243`

5.3.3 Precautions for Backup

- The system cache of the secondary volume must be cleared before executing this command. Execute the `EX_DRM_CACHE_PURGE` command on the backup server to clear the system cache.
- When you execute a backup command for a copy group in the `PAIR` status, the status of the copy group is changed to `PSUS`. To put the copy group in the `PAIR` status, use the `drmresync` command.
- If you execute a backup command when information about the operation target object is not created in a dictionary map file, a dictionary map file is created by the command. In this case, the processing time required to create the dictionary map file is added to the total backup processing time.
- Make sure no other volume is mounted on the directory of the volume to be backed up when performing an online backup (see section 2.12.2).
- If there are multiple mount points specified for a single logical volume, as shown below, any backup will fail and an error message (`KAVX0006-E` and `DRM-10064`) will be output.
  - Two mount points (`M:\`, `M:\MNT`) are specified for `Harddisk1`.
  - Two mount points (`M:\`, `L:\MNT`) are specified for `Harddisk1`.
- If different logical volumes are mounted at mount points in a hierarchical relationship, such as `M:\` and `M:\MNT`, any mount points that hold a lower-level directory mount point (in this case, `M:\`) cannot be specified for backup. If specified, the backup will fail, and an error message (`KAVX0006-E` and `DRM-10062`) will be output.

  If this happens, only the lowest-level mount point can be specified for backup, so if you want to specify multiple mount points for backup (including directory mount points), create a configuration in which there is no hierarchal relationship between the mount points.
- For copy groups that have been backed up using VSS, do not perform operations that apply for a backup that does not use VSS.
  - For a single copy group, use only the set of operations that is appropriate, which depends on whether the backup was obtained by using VSS.
  - For copy groups that have been backed up by using VSS, if a backup is to be obtained that does not use VSS, you need to restart the backup server before you make the non-VSS backup.

- If you use Protection Products to execute a VSS backup, the backup operation might fail. If this happens, the status of the Windows Logical Disk Manager Administrative Service will remain stopped, and event logs that have event ID `17`, `1`, and `10010` will be output.

  The backup attempt fails because the Logical Disk Manager Administrative Service hangs up. If this happens, restart the backup server. After the restart, check the effect of the operation on the backup server.

5.3.4 Precautions for Restore Operations

- Make sure that the pair statuses of the copy group to be restored are `PSUS` (for primary volume) and `SSUS` (for secondary volume).
- Do not use the `drmresync` command while the `drmmediabackup` command is backing up data from tape to the secondary volume, while the `drmmediarestore` command is restoring data from the secondary volume to tape, or while the `drmmount` command is mounting the secondary volume.
- The following points should be considered regarding restoration when performing a configuration change by adding a volume mounted in an arbitrary directory on a drive.

  If a configuration change is made to mount a volume under `M:`, at `M:\mnt`, the `drmfsrestore` command is allowed to restore the backup information before the configuration change (back up only the volume mounted at `M:`). At this time, if restoration is executed with it mounted at `M:\mnt`, the information mounted at `M:\mnt` on the volume with `M:` as the mount point directory might be overwritten and the `M:\mnt` mount released.

  Be careful when restoring the backup information before the volume configuration change or when performing a restoration by specifying a portion of the mount points in the `-target` option of the `drmfsrestore` command.
- When you are restoring a backup obtained using NetBackup from a tape device, the Protection Products restore command will result in an error if there are differences in the mount point specification between the backup and the restoration, as shown below:

  Backup:

  > Make the backup from the secondary volume of the directory mount point specification.

  Restore:

  > Specify immediately under the drive letter as the mount point, and then perform the restoration.

  For the restore processing to succeed, you must, change the restore target to the directory mount point specification. To prevent this problem from occurring, specify immediately below the drive letter as the mount point, and then making the backup.
- When using QuickShadow, consider free space of the data pool when you perform restore operations.
- Check free space of the data pool regularly, and increase data pool capacity if necessary.
- For details on how to set the QuickShadow data pool, see the QuickShadow documentation.

- For details on how to restore data from a tape to a primary volume in a QuickShadow configuration, see Appendix F, *Restoring from a Tape Directly to a Primary Volume in a QuickShadow Configuration*.
- When restoration is executed in a Quick Restore environment, the volume maps on the primary volume and secondary volume are swapped. Therefore, if performance of the disks used for the primary volume and secondary volume are different, performance for the primary volume will change and restoration might cause deterioration in performance. In such a case, performance can be restored when restoration is performed again. For details on Quick Restore, see the manual for ShadowImage.

5.3.5 Precautions for Executing Commands Simultaneously

While a command is executing, if a user attempts to execute another command that cannot be executed concurrently with the executing command, the system will attempt to execute the latter command indefinitely until the former command terminates. To prevent the system from attempting to execute such a command indefinitely (until the former command terminates), edit the `init.conf` file. For details, Precautions for Executing Commands in a Cluster Environment

In a cluster environment, you need to set the environment variable `DRM_HOSTNAME` before executing a Protection Manager command. The following shows an example:

```
PROMPT> set DRM_HOSTNAME=virtual-server-name
```

If the backup target is an Exchange database, specify the name of the Exchange virtual server as the virtual server name.

When backup and restore are to be executed on Windows Server 2003 with a cluster configuration that uses VCS clusters, use the Windows Computer Management utility to ensure that the Windows Management Instrumentation service startup type is not set to disabled.

If the startup type is set to disabled, backup commands that have `cold` specified in the `mode` option, and restore commands that have the `resync` option specified, will result in an error with a message indicating that the cluster resource is offline.

5.3.6 Precautions for Using Backup Servers

5.3.6.1 Precautions for Mounting Secondary Volumes

The `drmmount` command stores information about the mounted secondary volume in the following file:

Protection-Manager-installation-directory\tmp\mntpt.dat

Do not delete this file while the secondary volume is being mounted by the `drmmount` command.

- When using the drmtapebackup command or drmtaperestore command, do not use the drmmount command to mount the secondary volume. If you execute the drmtapebackup command or drmtaperestore command while the secondary volume is being mounted by the drmmount command, the command will end with an error.
- When you execute the drmmount command with a backup ID specified, the copy group corresponding to the backup ID is locked. Such copy groups can be unlocked by executing the drmumount command with the same backup ID. As such, if you use the drmmount command to mount the secondary volume, make sure to use the drmumount command unmount the volume. If, for some reason, the volume cannot be unmounted by using the drmumount command, use operating system commands to unmount the volume, and then use the drmcgctl command to unlock the copy group corresponding to the backup ID specified using the drmumount command.
- If NetBackup is used with tape backup management products, the following points should be taken into consideration with respect to the drmmount command when the -mount_pt option is specified:
  - If the drmmediabackup command and the drmmediarestore command are used with an arbitrary directory specified in the drmmount command's -mount_pt option and the secondary volume is allocated to a mount point, Protection Products require that the Policy registered with NetBackup has the **cross mount point** checkbox set to **ON**.
  - If the drmmediabackup command and the drmmediarestore command are used with an arbitrary directory specified in the drmmount command's -mount_pt option, the -raw option cannot be specified in the drmmediabackup command.

    If the -raw option is used in the drmmediabackup command, either do not specify the -mount_pt option in the immediately preceding drmmount command, or else specify the drive letter in the -mount_pt option.
  - If an arbitrary directory is specified in the drmmount command's -mount_pt option and the backup data obtained using the drmmediabackup command is to be restored, the drmmount command executed before the drmmediarestore command must specify an arbitrary directory in the -mount_pt option.
- After Protection Manager are upgraded to version 3.5 or later, you may not be able to unmount a secondary volume that was mounted using a version earlier than 3.5. In such a case, use the Windows disk management program to unmount the volume, and then mount it again.
- After Protection Manager are upgraded to version 3.5 or later, you may not be able to mount a secondary volume that was unmounted using a version earlier than 3.5. In such a case, use the Windows disk management program to mount the volume, and then unmount it again.

5.3.6.2 When Mounting Data to a Backup Server

When Protection Manager performs tape backup, it mounts the secondary volume to the backup server. If you specify a mount point in the command arguments when performing tape backup, Protection Manager automatically uses the next available drive (in alphabetical order) after the specified drive excluding drive A and drive B. You cannot specify a drive other than the first drive for mounting data.

You will need to specify the drive to be used when mounting data to the backup server. Specify this drive in the configuration definition file for tape backup according to the same sequence as that used when Protection Manager automatically mounts data using a tape backup. For details on the configuration definition file for tape backup, see section 4.11.2.

Make sure to specify the same drive for use at backup and at restoration. If the drive used by Protection Manager at backup is used at restoration, restoration from the tape may fail.

5.3.6.3 When Importing Backup Information

If you use the `drmdbimport` command to import backup information, including the copy group of a mounted volume, while the secondary volume is mounted using the `drmmount` command, the import will fail. In this case, use the `drmumount` command to unmount the secondary volume, and then execute the `drmdbimport` command again.

5.3.6.4 Command Execution Conditions

If you execute the `drmmount` command, `drmmediabackup` command, or `drmmediarestore` command for a copy group in the `PAIR` status, a detailed message will be output indicating that the pair status of the copy group is invalid, and then the command terminates with an error.

In this case, execute the command after using the RAID Manager `pairsplit` command to set the pair status of the copy group to `SSUS`.

5.3.7 Precautions for Updating the Dictionary Map File

You must update the dictionary map file at the following times (regardless of whether the operation target is subject to backup):

- When the RAID Manager configuration definition files have been modified.
- When the volume pair configuration has changed.
- When a mount point (in Windows, a drive letter) has changed.
- When the disk configuration has changed because of addition or removal of a hard disk.
- When the `drmdbsetup` utility was executed to change the storage location of the dictionary map file.

- When a SQL Server instance has been created or deleted.
- When a SQL Server database has been added or deleted.
- When the name of an SQL Server database has been changed.
- When a SQL Server database configuration file has been added or deleted.
- When the name of SQL Server database configuration file has been changed.
- When a SQL Server database configuration file has been moved.
- When a dynamic volume has been expanded.

For details on the procedure for updating the dictionary map file, see section 4.13.

5.3.8 Tools That Cannot Be Used When Executing Commands

During execution of a Protection Manager command, if the primary or secondary volume is being used by another application, a problem might occur. For example, RAID Manager might not operate normally, a mount or unmount operation might fail, or a data inconsistency might occur. Other applications include:

- Disk management functions
- The `CHKDSK` command
- Disk defrag
- Performance logs
- Virus checking
- Programs that trace directories and files in a drive (such as index services or DLC services)
- Applications that manipulate disks (backup products other than Protection Manager)
- VERITAS Enterprise Administrator

These applications are sometimes executed in the background by a service. Adjust the starting times for these applications so that they will not conflict with Protection Manager commands.

The following points should be taken into consideration with respect to the Windows index service (`cisvc.exe`):

- When you execute Protection Products commands, check that the master binding of the index service is not operating. File access by the index service can conflict with file operations from Protection Products commands or from products to which Protection Products are linked, which can cause an error in the index service master binding or in the Protection Products command.
- Do not use Protection Products to back up and restore the volume that holds the index service catalog. Because the Protection Products restore command also causes the index service catalog to be restored at the same time, there is a possibility that operations unexpected by the index service will be performed.

If Provisioning Manager is also being used, the following should be noted:

- On hosts that use a Windows OS, do not perform Protection Products operations while a host setting operation or a host refresh operation is being performed or for about 10 minutes following the operation. Otherwise, the Protection Products commands might result in an error If this happens, wait at least 10 minutes after attempting the Provisioning Manager operation before performing the Protection Products operation again.

5.3.9 Disk Group Configuration That Does Not Allow Backup (Dynamic Disk Configuration)

In a dynamic disk configuration, you cannot back up a file system or database if, as shown in the following figure, there is a logical volume that has not been mounted in the disk group in which the target file system or database was created. Figure 5.2 shows this condition.

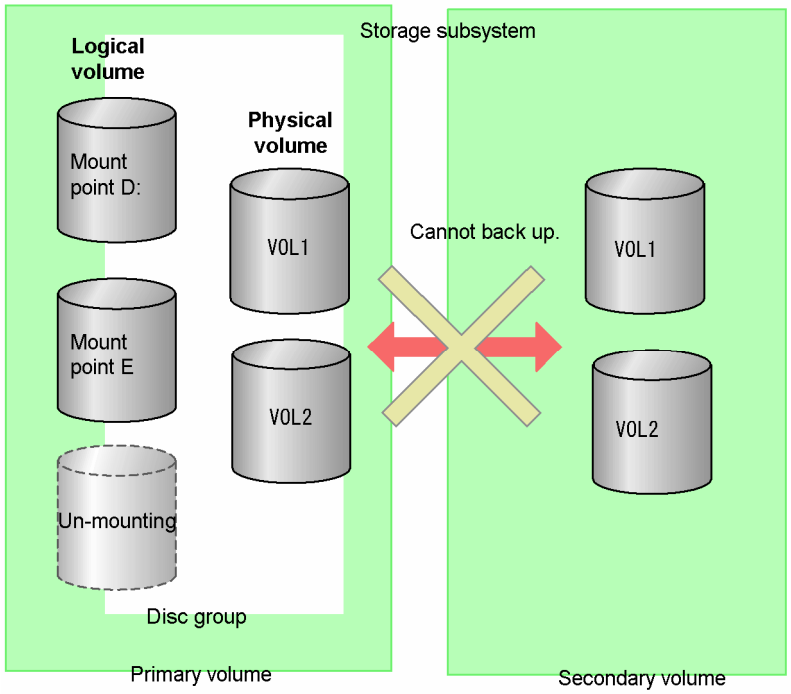


Figure 5.2 Disk Group Configuration that Does Not Allow Backup

In such a case, change the configuration to one of the following volume configurations:

- Delete the logical volume that has not been mounted, from the disk group.
- Mount the logical volume onto the file system so that the file system is eligible to be backed up.

For details on the relationship between file systems that can be backed up and the configuration of the disk group, see section 2.12.1.

After you have changed the volume configuration, execute the `drmfsdisplay` command with the `-refresh` option specified to update the dictionary map file.

5.3.10 About Pair Volume Resynchronization in a Cascade Configuration

When you are performing pair volume resynchronization in a cascade configuration (a multi-layer pair configuration), the status of the pair that follows the pair to be resynchronized (secondary volume/primary volume and secondary volume) must be PSUS or SMPL.

If you perform a pair status check by using a backup, restore, or resynchronization command when the status of the pair that follows the pair to be resynchronized is not PSUS or SMPL, the command will result in an error and a message will be output.

5.4 Precautions When Executing Commands

This section lists precautions to observe when executing commands.

5.4.1 Permissions Required for Users Executing Commands

The following permissions are required for executing Protection Manager commands:

- Administrator permissions for the OS: A user executing a Protection Manager command must have local Administrator authorization on the server on which the command is to be executed.
- Database access permissions
  - When the backup target is a SQL Server database

    When executing a Protection Manager command, the user accesses the SQL Server using Windows authentication. Therefore, users who execute Protection Manager commands should be registered as a member of the sysadmin fixed server role of the SQL Server.
  - When the backup target is an Exchange database

    Users who execute Protection Products commands should be members of the Enterprise Admins group or the Exchange Domain Servers group of the domain.

Table 5.7 Required Permissions for Commands Used to Back up and Restore a File System

Extended command name	OS administrator permissions	Database access permissions
EX_DRM_FS_BACKUP	Needed	Not Needed
EX_DRM_FS_DEF_CHECK	Needed	Not Needed
EX_DRM_FS_RESTORE	Needed	Not Needed
drmfsbackup	Needed	Not Needed
drmfsrestore	Needed	Not Needed
Drmfscat	Needed	Not Needed
drmfsdisplay	Needed	Not Needed

Table 5.8 Required Permissions for Common System Commands

Extended command name	OS administrator permissions	Database access permissions
EX_DRM_BACKUPID_SET	Needed	Not Needed
EX_DRM_CG_DEF_CHECK	Needed	Not Needed
EX_DRM_DB_EXPORT	Needed	Not Needed
EX_DRM_DB_IMPORT	Needed	Not Needed
EX_DRM_FTP_GET	Needed*	Not Needed
EX_DRM_FTP_PUT	Needed*	Not Needed
EX_DRM_HOST_DEF_CHECK	Needed	Not Needed
EX_DRM_RESYNC	Needed	Not Needed
drmappcat	Needed	Not Needed
drmcgctl	Needed	Not Needed
drmclusinit	Needed	Not Needed
drmdbexport	Needed	Not Needed
drmdbimport	Needed	Not Needed
drmdevctl	Needed	Not Needed
drmhostinfo	Needed	Not Needed
drmlicense	Needed	Not Needed
drmresync	Needed	Not Needed
drmvssimport	Needed	Not Needed

*An FTP user requires OS administrator permissions on the backup server.

Table 5.9 Required Permissions for Tape System Commands

Extended command name	OS administrator permissions	Database access permissions
EX_DRM_CACHE_PURGE	Needed	Not Needed
EX_DRM_MOUNT	Needed	Not Needed
EX_DRM_TAPE_BACKUP	Needed	Not Needed
EX_DRM_TAPE_RESTORE	Needed	Not Needed
EX_DRM_UMOUNT	Needed	Not Needed
drmmediabackup	Needed	Not Needed
drmmediarestore	Needed	Not Needed
drmmount	Needed	Not Needed
drmtapebackup	Needed	Not Needed
drmtapecat	Needed	Not Needed
drmtapeinit	Needed	Not Needed
drmtaperestore	Needed	Not Needed
drmumount	Needed	Not Needed

Table 5.10 Required Permissions for Utility Commands

Extended command name	OS administrator permissions	Database access permissions
drmdbconvert	Needed	Not Needed
drmdbsetup	Needed	Not Needed

Table 5.11 Required Permissions for Commands Used to Back up a SQL Server Database

Extended command name	OS administrator permissions	Database access permissions
EX_DRM_SQL_BACKUP	Needed	Needed
EX_DRM_SQL_DEF_CHECK	Needed	Needed
EX_DRM_SQL_RESTORE	Needed	Needed
EX_DRM_SQL_TLOG_BACKUP	Needed	Needed
EX_DRM_SQLFILE_EXTRACT	Needed	Not Needed
EX_DRM_SQLFILE_PACK	Needed	Not Needed
drmsqlbackup	Needed	Needed
drmsqlcat	Needed	Not Needed
drmsqldisplay	Needed	Needed
drmsqlinit	Needed	Needed
drmsqllogbackup	Needed	Needed
drmsqlrecover	Needed	Needed
drmsqlrecovertool	Needed	Needed
drmsqlrestore	Needed	Needed

Table 5.12 Required Permissions for Commands Used to Back up an Exchange Database

Extended command name	OS administrator permissions	Database access permissions
EX_DRM_EXG_BACKUP	Needed	Needed
EX_DRM_EXG_DEF_CHECK	Needed	Needed
EX_DRM_EXG_RESTORE	Needed	Needed
EX_DRM_EXG_VERIFY	Needed	Not Needed
drmexgbackup	Needed	Needed
drmexgcat	Needed	Not Needed
drmexgdisplay	Needed	Needed
drmexgrestore	Needed	Needed
drmexgverify	Needed	Not Needed

5.4.2 Concurrent Execution of Commands

Parallel execution (parallelize, then execute) of commands is not an available Protection Manager feature. Commands may only be executed after preceding commands have finished. The same applies when there are multiple virtual servers in a cluster configuration. In addition, the same applies to basic commands that are executed when an extended command is executed. Therefore, it is necessary to consider execution times when executing commands.

Table 5.13 indicates which commands can and can not be run concurrently with other commands when backing up and restoring a file system.

Table 5.13 Concurrent Command Execution When Backing up and Restoring a File System

Command	Function summary	Possible to run with other commands?
`drmfsbackup`	Backs up a file system to a disk.	No
`drmfscat`	Displays a listing of the backup information for a file system.	Yes
`drmfsdisplay`	Displays a listing of file system information. Updates the dictionary map file to the latest status.	No[1]
`drmfsrestore`	Restores a file system.	No

[1]This command cannot be executed with other commands when the `-refresh` option is specified. This command can be executed with other commands when other options are specified.

Table 5.14 indicates which common commands can and can not be run concurrently.

Table 5.14 Concurrent Execution of Common Commands

Command	Function summary	Possible to run with other commands?
drmappcat	Displays catalog information on a host.	Yes[1]
drmcgctl	▪ Locks copy groups. ▪ Unlocks copy groups. ▪ Displays a list of copy groups.	No[2]
drmclusinit	Registers the parameters of the cluster product.	Yes
drmdevctl	Enables or disables concealment of a device in the secondary volume.	No
drmdbexport	Exports backup information to a file.	No
drmdbimport	Imports backup information from a file.	No
drmhostinfo	Displays a list of host information.	Yes
drmlicense	Displays, adds, or updates license information.	Yes
drmresync	Resynchronizes copy groups and deletes the backup data.	No
drmvssimport	Starts/stops the VSS import server.	Yes

[1] When these commands are executed with the -delete option, these commands cannot execute concurrently with other commands. When these commands are executed with options other than the -delete option, these commands can execute concurrently with any other command.

[2] You can execute other commands simultaneously with the copy group display command.

Table 5.15 indicates which tape-related commands can and can not be run concurrently with other commands.

Table 5.15 Concurrent Execution of Tape-Related Commands

Command	Function summary	Possible to run with other commands?
drmmediabackup	Backs up backup data to tape.	Yes[1]
drmmediarestore	Restores backup data from tape.	Yes[1]
drmmount	Mounts a disk volume.	No
drmtapebackup	Copies the backup data to tape.	No
drmtapecat	Displays a listing of backup information for a tape.	Yes[2]
drmtapeinit	Registers the parameters for a backup management product.	Yes
drmtaperestore	Restores data from a backup on tape.	No
drmumount	Unmounts a disk volume.	No

[1] These commands can execute concurrently with any other command only when the backup management product is running.

[2] When these commands are executed with the `-delete` option, they cannot execute concurrently with other commands. When these commands are executed with options other than the `-delete` option, they can execute concurrently with any other command.

Table 5.16 indicates which utility commands can and can not be run concurrently with other commands.

Table 5.16 Concurrent Execution of Utility Commands

Command	Function summary	Possible to run with other commands?
drmdbconvert	Enables the latest version of Protection Manager to use backup catalogs created by an old version of Protection Manager.	No
drmdbsetup	Creates or deletes a Protection Manager database.	No

Table 5.17 indicates which commands can and can not be run concurrently when backing up a SQL Server database.

Table 5.17 Concurrent Command Execution When Backing up a SQL Server Database

Command	Function summary	Possible to run with other commands?
drmsqlbackup	Backs up a SQL Server database to a disk.	No
drmsqlcat	Displays a listing of the backup information for a SQL Server database.	Yes
drmsqldisplay	Displays a listing of SQL Server database information. Updates the SQL Server database information in the dictionary map file.	Yes[1]
drmsqlinit	Registers SQL Server parameters.	Yes
drmsqllogbackup	Backs up the SQL Server transaction log file.	No
drmsqlrecover	Recovers a restored SQL Server database.	No
drmsqlrecovertool	Recovers a restored SQL Server database via a GUI.	No
drmsqlrestore	Restores a SQL Server database.	No

[1] This command cannot be executed with other commands when the -refresh or -remote option is specified. This command can be executed with other commands when other options are specified.

Table 5.18 indicates which commands can and can not be run concurrently when backing up an Exchange database.

Table 5.18 Concurrent Command Execution When Backing up an Exchange Database

Command	Function summary	Possible to run with other commands?
rmexgbackup	Backs up a storage group to a disk.	No
drmexgcat	Displays a listing of the backup information for a storage group.	Yes
drmexgdisplay	Displays a listing of storage group information. Updates the storage group information in the dictionary map file.	Yes[1]
drmexgrestore	Restores a storage group.	No
drmexgverify	Verifies the integrity between a storage group and backup information.	No

[1] This command cannot be executed with other commands when the -refresh option is specified. This command can be executed with other commands when other options are specified.

5.4.3 Retry Time When an Error Occurs

When an error occurs, the system retries the command. Therefore, command execution may take some time (the additional time needed to retry the command). The command retry interval and the number of times to retry depend on the values set in the *xxx*_RETRY_TIME and *xxx*_RETRY_WAIT parameters in the Protection Manager configuration definition file (init.conf) and RAID Manager-linkage definition file (DEFAULT.dat). Depending on the values set in these parameters, more than the usual amount of time may be required for executing a command when an error occurs.

5.4.4 Stopping Retry Operations Caused by Commands That Cannot Execute Concurrently

While a command is executing, if a user attempts to execute another command that cannot be executed concurrently with the executing command, the system will attempt to execute the latter command indefinitely until the former command terminates. To prevent the system from attempting to execute such a command, you need to edit the Protection Products configuration definition file (init.conf), which is stored in the following location:

Protection-Products-installation-directory\conf\init.conf

init.conf file format

```
COM_RETRY_TIME=number-of-retry-attempts
COM_RETRY_WAIT=duration-between-retry-attempts
```

The following values can be specified:

- Number of retry attempts: 0 to 3,600, with 0 representing indefinite retries. If you want to stop indefinite retries, specify a value other than 0.
- Duration between retry attempts: 0 to 3,600 between each retry attempt (in ten-second intervals)

In a cluster configuration of the active-active type, commands that cannot be executed concurrently on a system might be executed at the same time if failover occurs due to system failure. If commands that cannot execute concurrently are executed, the second command is retried indefinitely until the first command finishes. If you want to stop the indefinite retry, edit the Protection Manager configuration definition file (init.conf).

5.4.5 Executing Multiple drmmediabackup or drmmediarestore Commands Concurrently

If you want to execute multiple `drmmediabackup` commands or `drmmediarestore` commands concurrently, in the backup server specify the retry count and retry period for the command. For more details, see section 5.4.4.

When the `drmmediabackup` or `drmmediarestore` command is executed, processing that cannot execute the same command more than once at the same time is performed before and after the backup management product starts. If you attempt to execute the same command during this processing, the system retries execution of the activated command until the backup management product starts. Therefore, when you want to execute multiple `drmmediabackup` commands or `drmmediarestore` commands at the same time, specify a sufficient retry period. If the retry period is not sufficient, retries of the activated command may exceed the retry period, and an error will occur.

To specify the retry period, specify a period longer than the following formula:

(the period from the time the `drmmediabackup` command or the `drmmediarestore` command is executed to the time the backup management product starts) ×(the number of commands executed concurrently)

If you attempt to execute multiple extended commands (`EX_DRM_TAPE_BACKUP` or `EX_DRM_TAPE_RESTORE`), more than one instance of the same command cannot run concurrently during the period from the time the `drmmount` command is executed by the extended command to the time the backup management product is started by the `drmmediabackup` or the `drmmediarestore` command. Therefore, when you want to execute multiple extended commands (`EX_DRM_TAPE_BACKUP` or `EX_DRM_TAPE_RESTORE`), specify a sufficient retry period. If the retry period is not sufficient, retries of the activated command may exceed the retry period, and an error will occur.

To specify a retry period, specify a period longer than the following formula:

(the period from the time `EX_DRM_TAPE_BACKUP` or `EX_DRM_TAPE_RESTORE` is executed to the time the backup management product starts) x (the number of commands executed concurrently)

5.5 Starting and Stopping the Protection Manager Service

If the OS is Windows, the Protection Manager service is registered automatically with the Windows system when you install Protection Manager Copy Controller.

The account of the user who executes the service must have local Administrator permissions and the right to log on as a service.

Note: The user accounts of the users who will execute the Protection Manager service must be specified during installation of Protection Manager Copy Controller.

When the functions listed in the table below are used, the Protection Manager service must be running on a file server, database server, or backup server.

Table 5.19 Functions that Require that the Protection Manager Service Be Running

Function	Servers
System configuration check	FS, DB, BK
Check for existence of files used by the Setup GUI	FS, DB
Check for existence of user script files	FS, DB
Execution of commands listed in user script files	BK
Receive and import backup catalogs forwarded from another server	BK
Compatibility checks before executing a backup	FS, DB
Check of the secondary volume status during backup	BK
Backup using VSS (VSS snapshot preparation and import, writing of metadata to the secondary volume, Exchange database verification)	BK
Restore from a backup using VSS of an Exchange database to the primary volume (reading metadata from the secondary volume)	BK

Legend:

FS: File Server, DB: Database Server, BK: Backup Server

5.5.1 Starting the Protection Manager Service

The Protection Manager service starts automatically when the Windows system starts on the application server where Protection Manager Copy Controller is installed. This occurs because the startup type of the Protection Manager service is set to **Automatic** during installation.

5.5.2 Restarting the Protection Manager Service

When the contents of the Protection Products' `init.conf` and `default.dat` definition files are changed, you must use the Services window of **Computer Management** in Windows to restart the Protection Manager service. However, if the Setup GUI is used to change the contents of these definition files, the service will restart automatically.

5.5.3 Stopping the Protection Manager Service

When the Protection Manager Copy Controller is uninstalled, the Protection Manager service stops and is deleted from Windows.

Chapter 6 Using Protection Manager With File Systems

This chapter describes procedures for running Protection Manager when backing up a file system. For details on setting up commands, see the *HiCommand® Protection Manager Command Reference* manual.

- Operations for Backing up and Restoring a File System (see section 6.1)
- Backing up and Restoring Data Between Volumes (see section 6.2)
- Backing up and Restoring by Using a Tape Device (see section 6.3)
- Backing up a File System by Using a User Script (see section 6.4)
- Backup and Restore Between Remote Sites (Remote Copy) (see section 6.5)
- Using a Backup Management Product That Cannot Link with Protection Manager (see section 6.6)
- Backing up and Restoring a File System Using a File Share (see section 6.7)
- Deleting Backup Data (see section 6.8)

6.1 Operations for Backing up and Restoring a File System

This section provides examples of the following kinds of backup and restore operations:

- Backing up and restoring a file system between volumes
- Backing up a file system to tape, and restoring the file system
- Backing up and restoring a file system with a backup management product that is unable to link with Protection Manager
- Backing up and restoring a file system using a file share

The rest of this section explains points to note when backing up and restoring a file system. For details on common operating notes that apply to all kinds of objects subject to backup, see section 5.3.

6.1.1 Precautions When Backing up a File System

Note the following points when backing up a file system:

- The volume being used by the operating system cannot be backed up.
- All the application programs using the volume to be backed up must be stopped.
- If you use the command, the cache of the secondary volume must be cleared. To clear the system cache, execute the `EX_DRM_CACHE_PURGE` command on the backup server.
- When performing an online backup, make sure that another volume is not mounted on the directory of the volume to be backed up.
- When a VSS backup is executed, the following event information might be output to the Windows application event log on the backup server:

  Event type: error

  Event source: PlugPlayManager

  Event ID: 12

  or

  Event type: warning

  Event source: ftdisk

  Event ID: 57

  or

  Event type: warning

  Event source: disk

  Event ID: 51

  This event information is logged during a VSS backup when the secondary volume becomes temporarily hidden. However, this will not influence the backup operation.

- When using VSS to back up data, you cannot back up file systems on a dynamic disk.

6.1.2 Precautions for Restoring a File System

Note the following points when restoring a file system:

- Before performing restoration, all the application programs using the volume to be restored must be stopped.
- The volume being used by the operating system cannot be restored.
- Do not use the `drmfsrestore` command in the following situations:
  - When you back up a file system from a secondary volume to tape device by using the `drmmediabackup` or `drmtapebackup` command.
  - When you restore a file system from the tape device to a secondary volume by using the `drmmediarestore` or `drmtaperestore` command.
  - When you mount a secondary volume by using the `drmmount` command.
- When you execute the `drmfsrestore` command, an error might result if the copy groups that are the restore targets and those that are not the restore targets are not in the following pair statuses:

  Pair status of copy group of the target volume:

  The primary volume is PSUS, and the secondary volume is SSUS.

  Pair status of copy group related to the target volume:

  PSUS or SMPL.

  Before performing restoration, you should use the RAID Manager `pairdisplay` command or `pairevtwait` command to check the pair status of the copy group.
- While restoring data obtained in a VSS backup, you must not open the physical disk resource properties on the cluster administrator window. If the properties are open, restoration may fail.

6.1.3 Precautions for Executing Commands in a Cluster Environment

- Do not access the operation-target volume (for example, by using Windows Explorer) while a backup or restoration command is being executed in a VERITAS Cluster Server environment. Doing so might place the cluster resources offline, resulting in an error in the backup or restoration command.
- If the `drmfsrestore` command is executed when the cluster resource status of the file system is offline and the environment setting is such that the dictionary map file is placed on a shared disk, the `drmfsrestore` command might result in an error because it will rely on cluster resources that have gone offline.

  The disk that stores the dictionary map file must be a cluster resource that has been set online. Before execution, you should check that it is in that status.

6.1.4 About Forced Termination of Commands

Do not force termination of any running commands provided by Protection Manager. If you force termination of these commands, the pair status and backup catalog of the copy group may become invalid. Perform the backup operation as shown in section 4.18.

6.2 Backing up and Restoring Data Between Volumes

In a configuration that uses only one server, you use commands to back up and restore data between the primary volume and secondary volume. Install the Protection Manager programs in the file server or the database server, and execute extended commands.

Figure 6.1 shows a system configuration for backing up and restoring data between volumes for a file system.

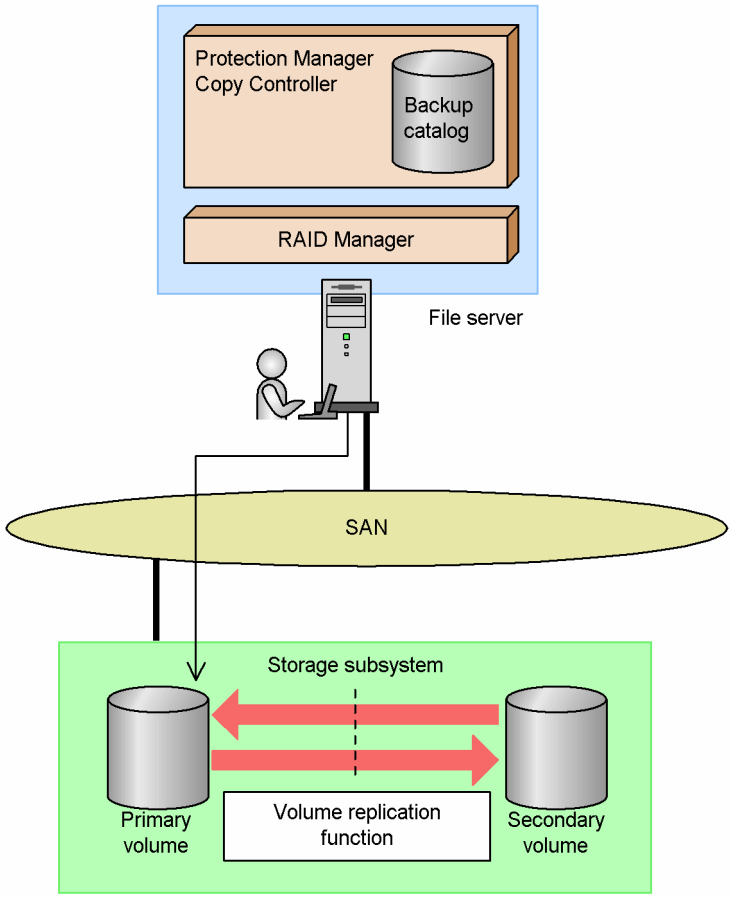


Figure 6.1 System Configuration for Backing up and Restoring Data Between Volumes (File System Backup)

6.2.1 Processing Flow for Backing up and Restoring Data Between Volumes

The processing flow for backing up a file system to a secondary volume and restoring it to the primary volume on a single-server system is shown below. Execute commands on the file server.

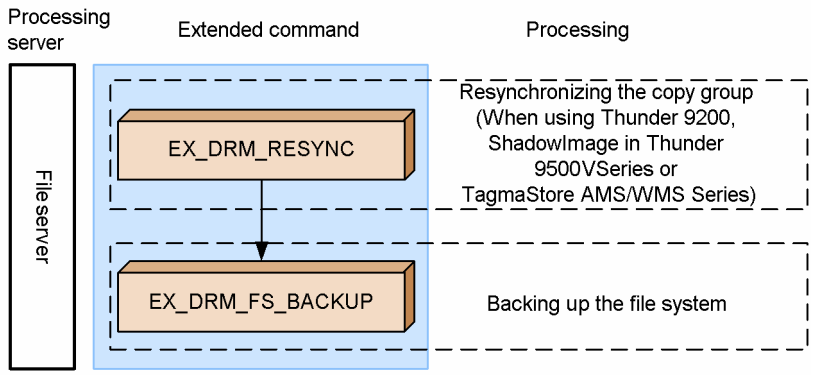


Figure 6.2 Procedure for Backing up a File System to the Secondary Volume

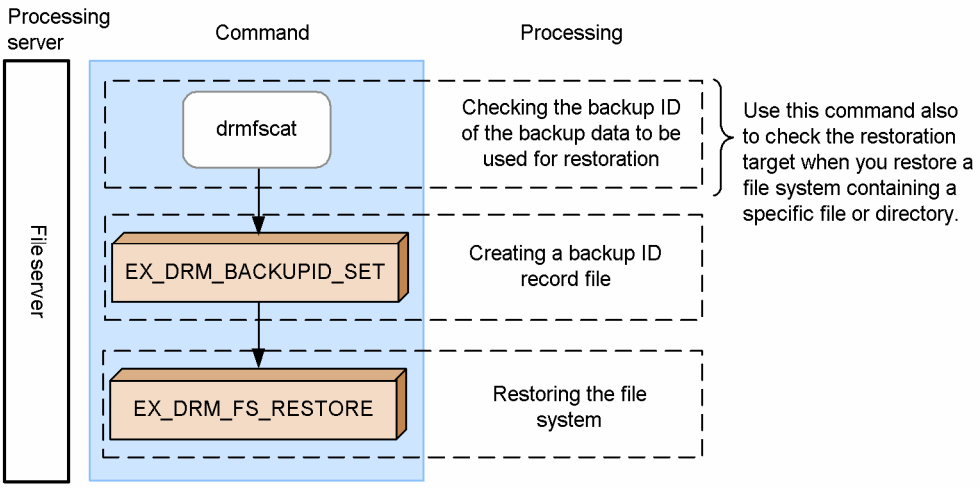


Figure 6.3 Procedure for Restoring a File System to the Primary Volume

6.2.2 Backing up a File System to a Secondary Volume

This section provides an example of backing up the mount point `E:` on the file server `FSServer` to a secondary volume in an online backup operation. Use `Operation_A` as an operation ID to identify the object to be backed up.

To back up the file system:

1. Resynchronize the copy groups.

   If you are using ShadowImage with a TagmaStore AMS or Thunder 9500V, or if you are using a Thunder 9200, execute `EX_DRM_RESYNC` on the file server to resynchronize the copy group. Resynchronizing the copy groups before you perform backup enhances the speed of backup processing. In the following example, `VG01,vol01` is a copy group name:

```
FSServer > EX_DRM_RESYNC Operation_A -cg VG01,vol01 -copy_size 7
```

2. Back up the file system to the secondary volume.

   Back up the file system to the secondary volume. To do this, execute `EX_DRM_FS_BACKUP`. Specify the operation ID `Operation_A` as the argument.

```
FSServer > EX_DRM_FS_BACKUP Operation_A
```

6.2.3 Restoring a File System to a Primary Volume

This section provides an example of restoring a file system when the backup data is stored on a secondary volume. This example resynchronizes the secondary volume to the primary volume and then restores the data. This example uses `Operation_A` as the operation ID that identifies the file system to be restored.

To restore the file system:

1. Check the backup ID of the backup data used to restore the file system, by executing the `drmfscat` command on the file server with a mount point name or mount point directory target definition file name specified as an argument.

```
FSServer >drmfscat E:
INSTANCE: E:
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: E:    ORIGINAL-ID: 0000000001
START-TIME:2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: FSServer
T FILE     FS          DG   DEVICE       COPY-GROUP
F -         E:          -    Harddisk2   VG01,vol01
F -         -           -    -             VG01,vol01
:
```

   If you restore a file system that contains a specific file or directory by executing a restoration command with the `-target` option specified, check the restoration target.

2. Create a backup ID record file by, executing `EX_DRM_BACKUPID_SET` on the backup server with the backup ID specified.
   To create a backup ID record file, execute `EX_DRM_BACKUPID_SET` with the backup ID specified.

```
FSServer > EX_DRM_BACKUPID_SET Operation_A -backup_id 0000000001
```

3. Restore the data of the secondary volume to the primary volume. Resynchronize the primary volume and secondary volume and then restore the data. To restore the data, execute `EX_DRM_FS_RESTORE` on the file server.

```
FSServer > EX_DRM_FS_RESTORE Operation_A -resync
```

6.3 Backing up and Restoring by Using a Tape Device

The following example is based on the system configuration shown in the following figure. A single-file-server system is used in this example, but a multi-file-server system can be used as appropriate.

Figure 6.4 shows a system configuration for backing up a file system to tape and restoring from tape.

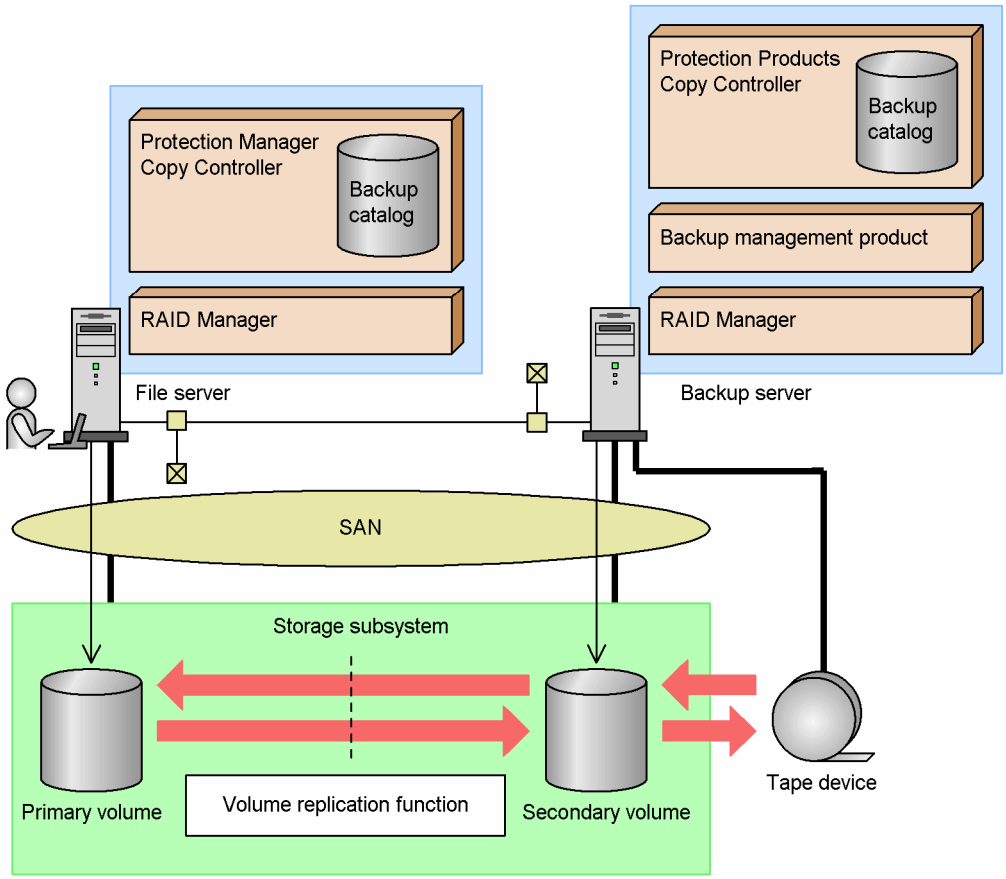


Figure 6.4 Configuration for Backup and Restore of a File System to/from Tape

With Protection Manager, file servers can be clustered under Windows. With file servers configured in a failover cluster, if the active server fails, a standby server is available to take over operations from the active server.

This example assumes the following conditions:

- The system is equipped with a file server (server name: `FSServer`), and a backup server (server name: `BKServer`) with a tape device connected to it.
- The primary volume and secondary volume are defined as a pair at the file server and the backup server.
- The backup management product is installed on the backup server.
- The FTP server is running on the backup server, thus allowing the users who can log on to the OS on each file server to log in the FTP server and transfer files to and from it. The FTP user ID must be `admin`, and the FTP user password must be `password`.
- The parameter for the backup management product has been registered using the `drmtapeinit` command.
- The `cscript //H:Cscript` command is executed at the command prompt to modify the host parameters.
- The primary volume paired with the secondary volume to be backed up is not backed up until the backup of the secondary volume to tape is completed.
- The mount point `E:` exists on `FSServer`.
- The mount point to be backed up has been formatted by NTFS.
- The secondary volume is normally not mounted, but it is mounted on the E drive only when needed.
- At each file server and the backup server, the storage directories for extended-command temporary files specified in the operation definition file have been created.

6.3.1 Processing Flow for Tape Device Backup

This section describes the processing flow of backing up a file system to tape and restoring it to the primary volume on a multi-server system. A single-file-server system is used in this example, but a multi-file-server system may be used as appropriate.

When usual operation is in the split state, resynchronize the copy group before backing up the data.

When usual operation is in the pair state, there is no need to resynchronize the copy group before backing up data. Instead, when the tape backup is complete, resynchronize the copy group to initialize it.

Figure 6.5 shows the procedure for backing up a file system to tape. Figure 6.6 shows the procedure for restoring a file system from tape.

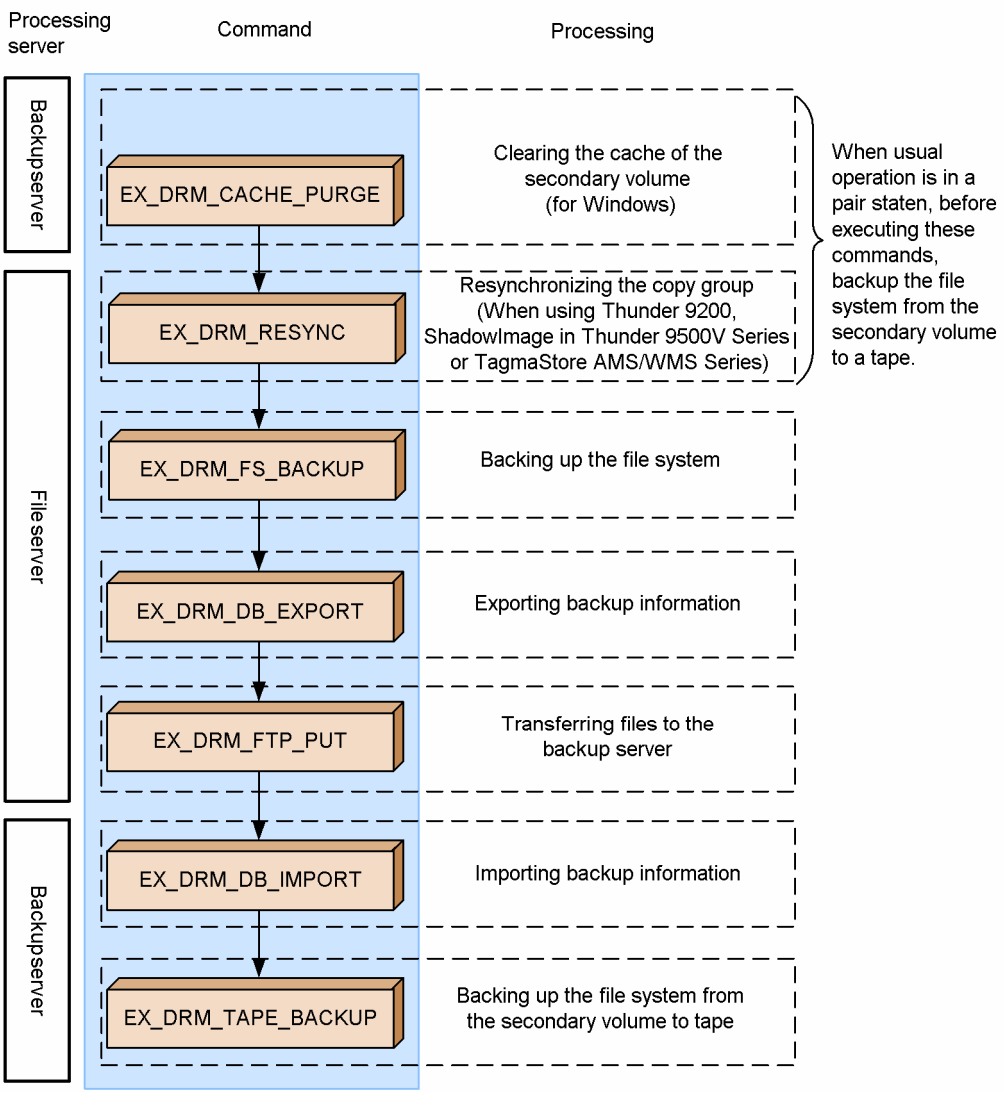


Figure 6.5 Procedure for Backing up a File System to Tape

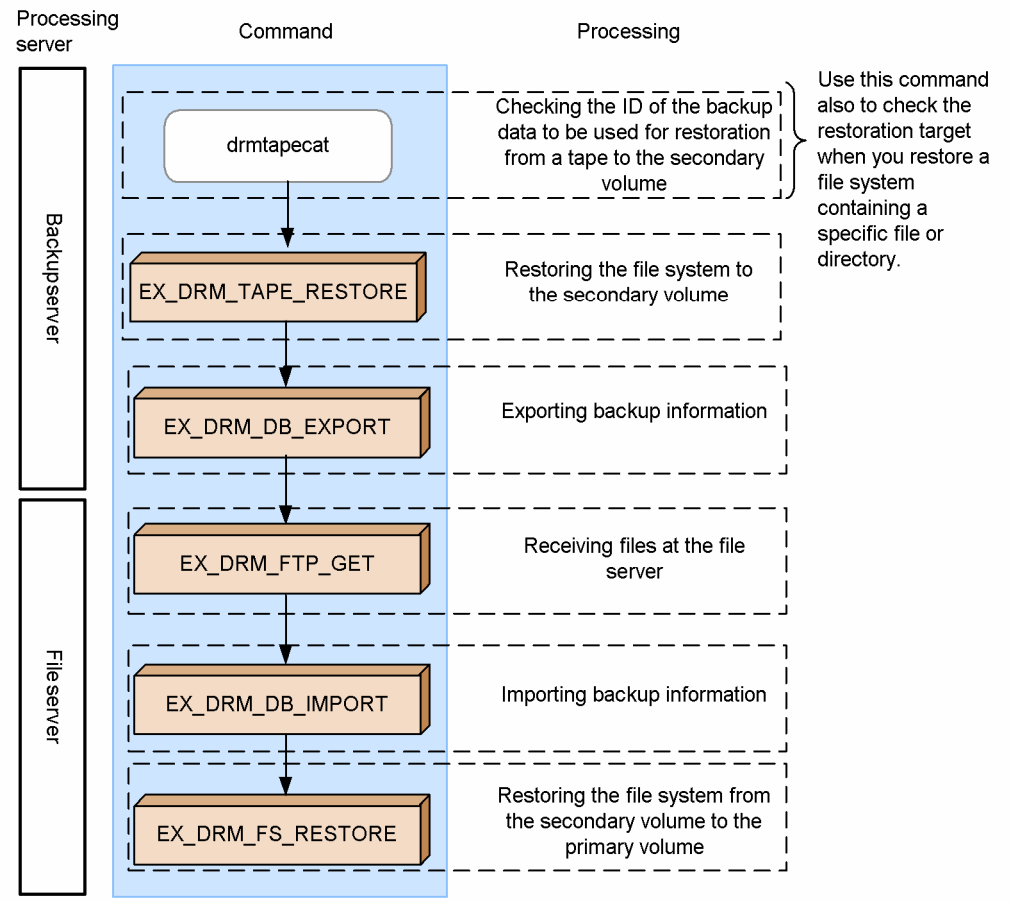


Figure 6.6 Procedure for Restoring a File System from Tape

6.3.2 Backing up a File System to Tape

This section gives an example of backup of a file system to a tape. In this example, we back up the file server `FSServer` to a secondary volume in an online backup operation before backing it up from a secondary volume to tape. As a mount point, `E:` is used. The example uses `Operation_A` as the operation ID that identifies the file system to be backed up.

6.3.2.1 Resynchronizing Copy Groups

When usual operation is in the split state, resynchronize the copy groups before backing up the data.

When usual operation is in the pair state, there is no need to resynchronize the copy groups before backing up data. Instead, when the tape backup is complete, resynchronize the copy group to initialize it.

To resynchronize the copy group:

1. Clear the secondary volume cache. on the backup server before backing up the data. To clear the system cache, execute `EX_DRM_CACHE_PURGE` on the backup server to mount or unmount the secondary volume. For now, assume the copy group name as `VG01,vol01`.

```
BKServer > EX_DRM_CACHE_PURGE Operation_A -cg VG01,vol01
```

2. If you are using a Thunder 9200 Series, and ShadowImage with a Thunder 9500V Series or a TagmaStore AMS/WMS Series, execute `EX_DRM_RESYNC` on the file server. Resynchronizing the copy group before backing up data will help speed up the backup time.

```
FSServer > EX_DRM_RESYNC Operation_A -cg VG01,vol01 -copy_size 7
```

6.3.2.2 Backing up a File System

To back up the file system to the secondary volume:

1. Execute `EX_DRM_FS_BACKUP`. Specify the operation ID `Operation_A` as the argument:

```
FSServer > EX_DRM_FS_BACKUP Operation_A
```

2. To check that backup information bearing the date on which the file system had been backed up from the primary volume to the secondary volume is available, execute the `drmfscat` command on the file server.

```
FSServer > drmfscat E:
INSTANCE: E:
BACKUP-ID:0000000001 BACKUP-MODE: ON-LINE INSTANCE: E: ORIGINAL-ID:0000000001
START-TIME:2003/08/01 19:00:00 END-TIME: 2003/08/01 19:03:00
:
:
```

   In this example, information about the backup operation carried out in step 1 has been loaded in the backup catalog with the new backup ID `0000000001`.

3. Export the information about the backup to temporary files. To back up data from the secondary volume to a tape, you must copy information about the backup that was performed from the primary volume to the secondary volume. Copy this information to the backup server. To export information about backup to temporary files, execute `EX_DRM_DB_EXPORT`. The temporary files are stored in the temporary directory for extended commands, on the file server.

```
FSServer > EX_DRM_DB_EXPORT Operation_A
```

4. Batch-transfer the temporary files from the file server to the backup server. To transfer the files, execute `EX_DRM_FTP_PUT` on the file server. In this example, the user ID used for logon to the FTP server is `admin` and the password is `password`. The temporary files are stored in the temporary directory for extended commands, on the backup server.

```
FSServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

5. Import the temporary files, which were transferred from the file server, into the backup catalog on the backup server. To import the temporary files, execute EX_DRM_DB_IMPORT on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

6. To check that backup information bearing the date on which the temporary files had been backed up from the primary volume to the secondary volume has been imported on the backup server, execute the drmfscat command on the backup server.

7. Back up the data from the secondary volume to a tape. To back up the data, execute EX_DRM_TAPE_BACKUP on the backup server. For the tape backup mount point, specify the E drive (drive letter E:).

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   Executing a backup in this example will load information about the backup operation in the backup catalog with new backup ID 0000000002.

8. Check that the file system has been backed up correctly. To check that backup information bearing the date on which the file system had been backed up from the secondary volume to tape is available, execute the drmtapecat command on the backup server.

6.3.2.3 Backup Using VSS

The following explains an example of using VSS to back up a file system to tape. When VSS is used, backup is performed with the disk drive mounted. In this example, using VSS we back up the file server FSServer to a secondary volume. Then we back it up from a secondary volume to tape, using the E: drive for the mount point. Operation_A is used as the operation ID.

To make a backup by using VSS, the Protection Manager service must be running on the backup server.

To use VSS to back up a file system:

1. Backup server, execute the drmvssimport start command to start the VSS import server. This step is not necessary if the VSS import server is already running.

```
BKServer > drmvssimport start
KAVX5009-I The VSS import server will now start.
KAVX5010-I The VSS import server has started.
```

2. Use VSS to back up the file system to the secondary volume. To do this, execute EX_DRM_FS_BACKUP. Specify the operation ID Operation_A as the argument.

```
FSServer > EX_DRM_FS_BACKUP Operation_A -mode vss
```

3. Check that backup was performed correctly. To check that backup information bearing the date on which the file system had been backed up from the primary volume to the secondary volume is available, execute the drmfscat command on the file server.

```
FSServer > drmfscat E:
INSTANCE: E:
BACKUP-ID:0000000001 BACKUP-MODE: VSS INSTANCE: E: ORIGINAL-ID:0000000001
START-TIME: 2003/08/01 19:00:00 END-TIME: 2003/08/01 19:03:00
:
:
```

In this example, information about the backup operation carried out in step 1 has been loaded in the backup catalog with the new backup ID 0000000001.

4. Export the backup information to temporary files. To perform backup from the secondary volume to tape, you need to copy information, about the backup operations from the primary volume to the secondary volume, to the backup server. To export information about backup to temporary files, execute EX_DRM_DB_EXPOR. The temporary files are stored in the temporary directory for extended commands, on the file server.

```
FSServer > EX_DRM_DB_EXPORT Operation_A
```

5. Transfer the temporary files in batch, from the file server to the backup server. To transfer the files, execute EX_DRM_FTP_PUT on the file server. In this example, the user ID used for logon to the FTP server is admin and the password is password. The temporary files are stored in the temporary directory for extended commands, on the backup server.

```
FSServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

6. Import the temporary files transferred from the file server, into the backup catalog of the backup server into the backup catalog of the backup server. To import the temporary files, execute EX_DRM_DB_IMPORT on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

7. Check that the temporary files have been imported correctly. To check that backup information bearing the date on which the temporary files had been backed up from the primary volume to the secondary volume has been imported on the backup server, execute the drmfscat command on the backup server.

8. Back up the data from the secondary volume to a tape by executing EX_DRM_TAPE_BACKUP on the backup server. Here, the drive letter of the secondary volume is E:.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   When backup is performed, information about these backup operations is registered in the backup catalog with the new backup ID 0000000002.

9. Check that the file system has been backed up correctly. To check that backup information bearing the date on which the file system had been backed up from the primary volume to the secondary volume is available, execute the drmtapecat command on the backup server.

   When you execute backup, information about this backup is registered in the backup catalog with the new backup ID 0000000002.

6.3.3 Restoring a File System from Tape

This section provides an example of restoring data that was backed up to a tape and recovering a file system. This example restores the data from the tape to a secondary volume, then resynchronizes the secondary volume to the primary volume, and restores the data. This example uses Operation_A as the operation ID that identifies the file system to be restored.

To restore the file system:

1. Check the backup ID of the backup data used to restore it from tape to the secondary volume. To check the backup ID, execute the `drmtapecat` command on the backup server.

```
BKServer >drmtapecat -hostname FSServer -l
BACKUP-ID ORIGINAL-ID HOSTNAME BACKUP-OBJECT INSTANCE SNAPSHOT TIME     EXPIRATION TIME
BACKUP-MEDIA
0000000002 0000000001 FSServer FILESYSTEM  E:   2003/08/01 19:00:00 2003/11/11 19:00:00  -
:
:
```

   If you restore a file system that contains a specific file or directory by executing a restoration command with the `-target` option specified, you must check the restoration target. In such a case, execute the `drmtapecat` command with the following options (arguments) specified:

   - `-o FILESYSTEM` *mount-point-directory-name-or-drive-name*
   - `-backup_id` *backup-ID*

2. Restore the backed up data from the tape to the secondary volume by executing `EX_DRM_TAPE_RESTORE` on the backup server.

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002
```

   When a restore is executed, information about the restore operation will be registered in the backup catalog of the backup server. The information has the new backup ID `0000000003`.

3. Check that the file system has been restored correctly from tape to the secondary volume. To check that backup information with the date of the backup from the primary volume to the secondary volume has been restored to the backup server, execute the `drmfscat` command on the backup server.

4. Export backup information to temporary files. To restore the data of the secondary volume to the primary volume, you must copy backup information about restore processing performed from the tape to the secondary volume. Copy this information to the file server. To export the backup information to temporary files in the temporary directory for extended commands, execute `EX_DRM_DB_EXPORT`.

```
BKServer > EX_DRM_DB_EXPORT Operation_A
```

5. Receive the temporary files on the file server. To batch-receive the temporary files held on the backup server onto the file server, execute `EX_DRM_FTP_GET` on the file server. In this example, the user ID used for logon to the FTP server is `admin` and the password is `password`. The temporary files are stored in the temporary directory for extended commands, on the file server.

```
FSServer > EX_DRM_FTP_GET Operation_A -server BKServer -user admin -password password
```

6. Import the temporary files, which were transferred from the backup server, into the backup catalog on the file server into the backup catalog on the file server. To import the temporary files, execute `EX_DRM_DB_IMPORT` on the file server.

```
FSServer >EX_DRM_DB_IMPORT Operation_A
```

7. Check that the temporary files have been imported correctly. To check that backup information bearing the date on which the temporary files had been backed up from the primary volume to the secondary volume has been imported on the file server, execute the `drmfscat` command on the file server.

8. Restore the data of the secondary volume to the primary volume. Resynchronize the primary volume and secondary volume and then restore the data. To restore the data, execute `EX_DRM_FS_RESTORE` on the file server.

```
FSServer > EX_DRM_FS_RESTORE Operation_A -resync
```

6.4 Backing Up a File System by Using a User Script

By using a backup command that specifies a user script, you can use a single operation to back up a file system to tape from the primary volume via the secondary volume.

The example is based on the system configuration shown in the figure below.

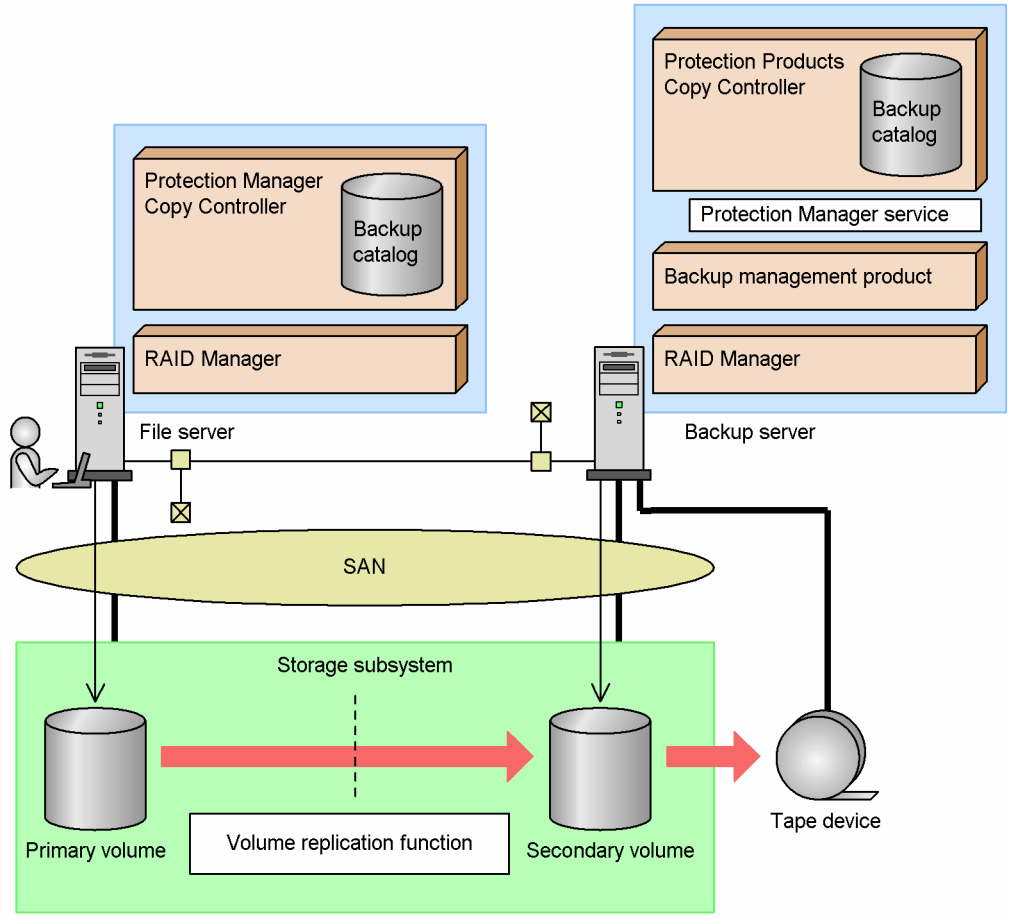


Figure 6.7 System Configuration for Backup of a File System to Tape

6.4.1 Processing Flow for User Script Backup

In this example, the processing flow of the `drmfsbackup` command that specifies a user script is as follows:

- All application programs that are using the `F` drive are stopped (this is a step in the user pre-processing section).
- The cache of the secondary volume is cleared (this is a step in the user pre-processing section).
- Once the cold backup from the primary volume to the secondary volume has been completed, the backup catalog is transferred to the backup server.

- On the backup server, the secondary volume is mounted on the `D` drive.
- The secondary volume is backed up to tape by using `NTBACKUP`, and then is unmounted (this is a step in the user post-processing section).
- The application programs that use the `F` drive are restarted without waiting for the tape backup operation to terminate (this is a step in the user post-processing section).

The `drmfsbackup` command terminates without waiting for the tape backup operation to terminate).

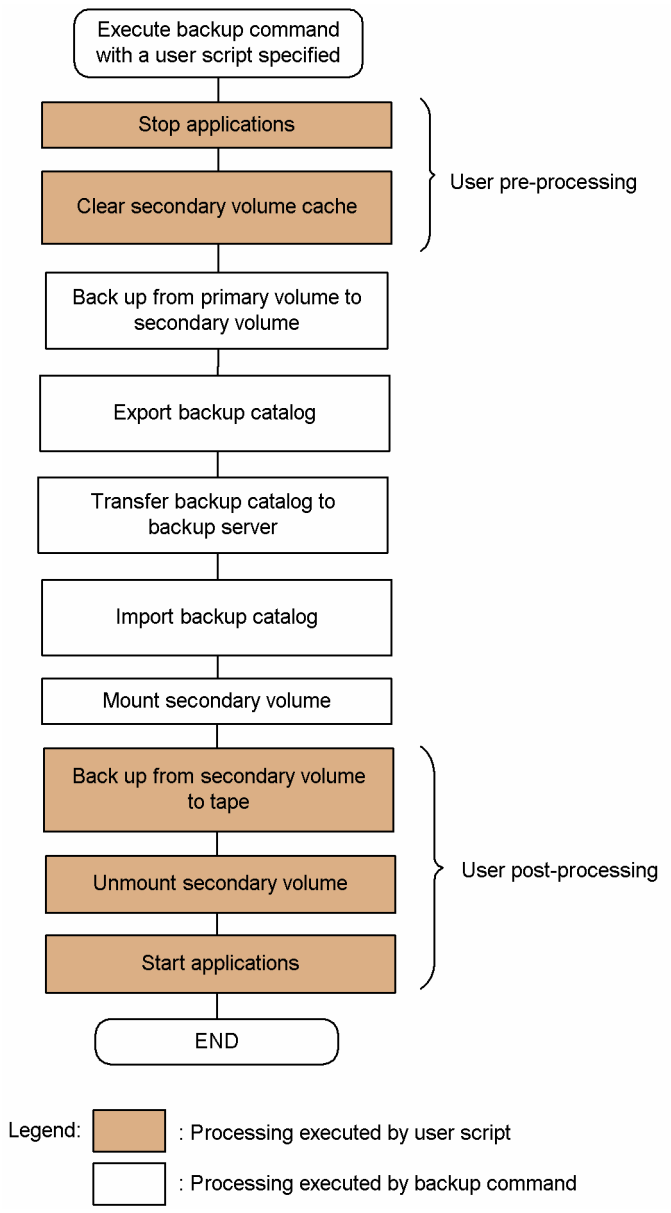


Figure 6.8 Processing Flow

6.4.2 User Script Example

The table below shows the user script that is created for the example.

Table 6.1 User Script Creation Example

Script text	Explanation
`LOCAL_BACKUP=YES ...(1)`	(1) `YES` is required.
`#Pre-processing section`	
`[PRE_PROC] ...(2)`	(2) Start of the user pre-processing section
`[CMD]`	
`#Stop the application programs that are using the file system` `CMDLINE=C:\tmp\stop_app.bat`	
`ENV=VAL=ApplicationName ...(3)`	(3) Pass the environment variable `VAL=ApplicationName` to the command.
`TIMEOUT=60 ...(4)`	(4) Take a 60-second timeout.
`END_CODE=100 ...(5)`	(5) Treat a command return value of 100 or greater as an error.
`LOCATION=LOCAL ...(6)`	(6) Execute on the local server.
`#Clearing of the secondary volume cache (mounting and unmounting)` `[CMD]`	
`CMDLINE="C:\program files\hitachi\drm\bin\drmmount.exe" -copy_group vg01,vol01` `LOCATION=REMOTE ...(7)`	(7) Execute on a remote server.
`[CMD]`	
`CMDLINE="C:\program files\hitachi\drm\bin\drmumount.exe" -copy_group vg01,vol01` `LOCATION=REMOTE`	
`#Post-processing section`	
`[POST_PROC] ...(8)`	(*8*) Start of the user post-processing section
`#Tape backup of the secondary volume`	
`[CMD]`	
`CMDLINE=C:\tmp\tapebackup.bat`	
`TIMEOUT=NOWAIT ...(9)`	(*9*) Execute the next command without waiting for the command to terminate.
`END_CODE=TERMINATE_NZ ...(10)`	(*10*) Treat a command return value of a value other than 0 as an error.
`LOCATION=REMOTE`	
`PARENT_STAT=NORMAL ...(11)`	(*11*) Execute only when the backup command is normal.
`[CMD]`	

Script text	Explanation

#Application recovery processing if there is a backup error
CMDLINE=C:\tmp\recovery.bat
TIMEOUT=60 ...*(12)*
 	
END_CODE=IGNORE ...*(13)*
 	
LOCATION=LOCAL
 	
PARENT_STAT=ERROR ...*(14)*
 	
[CMD]
 	
#Re-start the application programs that use the file system
CMDLINE=C:\tmp\restart_app.bat
 	
ENV=VAL=ApplicationName
 	
TIMEOUT=60
 	
END_CODE=100
 	
LOCATION=LOCAL
	(*12*) Take a 60-second timeout.
	
	(*13*) Ignore any command error.
	
	(*14*) Execute only when the backup command has an error.

The following is an example of backing up the secondary volume to tape by using `tapebackup.bat`:

```
rem Use NTBACKUP to execute the job Job1,and perform a copy backup of G:\ to tape Tape1
rem Set the backup comment passed to the DRMENV_COMMENT environment variable as the backup job explanation
"C:\Windows\system32\ntbackup.exe" backup G:\ /j "Job1" /a /t "Tape1" /D "%DRMENV_COMMENT%" /m copy
IF NOT "%errorlevel%"=="0" GOTO ERROR

rem After the tape backup, specify the backup ID imported to the backup server, and unmount the secondary volume
"C:\Program Files\Hitachi\DRM\bin\drmumount.exe" %DRMENV_R_BACKUPID%
IF NOT "%errorlevel%"=="0" GOTO ERROR
exit 0
:ERROR
exit 1
```

Note: A line that begins with the characters `rem` is a comment line.

6.4.3 Backup Execution Example

An example of using a user script is shown below. If you specified `LOCATION=REMOTE` in the user script, you must specify the `-s` option on the command line.

```
PROMPT>drmfsbackup F: -mode cold -script C:\tmp\script.txt -s BKHOST -auto_import
-auto_mount G: -comment TEST1
```

The execution results are shown below:

```
PROMPT> drmfsbackup F: -mode cold -script C:\tmp\script.txt -s BKHOST -auto_import
-auto_mount G: -comment TEST1
KAVX0001-I The drmfsbackup command will now start.
```

```
KAVX0256-I A connection to the Protection Manager service will now be established.
           Host name = BKHOST
KAVX0210-I The user script will now start.
           Procedure section = [PRE_PROC]
KAVX0263-I A command in the user script will now be executed.
           Host name = LOCAL
           Command line = C:\tmp\stop_app.bat
KAVX0213-I A command from the user script has ended.
           End code = 0
KAVX0263-I A command in the user script will now be executed.
           Host name = BKHOST
           Command line = "C:\program files\hitachi\drm\bin\drmmount.exe" -copy_group
vg01,vol01
KAVX0213-I A command in the user script has ended.
           End code = 0
KAVX0263-I A command in the user script will now be executed.
           Host name = BKHOST
           Command line = "C:\program files\hitachi\drm\bin\drmumount.exe" -copy_group
vg01,vol01
KAVX0213-I A command in the user script has ended.
           End code = 0
KAVX0211-I Execution of the user script has finished.
KAVX0019-I Copying file(s) now...
KAVX5108-I Resynchronizing a copy group.
           Copy group name = SI01,dev01
KAVX5111-I Performing unmounting.
           Mount point name = F:
KAVX5109-I Dividing a copy group in pair.
           Copy group name = SI01,dev01
KAVX5110-I Performing mounting.
           Mount point name = F:
KAVX0040-I The backup ID has been assigned as follows:
           Backup ID = 0000000001
KAVX5156-I The backup catalog will now be exported.
KAVX5157-I The backup catalog was exported.
KAVX5158-I The backup catalog will now be imported.
           Host name = BKHOST
KAVX5159-I The backup catalog has been imported.
           Dictionary map file path = C:\Program Files\Hitachi\DRM\db
           Imported backup ID = 0000000001
KAVX5162-I The secondary volume will now be mounted.
           Host name = BKHOST
           Imported backup ID = 0000000001
KAVX0001-I The drmmount command will now start.
KAVX0400-I Mounting will now start.
           Mount point = G:
KAVX0401-I Mounting has finished.
           Mount point = G:
KAVX0002-I The drmmount command will now end.
KAVX5163-I The secondary volume has been mounted.
KAVX0210-I The user script will now start.
           Procedure section = [POST_PROC]
KAVX0263-I A command in the user script will now be executed.
           Host name = BKHOST
           Command line = C:\tmp\tapebackup.bat
KAVX0213-I A command from the user script has ended.
           End code = 0
KAVX0263-I A command in the user script will now be executed.
           Host name = LOCAL
           Command line = C:\tmp\restart_app.bat
KAVX0213-I A command from the user script has ended.
           End code = 0
KAVX0211-I Execution of the user script has finished.
KAVX0257-I The connection with the Protection Manager service will now be cut.
KAVX0002-I The drmfsbackup command will now end.
```

6.5 Backup and Restore Between Remote Sites (Remote Copy)

This section provides examples for using volume replication functionality between storage subsystems, such as TrueCopy or UR, to back up a file system from the primary volume in the local site to the secondary volume in the remote site, and for performing restoration from the secondary volume in the remote site to the primary volume in the local site.

The following cases are explained:

- Backing up a file system to a remote site.
- Restoring a file system that was backed up to a remote site, to a local site (for when the local site is running properly).
- Restoring a file system that was backed up to a remote site, in the remote site (for when a failure occurs in the local site).
- Recovering a file system, from a remote site to the local site (for recovery from local site failure).

The following example is based on the system configuration shown in Figure 6.9.

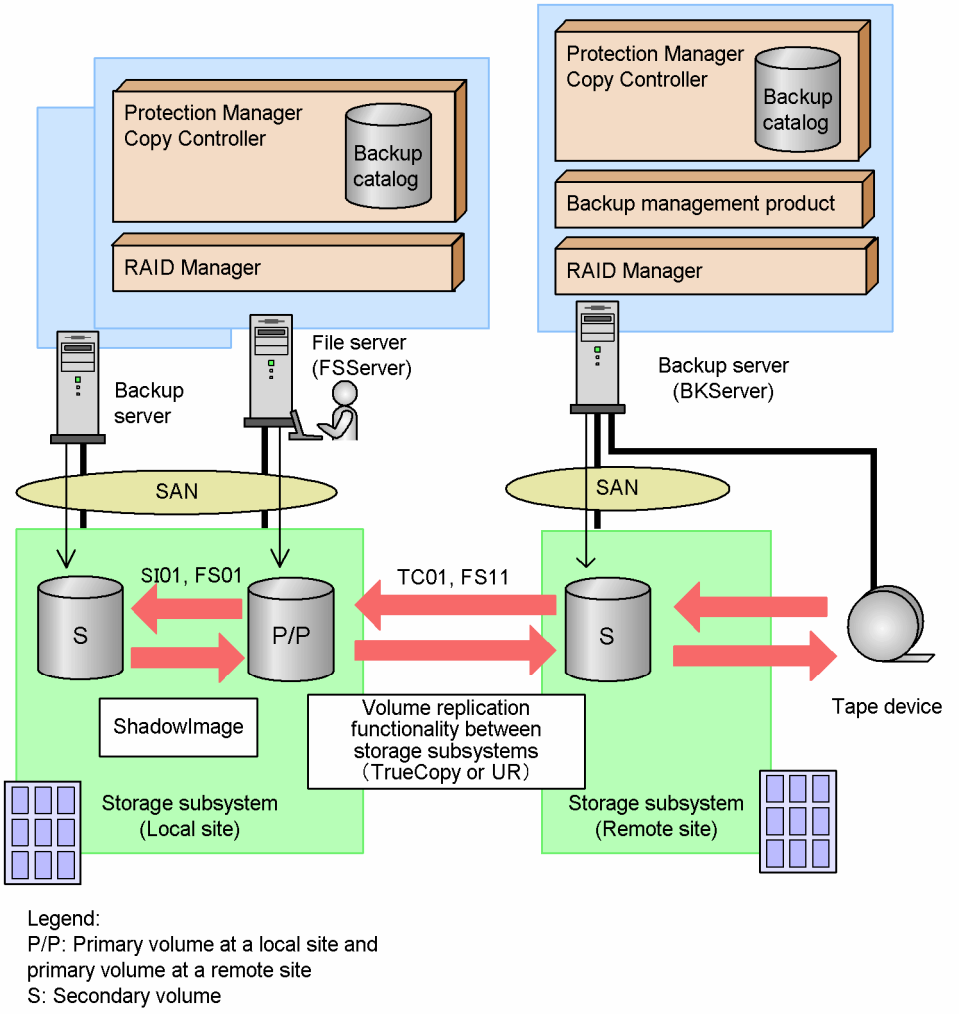


Figure 6.9 System Configuration for Backing up and Restoring a File System Between Remote Sites

The system prerequisites for this example are as follow:

- The system is equipped with a file server in the local site (server name: `FSServer`), and a backup server (server name: `BKServer`) with a tape device connected.
- An FTP service is running on the backup server of the remote site, and is set such that users who can log on to the OS on each file server can log on to the FTP server and transfer files. The FTP user ID must be `admin`, and the FTP user password must be `password`.
- The primary volume of the local site is mounted on the `G:` drive of `FSServer`.
- A TrueCopy pair (usual operation is in the pair state) is built for copy group name `TC01,FS11`.
- A ShadowImage pair is built for copy group name `SI01,FS01`.
- The name identifying the generation of the backup destination in the remote site is set to `remote_0`.
- Remote site volumes are not normally mounted, but are mounted to the `G:` drive of the backup server when necessary.

- The backup server in the remote site (BKServer) can be used as a file server when a failure occurs in the storage subsystem at the local site.
- Operation_A is used as the operation ID.

6.5.1 Backing up a File System to a Remote Site

This section provides an example of backing up a file system to a remote site. Procedures at the local site and remote site are explained separately.

6.5.1.1 Backup Operations at the Local Site

1. Check the volume to be backed up, as well as the name identifying the generation.

   On the file server, execute the drmfsdisplay command.

```
FSServer> drmfsdisplay -cf
INSTANCE: -
T FILE   FS   DG   DEVICE      COPY-GROUP     COPY-FUNC    GEN-NAME
F -      G:   -    Harddisk1   SI01,FS01      ShadowImage  local_0
- -      -    -    -           SI01,FS01      ShadowImage  local_0
F -      G:   -    Harddisk2   TC01,FS11      TrueCopy     remote_0
- -      -    -    -           TC01,FS11      TrueCopy     remote_0
FSServer>
```

2. Back up the file system to the secondary volume of the remote site. Perform a cold backup of the file system to the secondary volume of the remote site. Specify remote_0 for the name identifying the generation of the backup destination on the file server, and execute EX_DRM_FS_BACKUP.

```
FSServer > EX_DRM_FS_BACKUP Operation_A -mode cold -rc remote_0
```

3. Check that backup was performed correctly. On the file server, execute the drmfscat command, and check the backup information.

```
FSServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000001 BACKUP-MODE: COLD INSTANCE: G: ORIGINAL-ID: 0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:03:00 HOSTNAME: FSServer
T FILE          FS           DG        DEVICE              COPY-GROUP
F G:            G:           -         Harddisk1           TC01,FS11
F -             -            -         -                   TC01,FS11
FSServer >
```

   In this example, information about the backup operations carried out in step 2 has been registered in the backup catalog. The information has the new backup ID 0000000001.

4. Export the backup information to temporary files. To perform backup from the secondary volume to tape, you need to copy information about the backup operations from the primary volume to the secondary volume, to the backup server. To export information about backup to temporary files, execute EX_DRM_DB_EXPORT. The temporary files are stored in the temporary directory for extended commands, on the file server.

```
FSServer > EX_DRM_DB_EXPORT Operation_A
```

5. Transfer the temporary files in batch, from the file server to the backup server. To transfer the files, execute EX_DRM_FTP_PUT on the file server. The temporary files are stored in the temporary directory for extended commands, on the backup server.

```
FSServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

6.5.1.2 Backup Operations at the Remote Site

1. Import the temporary files transferred from the file server in the local site, into the backup catalog of the backup server. To import the temporary files, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

2. Check that the temporary files have been imported correctly. To check that backup information bearing the date on which the temporary files had been backed up from the primary volume to the secondary volume has been imported on the backup server, execute the `drmfscat` command on the backup server.

```
BKServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000001 BACKUP-MODE: COLD INSTANCE: G: ORIGINAL-ID: 0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:03:00 HOSTNAME: FSServer
T FILE          FS          DG        DEVICE              COPY-GROUP
F G:            G:          -         Harddisk2           TC01,FS11
F -             -           -         -                   TC01,FS11
BKServer >
```

3. Back up the data from the secondary volume to a tape. To back up the data, execute `EX_DRM_TAPE_BACKUP` on the backup server. For the tape backup mount point, specify the G drive (drive letter `G:`).

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A –mount_pt G:
```

   When a backup is executed, information about the backup operation will be registered in the backup catalog of the backup server. In this example, the information has the new backup ID `0000000002`.

4. Check that the file system has been backed up correctly. To check that backup information bearing the date on which the file system had been backed up from the primary volume to the secondary volume is available, execute the `drmtapecat` command on the backup server.

```
BKServer > drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000002  FILESYSTEM       2004/08/05 10:00:00  2004/11/05 10:00:00  MEDIA1
:
:
BKServer >
```

5. Clear the secondary volume cache. After the tape backup operation finishes, clear the system cache on the backup server before backing up the data.

   To clear the system cache, execute `EX_DRM_CACHE_PURGE` on the backup server to mount or unmount the secondary volume.

```
BKServer > EX_DRM_CACHE_PURGE Operation_A
```

6.5.1.3 Synchronizing Operations at the Local Site

Resynchronize the copy group by executing `EX_DRM_RESYNC` on the file server.

```
FSServer > EX_DRM_RESYNC Operation_A
```

6.5.2 Restoring a File System to the Local Site

This section explains an example for restoring a file system to a local site, when the local site is running normally, without failure. Procedures at the local site and remote site are explained separately.

6.5.2.1 Restore Operations at the Remote Site

1. Split the TrueCopy pair (usual operation is in the pair state).

```
BKServer > pairsplit -g TC01 -d FS11 -rw
BKServer >
```

2. Check the backup ID of the backup data. To check the backup ID, execute the `drmtapecat` command on the backup server.

```
BKServer > drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000002  FILESYSTEM       2004/08/05 10:00:00  2004/11/05 10:00:00  MEDIA1
:
:
BKServer >
```

3. Restore the backed up data from tape to the secondary volume. To restore the data, execute `EX_DRM_TAPE_RESTORE` on the backup server. For the mount point, specify the G drive (drive letter `G:`).

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002 -mount_pt G:
```

When a restore is executed, information about the restore operation will be registered in the backup catalog of the backup server. In this example, the information has the new backup ID `0000000003`.

4. Check that restoration from tape to the secondary volume was performed correctly. To check that backup information with the date of the backup from the primary volume to the secondary volume has been restored to the backup server, execute the `drmfscat` command on the backup server.

```
BKServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000003 BACKUP-MODE: COLD INSTANCE: G: ORIGINAL-ID: 0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:03:00 HOSTNAME: FSServer
T FILE          FS          DG        DEVICE             COPY-GROUP
F G:            G:          -         Harddisk2          TC01,FS11
F -             -           -         -                  TC01,FS11
BKServer >
```

5. Export the backup information to temporary files. To perform restoration from the secondary volume to the primary volume, copy backup information about the restoration operations from tape to the secondary volume, to the file server. To export the backup information to temporary files in the temporary directory for extended commands, execute `EX_DRM_DB_EXPORT`.

```
BKServer > EX_DRM_DB_EXPORT Operation_A
```

6.5.2.2 Restore Operations at the Local Site

1. Obtain the temporary files, on the file server of the local site. To batch-receive the temporary files held on the backup server onto the file server, execute `EX_DRM_FTP_GET` on the file server. The temporary files are stored in the temporary directory for extended commands, on the file server.

```
FSServer > EX_DRM_FTP_GET Operation_A -server BKServer -user admin -password password
```

2. Import the temporary files transferred from the backup server, into the backup catalog of the file server by executing `EX_DRM_DB_IMPORT` on the file server.

```
FSServer >EX_DRM_DB_IMPORT Operation_A
```

3. Check that the temporary files have been imported correctly. To check that backup information bearing the date on which the temporary files had been backed up from the primary volume to the secondary volume has been imported on the file server, execute the `drmfscat` command on the file server.

```
FSServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000002 BACKUP-MODE: COLD INSTANCE: G: ORIGINAL-ID: 0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:03:00 HOSTNAME: FSServer
T FILE          FS          DG        DEVICE             COPY-GROUP
F G:            G:          -         Harddisk1          FS001,DDrive
F -             -           -         -                  FS001,DDrive
FSServer >
```

4. Restore the data of the secondary volume to the primary volume. Resynchronize the primary volume and secondary volume and then restore the data. To restore the data, execute `EX_DRM_FS_RESTORE` on the file server.

```
FSServer > EX_DRM_FS_RESTORE Operation_A -resync
```

5. Resynchronize the copy group by executing `EX_DRM_RESYNC` on the file server.

```
FSServer > EX_DRM_RESYNC Operation_A
```

6.5.3 Restoring a File System to a Remote Site

This section provides an example of restoring a file system in a remote site when failure occurs in the local site.

6.5.3.1 Operations at the Remote Site

1. Split the TrueCopy pair (usual operation is in the pair state).

```
BKServer > pairsplit -g TC01 -d FS11 -S
BKServer >
```

2. Check the backup ID of the backup data by executing the `drmtapecat` command on the backup server.

```
BKServer > drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000002  FILESYSTEM       2004/08/05 10:00:00  2004/11/05 10:00:00  MEDIA1
:
:
BKServer >
```

3. Restore the backed up data from tape to the secondary volume by executing `EX_DRM_TAPE_RESTORE` on the backup server. For the mount point, specify the G drive (drive letter `G:`).

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002 -mount_pt G:
```

When a restore is executed, information about the restore operation will be registered in the backup catalog of the backup server. In this example, the information has the new backup ID `0000000003`.

4. Check that restoration from tape to the secondary volume was performed correctly. To check that backup information with the date of the backup from the primary volume to the secondary volume has been restored to the backup server, execute the `drmfscat` command on the backup server.

```
BKServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000003 BACKUP-MODE: COLD INSTANCE: G: ORIGINAL-ID: 0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:03:00 HOSTNAME: FSServer
T FILE          FS           DG        DEVICE              COPY-GROUP
F G:            G:           -         Harddisk2           TC01,FS11
F -             -            -         -                   TC01,FS11
BKServer >
```

5. Use the disk management function to mount the restored volume. The mount is to the `G:` drive.

6.5.4 Recovering a File System from a Remote Site to the Local Site

This section explains how to recover restored data in the remote site to the local site, after the file system is restored in the remote site and recovery has been performed for the failure in the local site. Procedures at the local site and remote site are explained separately.

6.5.4.1 Operations at the Local Site

In the file server at the local site, use a disk management function to unmount the volume that was mounted by using the disk management function.

6.5.4.2 Operations at the Remote Site

1. Use the disk management function to unmount the volume mounted using the disk management function, in the step in which the file system was restored in the remote site. The `G:` drive is unmounted.
2. Create a pair.

```
BKServer > paircreate -g TC01 -d FS11 -vl -f never
BKServer >
```

6.5.4.3 Operations at the Local Site

1. Swap the primary volume and the secondary volume.

```
FSServer > pairresync -g TC01 -d FS11 -swaps
FSServer >
```

2. Use the disk management function to mount the volume. The mount is to the `G:` drive.

6.6 Using a Backup Management Product That Cannot Link with Protection Manager

When using a backup management product that is unable to link with Protection Manager, perform the following procedure to back up data to tape, or restore data from tape. For more details on steps 1 to 6 in the backup procedure, see section 6.3.2. For more details on the restoration procedure, see section 6.3.3.

To back up a file system to tape when using a backup management product that is unable to link with Protection Manager is used):

1. If usual operation is in the split state, execute `EX_DRM_CACHE_PURGE` to clear the secondary volume cache.
2. When usual operation is in the split state, execute `EX_DRM_RESYNC` to resynchronize the copy group.
3. Execute `EX_DRM_FS_BACKUP` to back up the file system to the secondary volume.
4. Execute `EX_DRM_DB_EXPORT` to export the backup information to temporary files.
5. Execute `EX_DRM_FTP_PUT` to transfer the temporary files to the backup server.
6. Execute `EX_DRM_DB_IMPORT` to import the temporary files transferred from the file server to the backup catalog on the backup server.
7. Check the backup ID. To check the backup ID, open the backup ID record file (*operation-ID*.bid) in the following directory on the backup server:

   *FTP_HOME_DIR-value**FTP_SUB_DIR-value**operation-ID*\BK*operation-ID*.bid

8. Using the backup ID checked in the backup ID storage file as the argument, check the file subject to backup. To check the ID, execute the `drmfscat` command.
9. Check the storage location of the data files by executing the `drmmount` command using the backup ID as the argument. The command results show the mounted drive name. The mounted drive stores data files.
10. Use a backup management product to back up all the files in the mounted drive to tape.
11. Execute the `drmumount` command to unmount the mount point mounted in step 9.

To restore a file system from a tape (where a backup management product that is unable to link with Protection Manager is used):

1. Execute the `drmfscat` command to identify the backup ID for the backup catalog to be restored.
2. Execute the `drmmount` command, and then restore the backup from a tape to the storage location of the data files.
3. Execute `EX_DRM_BACKUPID_SET` with a backup ID specified to create a backup ID record file.
4. Carry out the procedures following step 4 in section 6.3.3 to restore the backup from the secondary volume to the primary volume.

6.7 Backing up and Restoring a File System Using a File Share

When backup information is transferred between a file server and a backup server, extended commands (`EX_DRM_FTP_PUT` or `EX_DRM_FTP_GET`) are normally used to transfer the information in an FTP operation. Instead of FTP, however, file sharing can also be used to copy backup information from server to server. The following sections describe examples of backing up or restoring a file system using a file share.

6.7.1 Preparing to Back up or Restore a File System by Using a File Share

To prepare for backing up or restoring a file system by using a file share:

1. Check the temporary directory for extended commands that was created automatically by the check tool for operation definition files. For more information about the temporary directory for extended commands, see section 4.15.9.
2. When running Explorer or any other tool, put the temporary directory for extended commands that is on the backup server in a shared state.
3. From the file server, mount the temporary directory for extended commands that is on the backup server.

   When running Explorer or any other tool on the file server, assign the shared temporary directory for extended commands to the backup server as a network drive. For now, assign it to `X:` on the file server.
4. Set access permissions for the automatically generated temporary directory for extended commands that is on the backup server to make it accessible to users who are executing extended commands from the file server to read from or write to that directory.

6.7.2 Example of Backing up a File System by Using a File Share

This section describes the procedure for backing up a file system by using a file share. For more operational details on steps 1 to 4 and step 6 and subsequent steps, see section 6.3.2.

To back up a file system by using a file share:

1. If usual operation is in the split state, execute `EX_DRM_CACHE_PURGE` to clear the secondary volume cache.
2. When usual operation is in the split state, execute `EX_DRM_RESYNC` to resynchronize the copy group.
3. Execute `EX_DRM_FS_BACKUP` to back up the file system to the secondary volume.
4. Execute `EX_DRM_DB_EXPORT` to export the backup information to temporary files.
5. Copy the temporary files from the file server to the backup server. To copy the temporary files from the temporary directory for extended commands on the file server to the temporary directory for extended commands on the backup server, execute a copy command on the file server.

```
FSServer > copy /y
extended-command-temporary-directory-on-the-file-server\Operation_A\DB\Operation_A.drm
X:\
```

6. Execute EX_DRM_DB_IMPORT to import the temporary files copied from the file server to the backup catalog on the backup server.
7. Execute EX_DRM_TAPE_BACKUP to back up the data on the secondary volume to tape.
8. If usual operation is in the pair state, execute EX_DRM_CACHE_PURGE to clear the secondary volume cache.
9. When usual operation is in the pair state, execute EX_DRM_RESYNC to resynchronize the copy group.

6.7.3 Example of Restoring a File System Using a File Share

This section explains how to restore a file system on the backup server via file sharing. For more operational details on steps 1 to 3 and step 5 and subsequent steps, see section 6.3.3.

To restore a file system using a file share:

1. Execute the drmtapecat command to check the backup ID of the backup data.
2. Execute EX_DRM_TAPE_RESTORE to restore the backup data from the tape to the secondary volume.
3. Execute EX_DRM_DB_EXPORT to export the backup information to temporary files in the temporary directory for extended commands.
4. Copy the temporary files from the backup server to the file server. To copy the temporary files from the temporary directory for extended commands on the backup server to the temporary directory for extended commands on the file server, execute the copy command on the file server.

```
FSServer > copy /y X:\Operation_A.drm
extended-command-temporary-directory-on-the-file-server\Operation_A\DB
```

5. Execute EX_DRM_DB_IMPORT to import the temporary files copied from the backup server to the backup catalog on the file server.
6. Execute EX_DRM_FS_RESTORE to restore the secondary volume data to the primary volume.

6.8 Deleting Backup Data

If the backup data on the secondary volume is no longer necessary, delete it after it has been backed up from a secondary volume to tape, or after the data has been restored from a secondary volume to the primary volume.

To delete the backup data, execute the `drmresync` command, specifying the backup ID of the data to be deleted.

```
PROMPT> drmresync -backup_id backup-ID
KAVX0001-I The drmresync command will now start.
KAVX0002-I The drmresync command will now end.
PROMPT>
```

When you execute this command, the primary volume and the secondary volume will be resynchronized, and mirrored. Now the backup information of the specified backup ID will be deleted from the backup catalog.

Chapter 7 Using HPtM With SQL Server Databases

This chapter describes how to operate Protection Manager when backing up a SQL Server database, focusing on actual procedures. For details on setting up commands, see the *HiCommand Protection Manager Command Reference* manual.

- Operations for Backing up and Restoring a SQL Server Database (see section 7.1)
- Backing up and Restoring Data Between Volumes (see section 7.2)
- Backing up and Restoring Using a Tape Device (see section 7.3)
- Backing Up a SQL Server Database Using a User Script (see section 7.4)
- Operation Using SQL Server Transaction Logs (see section 7.5)
- Backing up and Restoring Transaction Log Backup Files (see section 7.6)
- Backing up and Restoring Multiple Generations by Locking a Copy Group (see section 7.7)
- Performing Backup and Restore Operations Between Remote Sites (Remote Copying) (see section 7.8)
- Backing up and Restoring in a Multi-Target Configuration (see section 7.9)
- Performing Restoration and Recovery on a Host Other than the Backup Host (see section 7.10)
- Using the Log-Shipping Function of a SQL Server Database (see section 7.11)
- Backing up and Restoring in a Cascade or Multi-target Configuration (see section 7.12)
- Using a Backup Management Product That Cannot Link with Protection Manager (see section 7.13)
- Backing up and Restoring a SQL Server Database by Using a File Share (see section 7.14)
- Restoring Data to a SQL Server Instance That Is Different from the Backup Source (see section 7.15)
- Deleting Backup Data (see section 7.16)
- Making a SQL Server Database Backed Up to a Secondary Volume Available for Use (see section 7.17)
- Backing Up and Restoring Using the SQL Server Replication Configuration (see section 7.18)

7.1 Operations for Backing up and Restoring a SQL Server Database

This chapter provides examples of the following backup and restore operations:

- Backing up and restoring a SQL Server database from one volume to another
- Backing up a SQL Server database to tape and restoring it
- Backing up and restoring a transaction log
- Backing up and restoring with generation management
- Backing up and restoring a SQL Server database between remote sites
- Backing up and restoring with a multi-target configuration
- Restoring and recovering a SQL Server database by using a host that differs from the backup host
- Using the log-shipping function of a SQL Server database
- Backing up and restoring a SQL Server database in a cascade or multi-target configuration
- Backing up and restoring a SQL Server database by using a backup management product that cannot be linked with Protection Manager
- Backing up and restoring a SQL Server database by using file sharing
- Restoring data to a SQL Server instance that is different from the backup source

The rest of this section explains points to note when backing up and restoring SQL Server databases. For notes on operations that do not depend on a particular database, see section 5.3.

7.1.1 Precautions When Backing up and Restoring a Database

- When specifying the `-target` or `-f` option with the `drmsqlbackup` command, specify all databases which are contained in the same logical volume. The command will result in an error if all databases are not specified.
- When executing the `drmsqlrestore` command for a SQL Server database for which the name is changed, be sure to first detach the database to be restored. If you perform restoration operations without detaching the SQL Server database, the `drmsqlrestore` command will not terminate normally, and you may not be able to use the database after it has been restored.
- You can also specify the original ID for the backup ID of the `drmsqllogbackup` command. In this case, add `0:` to the beginning of the original ID, and specify it as an argument of the command, as you would with the backup ID. You can use this functionality when the target backup catalog has been accidentally deleted, and the backup ID cannot be determined. The following are usage examples:

  a) To view backup information for the transaction log executed for the original ID `0000000001`:

  ```
  PROMPT> drmsqllogbackup 0:0000000001 -v
  ```

  b) To delete backup information for the transaction log executed for the original ID `0000000001`:

  ```
  PROMPT> drmsqllogbackup 0:0000000001 -d
  ```

- The maximum number of databases that the `drmsqlbackup` command can back up at one time is 64. To back up 65 or more databases, perform separate executions of the `drmsqlbackup` command.
- Do not access the operation-target volume (for example, by using Windows Explorer) while a backup or restoration command is being executed in a VERITAS Cluster Server for Windows environment. Doing so might place the cluster resources offline, resulting in an error in the backup or restoration command.
- If the Protection Products backup command is executed for a database that precedes SQL Server 2000 Service Pack 3a, the command might terminate with an error and display an error message (`KAVX1212-E`, `KAVX0006-E`). This problem with SQL Server is outlined in Microsoft's Support Technical Information 891268. To avoid this problem, use SQL Server 2000 Service Pack 4 or a more recent version.
- If SQL Server 2000 Service Pack 4 is used in an IPF (Itanium Processor Family) environment, `hotfix902150` must be applied.

7.1.2 Precautions When Restoring a Database

- When you attempt to restore an SQL Server system database (master, model, msdb, or distribution), the SQL Server service to be restored needs to be stopped to recover the system database. The database to be restored cannot be accessed temporarily.
- During restoration of the SQL Server system databases (master, model, and msdb), do not connect to the SQL Server. If you connect to a database that is being restored, checking of the process status is repeated by the number of times specified in the parameter (number of retries and retry interval for checking the process status) of the Protection Products configuration definition file (`init.conf`). In this case, if the user is disconnected during repeated checks of the process status, restoration will resume.
- When performing a restoration, if the name of the drive that makes up the SQL Server database is different from the backed up name, restoration terminates with an error. Before performing restoration, with the `drmsqlcat` command or SQL Server Enterprise Manager, check the corresponding drive name of the restoration destination.
- When you specify the `-nochk_host` option to perform restoration on a different host from that where the data was backed up from, make sure that backup data is restored on the appropriate host, because a validity check of the host names in the backup catalog is not performed during the restoration.
- When you restore on a SQL Server database whose name has been changed, the target SQL Server database must be detached. If the SQL Server database is restored without detach processing, the command fails and the SQL Server database might not be able to be used after restoration. If a SQL Server database becomes unusable, detach the database, and then restore it again.
- After a database is restored, the owner of the database is changed to the user who performed the restoration. To change the owner, use SQL Server Enterprise Manager to attach the database again, or use the `sp_changedbowner` system stored procedure.
- When you perform restore operations using a template catalog, execute the `drmsqlrestore` command with the `-template` option.
- For non-cluster configurations, before performing restoration, make sure that the files and directories on the primary volume are not being used by other applications, as Protection Manager mounts and unmounts the primary volume during restoration processing. If a file or directory on the primary volume is being used by another application, cache synchronization during unmount processing will fail, and restoration processing will terminate in an error.

  In particular, keep the following in mind:

  - When using the command prompt to move a drive to the primary volume, be sure to close the command prompt window, as if you move a drive to a volume other than the primary volume, an error will occur during the unmount operation.
  - If a location within a drive of the primary volume is open in Explorer, either move the mount point of the drive to one other than the primary volume, or close Explorer.
  - If a location within a drive of the primary volume is open from an external computer, terminate the open application.
  - The primary volume may be opened by a service or resident process-monitoring program. In this case, be sure to terminate any such monitoring programs.

- Before performing restoration, check that all applications accessing the database to be restored are terminated. If any applications connected to the database are running, roll-forward attempts after metafile application will fail, and restoration cannot be performed correctly.

  For example, when ODBC sections cannot be established, if an application that performs retries is running, since an ODBC section establishment request is issued during the roll-forward after metafile application, the roll-forward will fail.

- When using Enterprise Manager to view a database to be backed up, before performing restoration, either cut the connection to the database in Enterprise Manager, or terminate Enterprise Manager. If restoration is performed while Enterprise Manager is connected to the database server, restoration will fail because the database is still being used.

- Make sure that all the restoration target databases are in a state where restoration can be performed. For details on the database status where restoration can be performed, see section 7.1.10.2. If the database is in the `unconfirmed` status, it will be deleted and then restored automatically. If even one of the restoration target databases is in a state where restoration cannot be performed, restoration might not be performed appropriately. In such a case, delete that database, and then re-execute the restoration command.

  For example, when restoration is executed without specifying the `-undo` option, after restoration the database changes to a reading status. If the restoration command is re-executed while the database is in this status, the command will terminate with an error. In such a case, delete the database and then execute the restoration command.

- When the `drmsqlrestore` command executes, it uses SQL Server with minimum startup. If the Windows firewall functionality has been set in the database server, then when executing the `drmsqlrestore` command to perform a database restoration that includes the system databases (master, model, and msdb), a dialog box might be displayed asking if the Windows firewall is to block SQL Server communication. If this dialog box is displayed, select **Do not block**. The `drmsqlrestore` command will continue processing without problems even if you do not respond to this dialog box.

7.1.3 Precautions When Recovering a Database

- It may take a while for the consistency check processing performed within the `drmsqlrecover` command to complete. In this case, you can use a parameter of the Protection Manager configuration definition file (`init.conf`) to specify whether this check processing is to be performed. For more details about options to select check processing, see section 4.3.6.

- The number of databases that can be displayed by using the `drmsqlrecovertool` is 128. If more than 128 databases exist, use the `drmsqlrecover` command to recover those databases that are not displayed.

7.1.4 Precautions When Backing up and Restoring Multiple Databases

If one instance contains multiple databases, the unit by which volumes can be backed up and restored differs depending on the primary volume storing each database.

- When multiple databases are stored in the same volume:

  Multiple databases in the same volume can be backed up or restored at the same time.

- When multiple databases are stored in the multiple volume:

  You can specify database names to back up or restore particular user databases within an instance.

7.1.4.1 When Multiple Databases Are Stored in the Same Volume

Databases in the same volume can be backed up or restored at the same time.

Figure 7.1 shows backup and restoration for databases stored on the same volume.

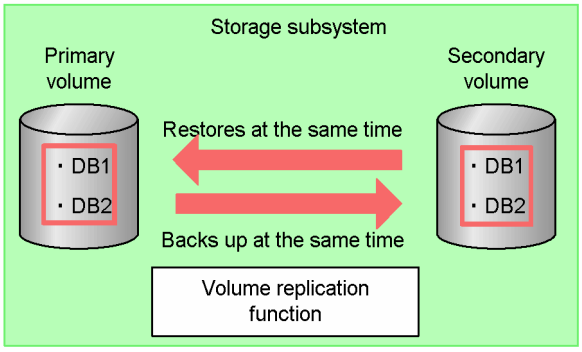


Figure 7.1 Backing up and Restoring Multiple Databases (Same Volume Configuration)

7.1.4.2 When Multiple Databases Are Stored in Multiple Volumes

You can specify database names to back up or restore particular user databases within an instance. When one instance contains multiple user databases, you can use Protection Manager to specify a particular instance name, and back up all user databases under that instance. To use this functionality, however, you must move or delete the sample database created during SQL Server installation. As a result, if one instance contains multiple user databases to be backed up, obtain backups by specifying multiple database names for `TARGET`, in the operation definition file for extended commands.

You can specify database names to back up or restore particular user databases within an instance.

Figure 7.2 shows backup and restoration for databases stored on the multiple volume.

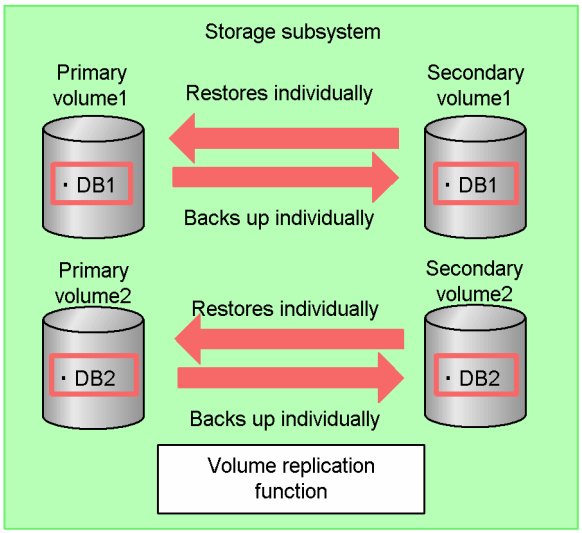


Figure 7.2 Backing up and Restoring Multiple Databases (Multiple Volume Configuration)

Note: When one instance contains multiple user databases, you can use Protection Manager to specify a particular instance name, and back up all user databases under that instance. To use this functionality, however, you must move or delete the sample database created during SQL Server installation. As a result, if one instance contains multiple user databases to be backed up, obtain backups by specifying multiple database names for `TARGET`, in the operation definition file for extended commands.

Protection Manager treats as restored any databases restored to the primary volume by data restored to the secondary volume from tape.

In the operations shown in Figure 7.3, the data restored from tape is stored on secondary volume 2, but when data on secondary volume 1 is restored to primary volume 1, Protection Manager treats this as a completed restoration. As such, `EX_DRM_SQL_RESTORE` cannot be used to restore DB2 on secondary volume 2 to primary volume 2. To restore the contents of secondary volume 2 to primary volume 2, perform the procedure again from tape restoration.

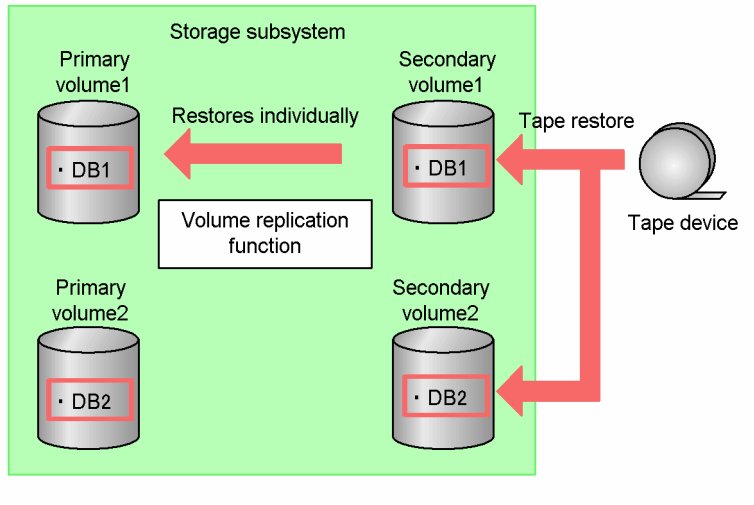


Figure 7.3 Tape Restoring Multiple Databases (Multiple Volume Configuration)

7.1.5 Database Status When Restoring and Recovering

When Protection Manager is used to restore data from a secondary volume to a primary volume, the status of the database becomes either loading or standby.

- Loading status (Reading)

  The database can neither be viewed nor updated.

- Standby status (Read-only)

  The database can be viewed, but not updated.

Transaction logs can be applied to databases in the loading status and standby status. For details about operation using transaction logs, see section 7.5.

Backup cannot be performed for databases in the loading status and standby status. To be backed up, a database in the loading status or standby status needs to be changed to a status that allows viewing and updating. To allow viewing and updating of a database in the loading status or standby status, use the `drmsqlrecovertool` command or `drmsqlrecover` command to recover the database. These commands can also be used to toggle the status of a database between loading and standby.

Figure 7.4 shows database statuses during restoration and recovery.

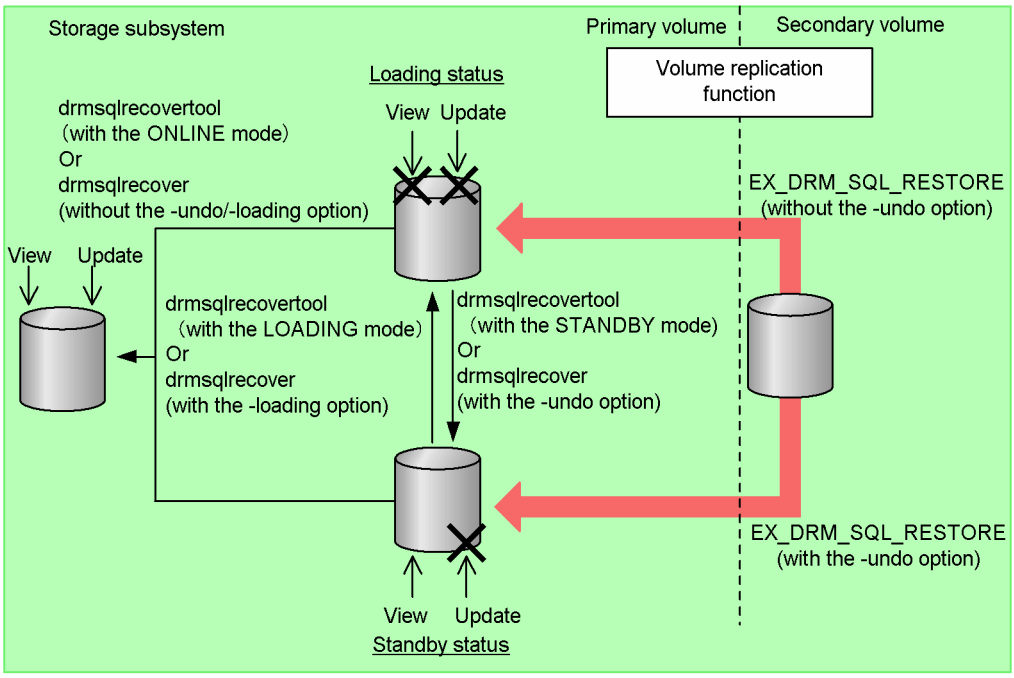


Figure 7.4 Database Statuses During Restoration and Recovery

Note that when a database in the standby status is viewed by Enterprise Manager, before executing the `drmsqlrecover` command or `drmsqlrecovertool` command, either cut the connection to the database in Enterprise Manager, or terminate Enterprise Manager. Since the database is locked when the `drmsqlrecover` command or `drmsqlrecovertool` command is executed while Enterprise Manager is connected to the database server, the command will terminate in an error.

7.1.6 Requirements for Transaction Log Backup

- Before transaction log backup, use the `drmsqlinit` command to set the log backup file storage directory.
- The backup catalog acquired by the `drmsqlbackup` command exists. When the `drmresync` command is executed, the backup catalog is deleted, so transaction log backup cannot be performed. When the `-no_cat` option is specified, transaction log backup can be performed even if the backup catalog does not exist.
- The instance to be backed up must be running.
- If the transaction log is corrupt, transaction log backup cannot be performed. If the KAVX1344-E error message is output when the `drmsqllogbackup` command with the `-no_truncate` option is executed, the transaction log might be corrupt.
- The recovery model of the database is Full or Bulk_Logged. (The recovery model Simple of the database is not included.)

7.1.7 Precautions for Transaction Log Backups

- `drmsqllogbackup` command cannot be applied to the system databases (`master`, `msdb`, `model`, `tempdb`, and `distribution`).
- Once a database has been restored, the log backup for different recovery paths are mixed and displayed.
- Specify the latest backup ID when you perform a transaction log backup by using the `drmsqllogbackup` command. If you specified the instance name, the latest backup ID is used.
- A transaction log backup is performed from the full backup that was last taken. However, when `LOCAL_BACKUP` in the user script file used during backup is set to `NO`, since a backup ID is not issued, the start point for log backup is not switched.

  As a result, the log backup information displayed by using the `drmsqllogbackup -v` command becomes inappropriate.

  Figure 7.5 shows the example when `LOCAL_BACKUP` is set to `YES` and the secondary volume on the local site is backed up. In this example, the point A is used as the start point and log backup 1 and 2 are obtained.

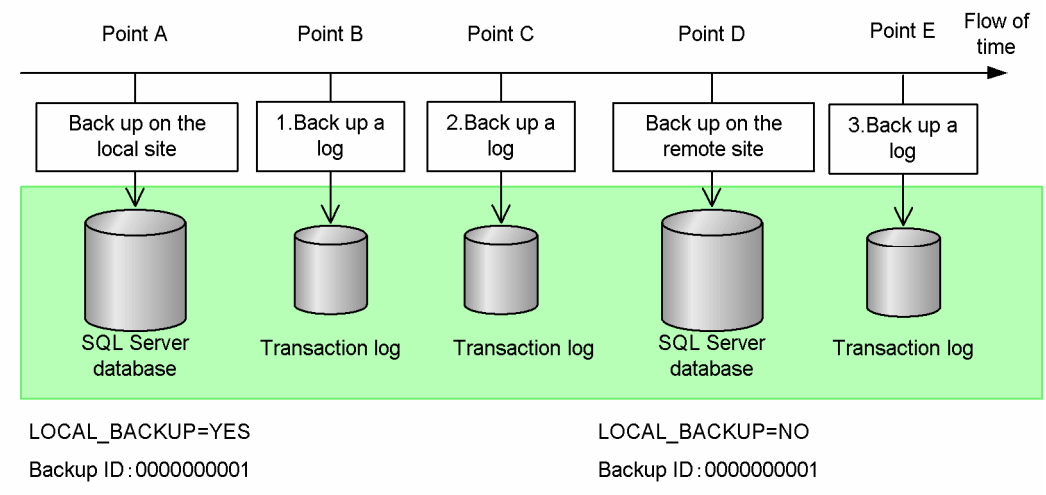


Figure 7.5 When `LOCAL BACKUP` is set to `NO` and the Secondary Volume on the Local Site is Backed up

Subsequently, the secondary volume on the remote site is backed up when `LOCAL_BACKUP` is set to `NO`.

The log backup 3 is obtained, with the backup ID `0000000001`.

When the `drmsqllogbackup -v` command is executed, log backup 1, 2, and 3 are displayed for the backup ID `0000000001`. If you perform restore operations by using the backup ID `0000000001`, log backup 3 cannot be recovered on the local site (however, log backup 3 can be recovered on the remote site).

Also, if an operation is performed on the remote site due to problems on the local site, the display of the `drmsqllogbackup -v` command on the remote site is inaccurate because the backup catalog does not exist on the remote site.

- Roll-forwards using a transaction log file which is marked by the `BEGIN TRANSACTION MARK` command of SQL Server, and Roll-forwards with a specified recovery time are not supported.

7.1.8 Precautions Regarding Sequences of Transaction Logs

To obtain backups of transaction logs, Protection Manager provides `EX_DRM_SQL_TLOG_BACKUP`. Protection Manager manages internal backups from primary volumes to secondary volumes, as well as transaction log backup sequences. As such, keep the following in mind.

7.1.8.1 Transaction Log Backups After Roll-Forward

After applying a transaction log backed up by Protection Manager, you must use Protection Manager commands to back up transaction logs, using one of the following methods:

- Apply backups of all transaction logs, using roll-forward (Figure 7.6)

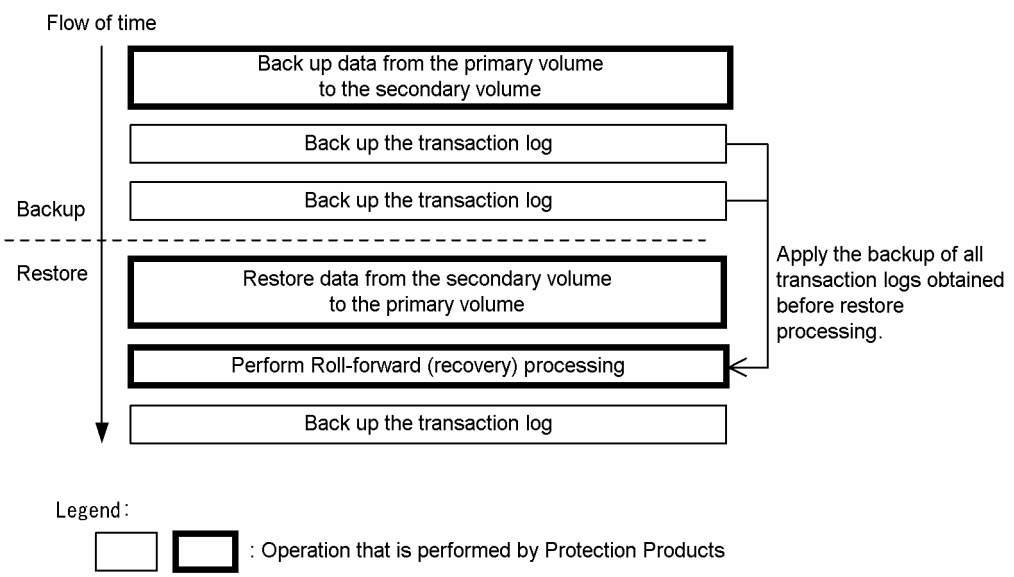


Figure 7.6 Transaction Log Backups After Roll-Forward (1)

- After performing roll-forward, use Protection Manager to perform backup from the primary volume to the secondary volume, and then obtain a backup of the transaction log files. (See Figure 7.7.)

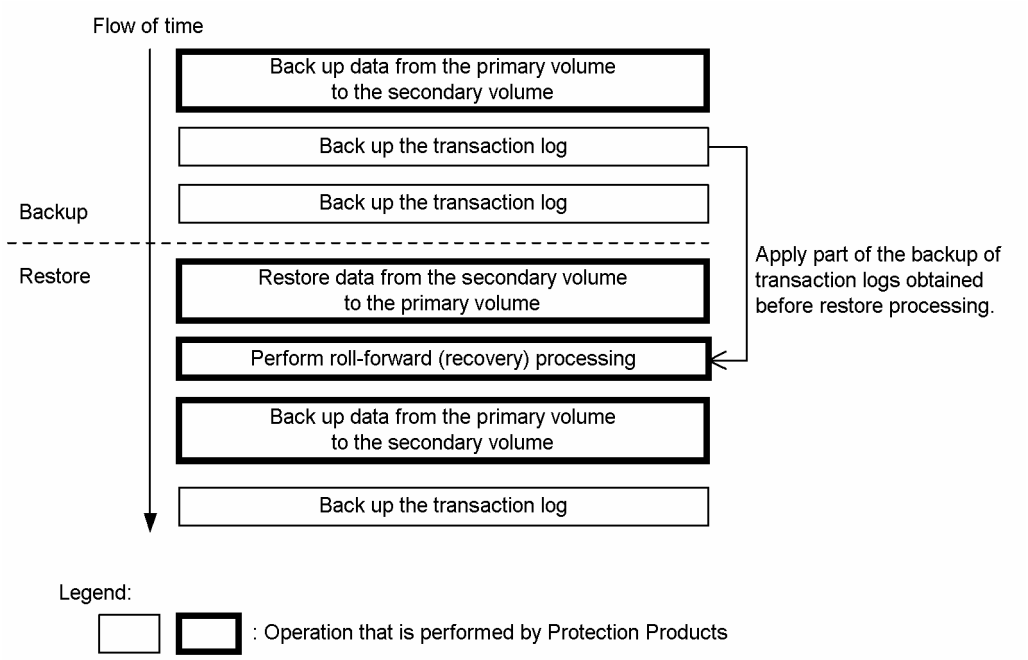


Figure 7.7 Transaction Log Backups After Roll-Forward (2)

In this case, when the second backup from the primary volume to the secondary volume is performed, the contents of the secondary volume are updated. To perform restoration again, to the secondary volume, the first backup from the primary volume to the secondary volume, first perform restoration from the tape device to the secondary volume, and then perform restoration from the secondary volume to the primary volume.

An error will occur if you try to back up the transaction log files without backing up the primary volume to the secondary volume (Figure 7.8).

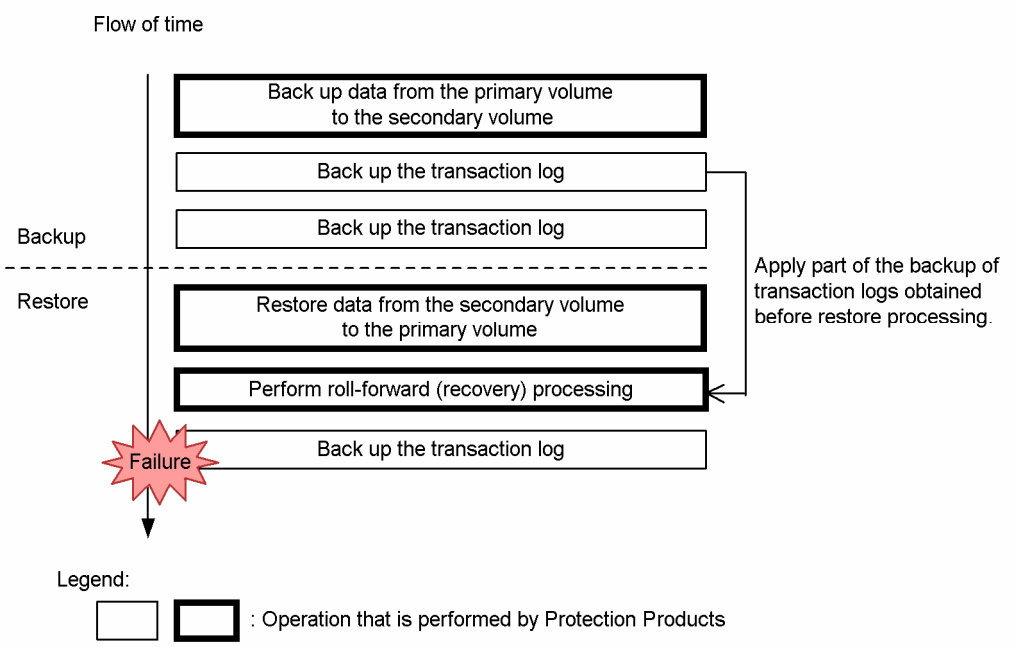


Figure 7.8 Transaction Log Backups After Roll-Forward (3)

7.1.9 Precautions Regarding Application of Transaction Logs

Application of transaction logs enables a database to be recovered from a backup at a certain point, or when the metafile is damaged. For such a recovery, the transaction logs must be backed up continuously. Also, a recovery cannot be performed if any transaction logs are missing.

In addition, transaction log files must be applied in the order they were backed up. This section describes the order of applying transaction logs when using the following procedures:

- Recovering the SQL Server database from the last backup data
- Recovering the SQL Server database from backup data prior to the last backup data

7.1.9.1 Recovering the SQL Server Database from the Last Backup Data

When you recover the SQL Server database from the last backup data, restore a previous backup from the primary volume to the secondary volume and then apply transaction logs.

The following figure shows an example of applying transaction log backups after restoring a previous backup from the primary volume to the secondary volume.

1. Transaction log backups at point B can be applied after the previous volume backup (point A) is restored.
2. Transaction log backups at point C can be applied after the previous transaction log backup (point B) is applied.

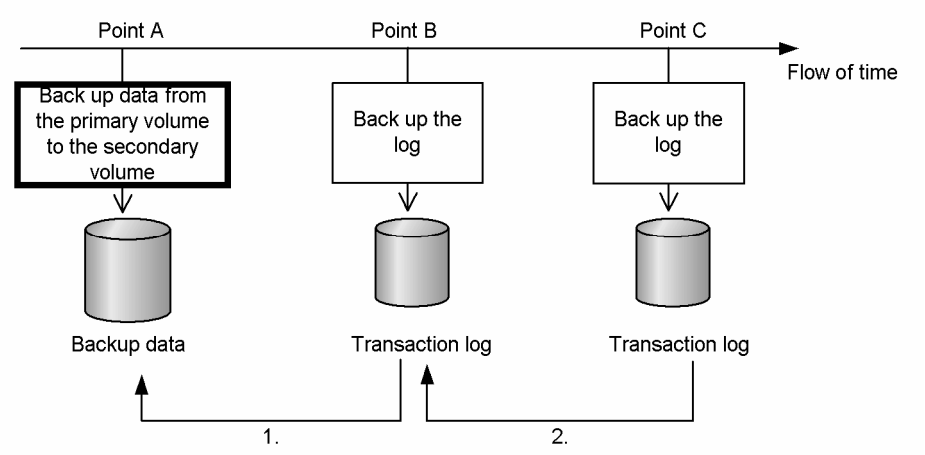


Figure 7.9 Order of Applying Transaction Log Backups (1)

7.1.9.2 Recovering the SQL Server Database from Backup Data Prior to the Last Backup Data

You can recover the SQL Server database from backup data prior to the last backup data when there is a problem in the latest backup data. For example, when you recover the SQL Server database from the second-last backup data, apply the transaction logs of the second-last backup from the primary volume to the secondary volume and then apply transaction log backup.

The following figure shows an example of applying a transaction log backup after the last transaction log obtained for the second-to-last backup from the primary volume to the secondary volume is applied. By applying the backups as shown in the example, even if a problem exists with the latest backup data, restoration can be performed to the most recent state, using the second-to-last backup data.

1. Transaction log backups at point B can be applied after the previous volume backup (point A) is restored.
2. Transaction log backups at point C can be applied after the previous transaction log backup (point B) is applied.
3. The transaction log backup at point E can be applied after the last transaction log obtained for the backup from the primary volume to the secondary volume is applied (point C).

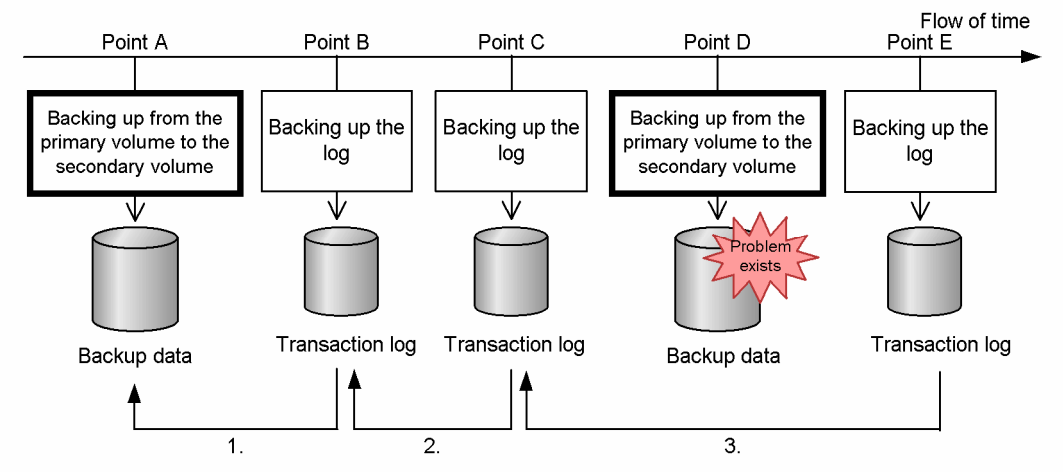


Figure 7.10 Order of Applying Transaction Log Backups (2)

7.1.10 SQL Server Database Requirements for Executing Commands

To execute the SQL `drmsql`***xxx*** command for operating the server database, the SQL Server service status, database status and database type must be taken into account. Table 7.1, Table 7.2, and Table 7.4 explain the requirements for executing commands.

7.1.10.1 SQL Server Database Requirements for Executing Commands

Table 7.1 Service Status

	Service status	
Commands	**Running**	**Stopping**
`drmsqlbackup`	Executable	(Not applicable)
`drmsqlcat`	Executable	Executable
`drmsqldisplay` (when the `-refresh` option is specified)	Executable	(Not applicable)
`drmsqldisplay` (when the `-refresh` option is not specified)	Executable	Executable
`drmsqlinit`	Executable	(Not applicable)
`drmsqllogbackup`	Executable	(Not applicable)
`drmsqlrecover`	Executable	(Not applicable)
`drmsqlrecovertool`	Executable	(Not applicable)
`drmsqlrestore` (when the system database# is included)	Executable	Executable
`drmsqlrestore` (when the system database# is not included)	Executable	(Not applicable)

Indicates the master, model, and msdb databases.

7.1.10.2 Database Status

Table 7.2 SQL Server Database Status Where a Command Can Be Executed (1)

	SQL Server database status				
Commands	**Online**	**Offline**	**Reading**	**Read-only**	**Unconfirmed**
`drmsqlbackup`	Yes (DB)	No	No	No	No
`drmsqlcat`	Yes	Yes	Yes	Yes	Yes
`drmsqldisplay` (when the `-refresh` option is specified)	Yes (DB)	Yes (DB)	Yes (DB)	Yes (DB)	Yes (DB)
`drmsqldisplay` (when the `-refresh` option is not specified)	Yes	Yes	Yes	Yes	Yes
`drmsqlinit`	Yes	Yes	Yes	Yes	Yes
`drmsqllogbackup` (when the `-no_truncate` option is specified)	Yes (DB)	No	No	No	Yes (DB)
`drmsqllogbackup` (when the `-no_truncate` option is not specified)	Yes (DB)	No	No	No	No
`drmsqlrecover` (when the `-undo` option is specified)	Yes	No	Yes (DB) (becomes read-only)	Yes (DB)	No
`drmsqlrecover` (when the `-loading` option is specified)	Yes	No	Yes (DB)	Yes (DB) (becomes reading)	No
`drmsqlrecover` (when the `-undo` and `-loading` options are not specified)	Yes	No	Yes (DB) (becomes online)	Yes (DB) (becomes online)	No
`drmsqlrecovertool` (when the Loading option is specified)	No	No	Yes (DB)	Yes (DB) (becomes reading)	No
`drmsqlrecovertool` (when the Standby option is specified)	No	No	Yes (DB)	Yes (DB) (becomes read-only)	No
`drmsqlrecovertool` (when the Online option is specified)	No	No	Yes (DB) (becomes online)	Yes (DB) (becomes online)	No
`drmsqlrestore` (when the `-undo` option is specified)	Yes (DB) (becomes read-only)	Yes (DB) (becomes read-only)	Conditional[1] (becomes read-only)	Yes (DB) (becomes read-only)	Yes (DB)[2] (becomes read-only)
`drmsqlrestore` (when the `-undo` option is not	Yes (DB) (becomes	Yes (DB) (becomes	Conditional[1] (becomes	Yes (DB) (becomes	Yes (DB)[2] (becomes

	SQL Server database status				
specified)	reading)	reading)	reading)	reading)	reading)

Legend:

Yes (DB): The command can be executed and the operations are performed on the database. The database status after the operations is displayed in parenthesis.

Conditional: The command can be executed in certain conditions. (Operations are performed on the database. The database status after the operations is displayed in parenthesis) The command can be executed when restoration of the system database is included.

Yes: The command can be executed; however, no operation is performed on the database.

No: The command cannot be executed.

[1]The command can be executed when restoration of the system databases (master, model, and msdb) is included. If you restore the user database only, delete the database in a state where a restoration command cannot be executed, and then re-execute the restoration.

[2]If the database is in the `unconfirmed` status, it will be deleted and then restored automatically.

Table 7.3 SQL Server Database Status Where a Command Can Be Executed (2)

	SQL Server database status				
Commands	**Offline and unconfirmed**	**Reading and unconfirmed**	**Read-only and offline**	**Read-only and unconfirmed**	**Read-only, offline, and unconfirmed**
`drmsqlbackup`	No	No	No	No	No
`Drmsqlcat`	Yes	Yes	Yes	Yes	Yes
`drmsqldisplay` (when the `-refresh` option is specified)	Yes (DB)	Yes (DB)	Yes (DB)	Yes (DB)	Yes (DB)
`drmsqldisplay` (when the `-refresh` option is not specified)	Yes	Yes	Yes	Yes	Yes
`Drmsqlinit`	No	No	No	No	No
`drmsqllogbackup` (when the `-no_truncate` option is specified)	No	No	No	No	No
`drmsqllogbackup` (when the `-no_truncate` option is not specified)	No	No	No	No	No
`drmsqlrecover` (when the `-undo` option is specified)	No	No	No	No	No
`drmsqlrecover` (when the `-loading` option is specified)	No	No	No	No	No
`drmsqlrecover` (when the `-undo` and `-loading` options are not specified)	No	No	No	No	No
`drmsqlrecovertool` (when the Loading option is specified)	No	No	No	No	No
`drmsqlrecovertool` (when the Standby option is specified)	No	No	No	No	No
`drmsqlrecovertool` (when the Online option is specified)	No	No	No	No	No
`drmsqlrestore` (when the `-undo` option is specified)	Yes (DB)# (becomes read-only)	Yes (DB)# (becomes read-only)	Yes (DB) (becomes read-only)	Yes (DB) (becomes read-only)	Yes (DB) (becomes read-only)

	SQL Server database status				
drmsqlrestore (when the -undo option is not specified)	Yes (DB)# (becomes reading)	Yes (DB)# (becomes reading)	Yes (DB) (becomes reading)	Yes (DB) (becomes reading)	Yes (DB) (becomes reading)

Legend:

Yes (DB): The command can be executed and the operations are performed on the database. The database status after the operations is displayed in parenthesis.

Conditional: The command can be executed in certain conditions. (Operations are performed on the database. The database status after the operations is displayed in parenthesis)

Yes: The command can be executed; however, no operation is performed on the database.

No: The command cannot be executed.

If the database is in the unconfirmed status, it will be deleted and then restored automatically.

7.1.10.3 Database Type

Table 7.4 Relationship Between the Type of SQL Server Database and Executable Commands

	SQL Server database type				
Commands	**System database**				**User database**
	master	**msdb**	**model**	**tempdb**	
drmsqlbackup	Yes (DB)	Yes (DB)	Yes (DB)	No (terminates with an error)	Yes (DB)
drmsqlcat	Yes	Yes	Yes	Yes	Yes
drmsqldisplay	Yes	Yes	Yes	Yes	Yes
drmsqlinit	Yes	Yes	Yes	Yes	Yes
drmsqllogbackup	No (No operation is performed)[1]	No (No operation is performed)[1]	No (No operation is performed)[1]	No (No operation is performed)[1]	Yes (DB)[2]
drmsqlrecover	No (No operation is performed)	No (No operation is performed)	No (No operation is performed)	No (terminates with an error)	Yes (DB)
drmsqlrecovertool	No (No operation is performed)	No (No operation is performed)	No (No operation is performed)	No (terminates with an error)	Yes (DB)
drmsqlrestore	Yes (DB)	Yes (DB)	Yes (DB)	Yes	Yes (DB)

Legend:

Yes (DB): The command can be executed and the operations are performed on the database. The database status after the operations is displayed in parenthesis.

Yes: The command can be executed, however, no operation is performed on the database.

No: The command cannot be executed. The command behavior is displayed in parenthesis.

[1]When the `-target` option or `-t` option is specified, the command terminates with an error.

[2]The command is executable only when the recovery type of the database is Full or Bulk_Logged.

7.2 Backing up and Restoring Data Between Volumes

7.2.1 System Configuration

In a configuration that uses only one server, you use commands to back up and restore data between the primary volume and secondary volume. Install Protection Products on the database server and execute the appropriate commands.

Figure 7.11 shows a system configuration for backing up and restoring data between volumes.

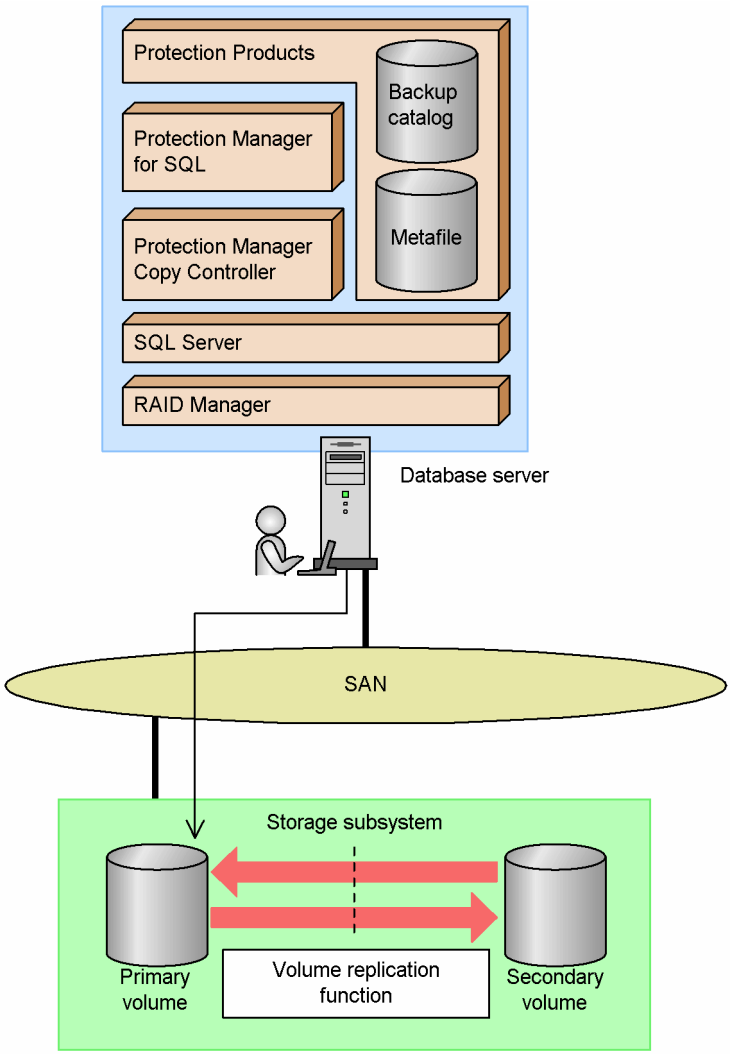


Figure 7.11 Backing up and Restoring Data Between Volumes (SQL Server Database Backup)

7.2.2 SQL Server Backup Processing Flow

The processing flow for backing up a SQL Server database to the secondary volume and restoring it to the primary volume on a single-server system is shown below. Commands are executed on the database server.

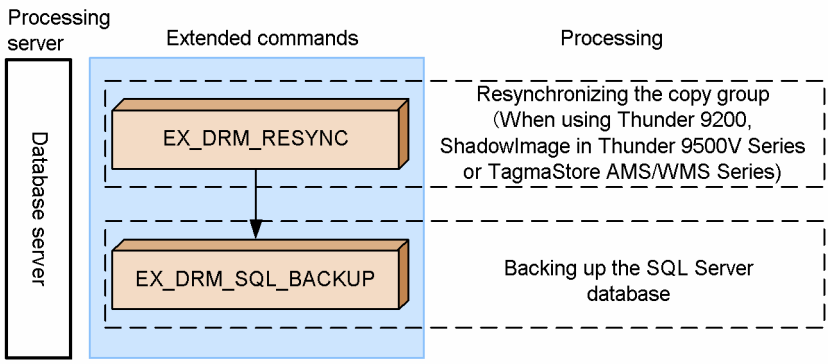


Figure 7.12 Procedure for Backing up a SQL Server Database to the Secondary Volume

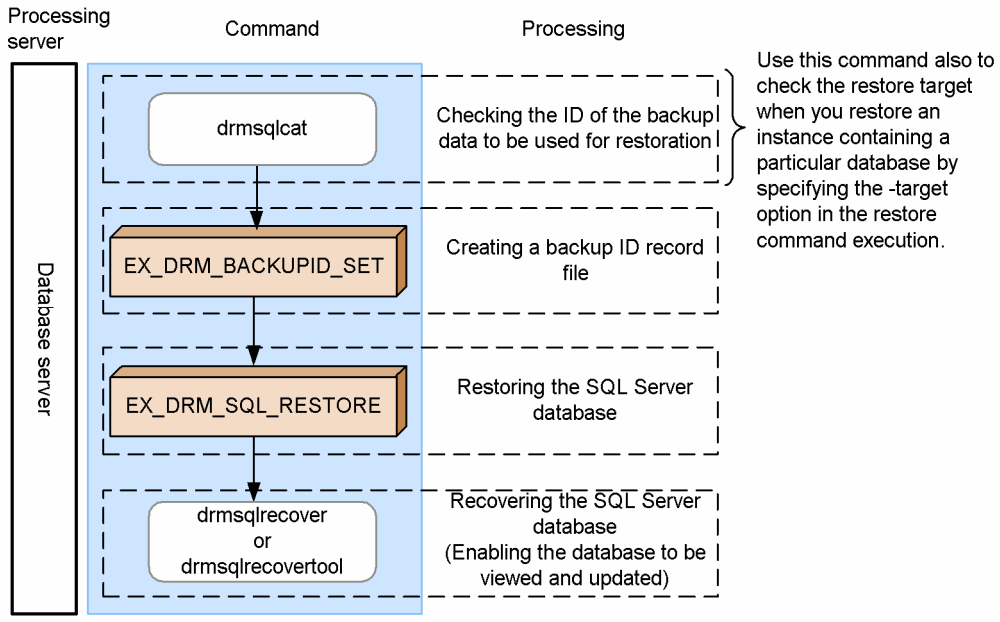


Figure 7.13 Procedure for Restoring a SQL Server Database to the Primary Volume

7.2.3 Backing up a SQL Server Database to a Secondary Volume

This section provides an example of performing an online backup of a SQL Server database between volumes.

To back up the SQL Server database:

1. Resynchronize the copy group.

   If you are using a Thunder 9200 Series, and ShadowImage with a Thunder 9500V Series or a TagmaStore AMS/WMS Series, execute `EX_DRM_RESYNC` to resynchronize the copy group. You can facilitate backup processing by resynchronizing the copy group in advance.

```
DBServer > EX_DRM_RESYNC Operation_A -cg VG01,vol01 -copy_size 7
```

2. Back up the SQL Server database onto the secondary volume.

   Perform an online backup of the SQL Server database. To perform the backup, execute `EX_DRM_SQL_BACKUP` with the operation ID `Operation_A`:

```
DBServer > EX_DRM_SQL_BACKUP Operation_A -system
```

7.2.4 Restoring a SQL Server Database to a Primary Volume

This section provides an example of recovering a SQL Server database when the backup data is stored on the secondary volume. In this example, you resynchronize the secondary volume with the primary volume to restore the data on the primary volume. The operation ID is `Operation_A`.

7.2.4.1 Restoring a SQL Server Database

To restore the SQL Server database:

1. Check the backup ID of the backup you want to use for restoration.

   Check the backup ID of the backup data to be used for restoring. To check the backup ID, execute the `drmsqlcat` command on the database server. Here, the predetermined instance (default) is used as the instance name of SQL Server.

```
DBServer >drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: default ORIGINAL-ID: 0000000001
START-TIME:2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: DBServer
T DB  OBJECT   FILE                              FS  DG  DEVICE  COPY-GROUP
M DB1 METAFILE D:\MSSQL\DBServer\INSTANCE_1\META  D:  -    -           -
D DB1 DATAFILE E:\SQL\data1                        E:  -  Harddisk1  VG01,sqlvol01
T DB1 TRANSACT  E:\SQL\tlog                        E:  -  Harddisk2  VG01,sqlvol01
:
```

   If you perform restoration in units of instances, including a specific database, by executing the restoration command with the `-target` option specified, check the restoration target.

2. Create a backup ID file.

To create a backup ID file, execute EX_DRM_BACKUPID_SET with the backup ID.

```
DBServer > EX_DRM_BACKUPID_SET Operation_A -backup_id 0000000001
```

3. Restore data in the secondary volume to the primary volume.

   To restore the primary volume by resynchronizing the secondary volume with the primary volume, execute EX_DRM_SQL_RESTORE on the database server.

```
DBServer > EX_DRM_SQL_RESTORE Operation_A -resync
```

   When EX_DRM_SQL_RESTORE is executed with the -undo option specified, restoration is performed while the database is in the standby status, in which the database is read-only. If the -undo option is omitted, after restoration, the status of the database changes to loading, and the database cannot be viewed. Note that the -undo option cannot be used unless the UNDO file storage directory has been set using the drmsqlinit command.

   Also, do not connect to SQL Server while executing EX_DRM_SQL_RESTORE.

4. Recover the SQL Server database.

   To recover the SQL Server database, execute the drmsqlrecover or drmsqlrecovertool command on the database server.

Section 7.2.4.2 shows an example of the drmsqlrecover command and drmsqlrecovertool dialog box. For details on the drmsqlrecover command or the drmsqlrecovertool dialog box, see the *HiCommand Protection Manager Command Reference* manual.

7.2.4.2 Using the drmsqlrecover Command to Recover a SQL Server Database

The drmsqlrecover command is useful for recovering particular databases and recovering multiple volumes at once.

To use the drmsqlrecover command to recover a SQL server database, specify an instance name to execute the drmsqlrecover command. The following command illustrates execution of the drmsqlrecover command to recover a SQL Server database:

```
DBServer > drmsqlrecover default
KAVX0001-I The drmsqlrecover command will now start.
KAVX0002-I The drmsqlrecover command will now end.
DBServer >
```

When you execute the drmsqlrecover command with an instance name specified, the database becomes accessible, and the recovery is complete.

When you execute the drmsqlrecover command with the -undo option specified, the database is recovered in the standby status (read-only and usable). When you execute with the -loading option specified, the database has the loading status and is not accessible. Note that the -undo option cannot be used unless the UNDO file storage directory has been set using the drmsqlinit command.

7.2.4.3 Using the drmsqlrecovertool Dialog Box to Recover a SQL Server Database

The `drmsqlrecovertool` command is useful for performing recovery and checking the status of a database at the same time.

To use the `drmsqlrecovertool` command to recover a SQL Server database:

1. Specify an instance name to execute the `drmsqlrecovertool` command:

```
DBServer > drmsqlrecovertool default
```

   When you execute the command, the `drmsqlrecovertool` dialog box is displayed (Figure 7.14).

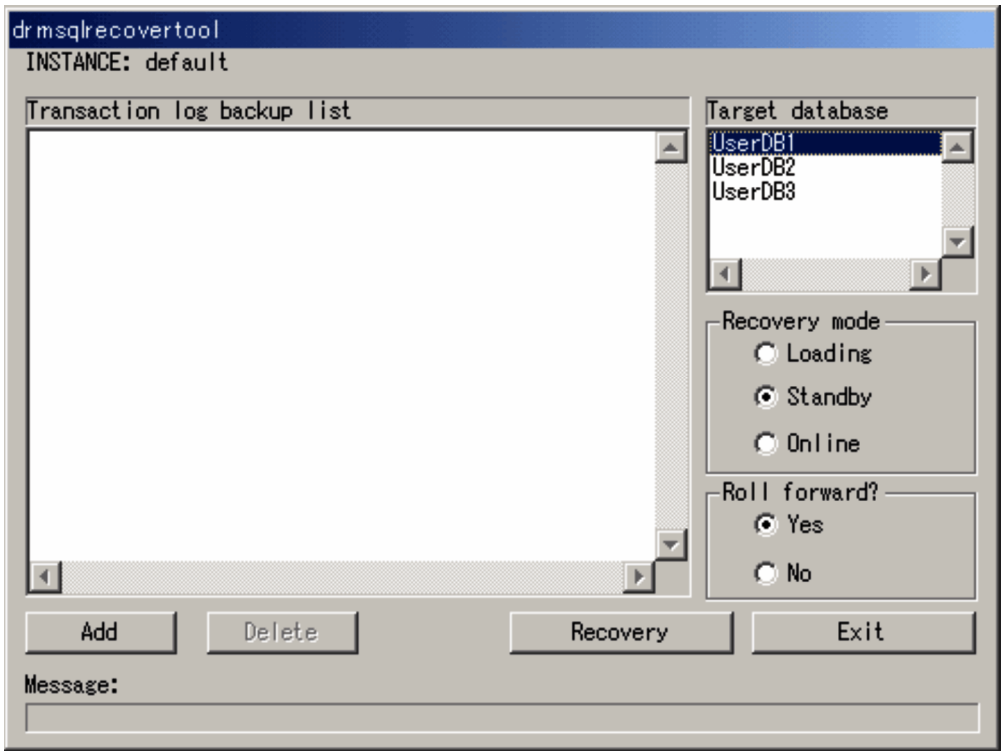


Figure 7.14 The `drmsqlrecovertool` Dialog Box

2. Select a recovery destination database in **Target database**.
3. Check the status of the database after recovery.

   For **Recovery mode**, select the status of the recovered database.

   Standby: Select this to perform recovery in the standby status (read-only and usable). Note that you cannot select **Standby** unless the UNDO file storage directory is set using the `drmsqlinit` command.

   Loading: Select this to perform recovery in the loading status.

   Online: Select this to make the database fully accessible.
4. Select whether to perform roll-forward.

To apply transaction logs (a roll-forward), select **Yes** for **Roll forward?**. If no transaction log is displayed in **Transaction log backup list**, **Roll forward?** can be set to either **Yes** or **No**.

5. Click the **Recovery** button.

   If **Online** is selected for **Recovery mode**, * is appended to the file name displayed in **Target database**, and recovery is complete.

7.3 Backup and Restore Using a Tape Device

7.3.1 System Configuration

This section uses examples to describe how to use extended commands. The example in Figure 7.15 assumes that the system is configured to use one database server, but you can also configure the system to use multiple database servers.

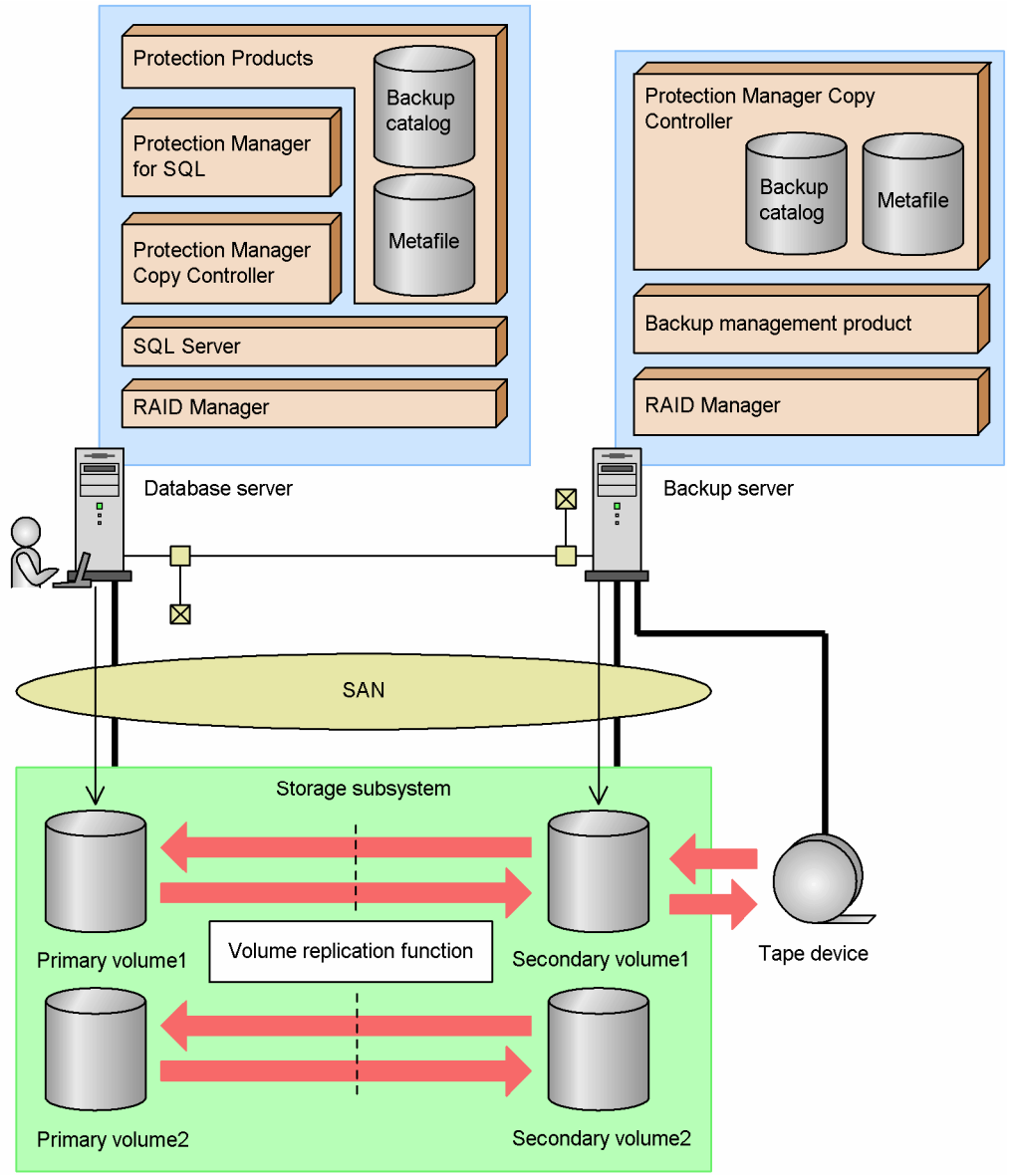


Figure 7.15 System Configuration for Backing up a SQL Server Database to Tape and Restoring from Tape

Protection Manager allows you to use a cluster configuration for database servers. If a failover cluster configuration is used for the database servers, the active server can be switched to the standby server when a failure occurs on the active server.

The examples in this section assume the following conditions:

- The primary volume is NTFS-formatted.
- The primary volume and the secondary volume are pair defined in the database server and the backup server.
- The backup management product is installed on the backup server.
- The parameters of the backup management product have been registered by using the `drmtapeinit` command.
- The FTP service on the backup server is started, and the system is set up so a user who has logged onto the OS of the database server can log onto the FTP server and transfer files to the FTP server. The FTP user ID is `admin`, and the FTP user password is `password`.
- Data for which an online back up was performed from the primary volume to the secondary volume was backed up to tape.
- The SQL Server instance (`INSTANCE_1`) exists, and services are running on database server A (`DBServer`).
- Temporary directory for Extended Commands and the VDI metafile storage directory are created by the database server and backup server.
- The secondary volume is not always mounted. It is mounted on the E drive (drive character `E:`) only while it is operating.
- The primary volume configured as a pair with the secondary volume to be backed up will not be backed up until the secondary volume is backed up onto tape.
- The `cscript //H:Cscript` command is executed from the command prompt and the host parameters are updated.

7.3.2 Processing Flow for Tape Device Backup

Figure 7.14 and Figure 7.15 show the flow of processing during backup of a SQL Server database to tape and restoration of the backed-up SQL Server database to the primary volume in a multiple server configuration system.

- When usual operation is in the split state, the copy group is first resynchronized, after which the data is backed up.
- When usual operation is in the pair state, it is unnecessary to resynchronize the copy group before backing up the data. After the data is backed up to tape, resynchronize the copy group to restore data to the initial state.

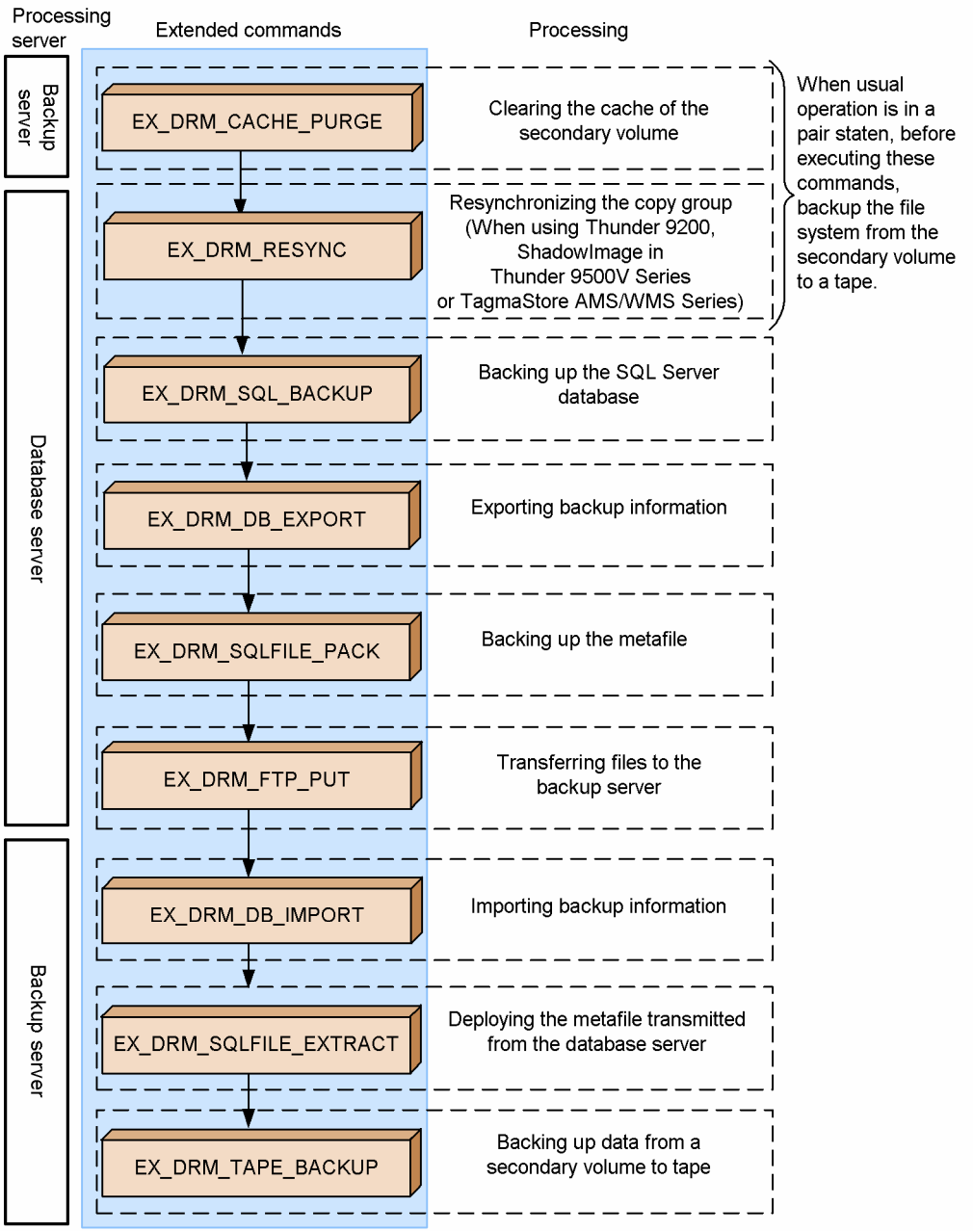


Figure 7.16 Procedure for Restoring a SQL Server Database to the Primary Volume

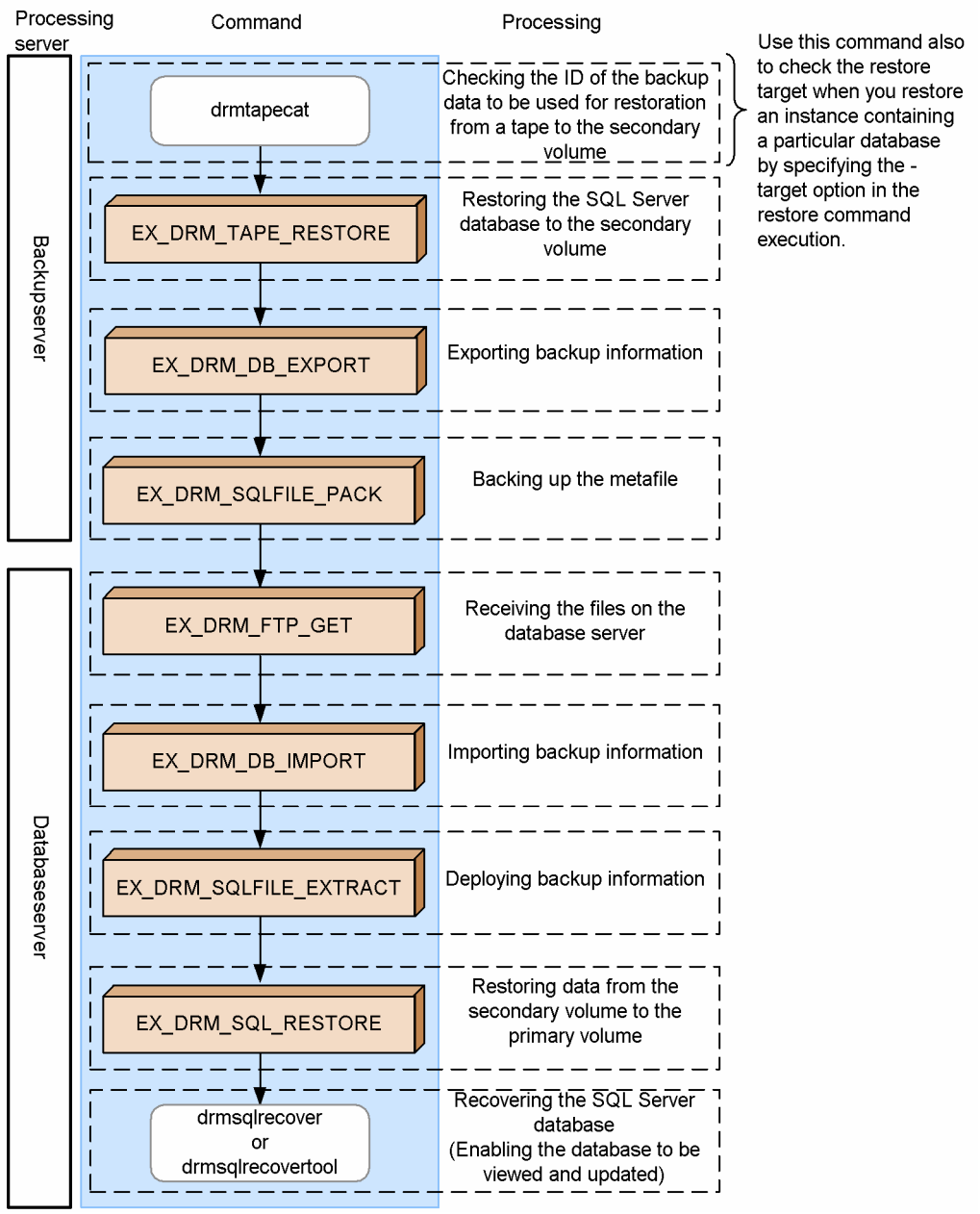


Figure 7.17 Procedure for Restoring a SQL Server Database from its Backup Tape

7.3.3 Backing up a SQL Server Database to Tape

This section provides an example of performing an online backup of a SQL Server database onto tape. In this example, an online backup of the `INSTANCE_1` instance of the `DBServer` database server is first performed to place a backup on the secondary volume, and then a backup is performed from the secondary volume to tape. The operation ID is `Operation_A`. Note that when Protection Manager is used in a cluster environment, be sure to set the `DRM_HOSTNAME` environment variable from the command prompt as follows before executing basic commands:

```
PROMPT> set DRM_HOSTNAME=virtual-server-name
```

7.3.3.1 Resynchronizing Copy Groups

When usual operation is in the split state, the copy group is first resynchronized, after which the data is backed up.

When usual operation is in the pair state, it is unnecessary to resynchronize the copy group before backing up the data. After the data is backed up to tape, resynchronize the copy group to restore data to the initial state.

Resynchronize the copy group.

1. For the copy pair configuration of multiple generations, check the copy group to be used next.

   To check the copy group to be used next, execute the `drmsqlcat` command on the backup server.

   If the copy group is not locked, the least recently backed up copy group is overwritten. You can execute the `drmcgctl` command to check the lock status of a copy group.

2. Clear the cache of the secondary volume.

   Before backing up SQL Server, clear the system cache on the backup server.

   Clear the cache of the secondary volume which belongs to copy groups `VG01,vol11` and `VG01,vol12`. To specify multiple copy group names, you can create a copy-group list file ahead of time, and simply specify the file. In this example, copy groups `VG01,vol11` and `VG01,vol12` are defined in the copy-group list file `CGLIST.txt`.

   ```
   BKServer > EX_DRM_CACHE_PURGE Operation_A -cg_file C:\temp\CGLIST.txt
   ```

3. Resynchronize the copy group.

   If you are using a Thunder 9200 Series, and ShadowImage with a Thunder 9500V Series or a TagmaStore AMS/WMS Series, execute `EX_DRM_RESYNC` to resynchronize the copy group. You can facilitate backup processing by resynchronizing the copy group in advance.

   ```
   DBServer > EX_DRM_RESYNC Operation_A -cg_file C:\temp\CGLIST.txt -copy_size 7
   ```

7.3.3.2 Backing up a SQL Server Database

To back up a SQL Server database:

1. Back up the SQL Server database from the primary volume to the secondary volume.

   Perform an online backup of the SQL Server database. To perform the backup, execute `EX_DRM_SQL_BACKUP` with the operation ID `Operation_A`.

   When backing up all databases, including the system databases (master, model, and msdb), specify the `-system` option for the `EX_DRM_SQL_BACKUP` parameter (if the `-system` option is not specified, only user databases will be backed up).

   Note that when backup is performed for all databases, including system databases, `TARGET_NAME`, in the operation definition file, must be blank.

   ```
   DBServer > EX_DRM_SQL_BACKUP Operation_A -system
   ```

2. Check that backup was performed properly.

   To check that backup information exists for the date that backup was performed from the primary volume to the secondary volume, execute the `drmsqlcat` command on the database server.

3. Export the backup information to temporary files.

   To back up data from the secondary volume to tape, copy the backup information to the backup server. To export the backup information to temporary files, execute `EX_DRM_DB_EXPORT`. These temporary files are stored in the temporary folder for extended commands, on the database server.

   ```
   DBServer > EX_DRM_DB_EXPORT Operation_A
   ```

   After `EX_DRM_DB_EXPORT` is executed, a *backup-ID_database-ID*.dmp file is created on the database server, in the temporary folder for extended commands. Check that a file was created for the date on which the command was executed.

4. Back up the SQL Server's metafile created at the backup to the temporary folder.

   Back up the SQL Server's metafile created at the backup to the temporary folder for extended commands that is on the database server so that the SQL Server's metafile and temporary files can be transmitted collectively to the backup server. To back up the SQL Server's metafile, execute `EX_DRM_SQLFILE_PACK`.

   ```
   DBServer > EX_DRM_SQLFILE_PACK Operation_A
   ```

   After `EX_DRM_SQLFILE_PACK` is executed, an *`operation-ID_nnn`*`.dmp` file is created in the temporary folder for extended commands, on the database server. Check that a file was created for the date on which the command was executed.

5. Transmit the metafile and temporary files to the backup server.

   To transmit the SQL Server's metafile and temporary files collectively to the backup server, execute `EX_DRM_FTP_PUT` on the database server. In this example, the user ID to log on to the FTP server is `admin` and the login password is `password`. The metafile and temporary files are stored in the temporary folder for extended commands, on the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

6. Import the temporary files, which were transmitted from the database server, into the backup catalog on the backup server.

   To import the temporary files, which were transmitted from the database server, into the backup catalog, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

   After `EX_DRM_DB_IMPORT` is executed, an *`operation-ID`*`.bid` file is created in the *`FTP_HOME_DIR-value`*`\`*`FTP_SUB_DIR-value`*`\`*`operation-ID`*`\BK\` directory. Check that a file was created for the date on which the command was executed.

7. On the backup server, deploy the metafile that was transmitted from the database server.

   To deploy the metafile transmitted from the database server, execute `EX_DRM_SQLFILE_EXTRACT` on the backup server.

   At this time, only the latest metafiles are expanded on the backup server.

```
BKServer > EX_DRM_SQLFILE_EXTRACT Operation_A
```

8. Back up the data of the secondary volume to tape.

   To back up the data of the secondary volume, execute `EX_DRM_TAPE_BACKUP` on the backup server. In this example, the drive letter of the secondary volume is assumed to be `E:` in this example.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   Note that you can omit the `-mount_pt` option if the mount point defined in the configuration definition file for tape backup (such as `NBU.DAT`) matches a mount point that is actually free.

   When this backup is executed, the backup information is registered, with a new backup ID of `0000000002`, in the backup catalog.

7.3.4 Restoring a SQL Server Database from Tape

This section provides an example of restoring a SQL Server database backed up to tape. In this example, you first restore data from a tape to the secondary volume, and then you resynchronize the secondary volume with the primary volume to restore the data on the primary volume. The operation ID is `Operation_A`.

7.3.4.1 Restoring a SQL Server Database

To restore the SQL Server database:

1. Check the backup ID of the backup you want to use for restoration.

   Check the backup ID of the backup data to be used for restoring. To check the backup ID, execute the `drmtapecat` command on the backup server.

```
BACKUP-ID ORIGINAL-ID HOSTNAME BACKUP-OBJECT INSTANCE  SNAPSHOT TIME      EXPIRATION TIME
BACKUP-MEDIA
0000000002 0000000001 DBServer MSSQL     instance1 2003/08/11 19:42:15 2003/11/11 19:42:15
-
:
:
```

   If you perform restoration in units of instances, including a specific database, by executing the restoration command with the `-target` option specified, you must confirm the restoration target. In this case, execute the `drmtapecat` command with the following options specified:

   - `-o MSSQL` *instance-name*
   - `-backup_id` *backup-ID*

2. Restore data from its backup tape to the secondary volume.

   To restore data, execute `EX_DRM_TAPE_RESTORE` on the backup server.

   If the `-mount_pt` option is specified during backup, specify the same mount point as specified during backup. If the `-mount_pt` option is omitted during backup, it can also be omitted during restoration.

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002 -mount_pt E:
```

3. Export the backup information to temporary files.

   To restore data from the secondary volume to the primary volume, the backup information about the restore operation from tape to the secondary volume needs to be copied to the database server. To export the backup information to temporary files, execute `EX_DRM_DB_EXPORT`. These temporary files are stored in the temporary folder for extended commands.

```
BKServer > EX_DRM_DB_EXPORT Operation_A
```

4. Back up the metafile created at the restoration to the temporary folder.

   Back up the metafile created at the restoration to the temporary folder for extended commands that is on the backup server, so that the metafile and temporary files can be transmitted collectively to the database server. To back up the metafile, execute `EX_DRM_SQLFILE_PACK` on the backup server. Note, however, that you do not need to execute this command if you did not register the VDI metafile storage directory when executing the `drmsqlinit` command.

```
BKServer > EX_DRM_SQLFILE_PACK Operation_A
```

5. On the database server, receive the metafile and temporary files.

   To receive the metafile and temporary files collectively from the backup server, execute `EX_DRM_FTP_GET` on the database server. In this example, the user ID to log on to the FTP server is `admin` and the login password is `password`. The metafile and temporary files are stored in the temporary folder for extended commands, on the database server.

```
DBServer > EX_DRM_FTP_GET Operation_A -server BKServer -user admin -password password
```

6. Import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server.

   To import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server, execute `EX_DRM_DB_IMPORT` on the database server.

```
DBServer >EX_DRM_DB_IMPORT Operation_A
```

7. On the database server, deploy the metafile that was transmitted from the backup server.

   To deploy the metafile transmitted from the backup server, execute `EX_DRM_SQLFILE_EXTRACT` on the database server.

```
DBServer > EX_DRM_SQLFILE_EXTRACT Operation_A
```

8. Restore data in the secondary volume to the primary volume.

   To restore the primary volume by resynchronizing the secondary volume with the primary volume, execute `EX_DRM_SQL_RESTORE` on the database server.

```
DBServer > EX_DRM_SQL_RESTORE Operation_A -resync
```

   When `EX_DRM_SQL_RESTORE` is executed with the `-undo` option specified, restoration is performed while the database is in the standby status, in which the database is read-only. If the `-undo` option is omitted, after restoration, the status of the database changes to loading, and the database cannot be viewed.

   Also, be sure not to connect to SQL Server while executing `EX_DRM_SQL_RESTORE`.

9. Recover the SQL Server database.

   To recover the SQL Server database, execute the `drmsqlrecover` or `drmsqlrecovertool` command.

The following section discusses an example of the drmsqlrecover command and drmsqlrecovertool dialog box. For details on this dialog box, For details on the drmsqlrecover command or the drmsqlrecovertool dialog box, see the *HiCommand Protection Manager Command Reference* manual.

7.3.4.2 Recovering a SQL Server Database with the drmsqlrecover Command

The drmsqlrecover command is useful for recovering particular databases and recovering multiple volumes at once.

Specify an instance name to execute the drmsqlrecover command. The following command illustrates execution of the drmsqlrecover command to recover the SQL Server database:

```
DBServer > drmsqlrecover default
KAVX0001-I The drmsqlrecover command will now start.
KAVX0002-I The drmsqlrecover command will now end.
DBServer >
```

When you execute the drmsqlrecover command with an instance name specified, the database becomes accessible, and the recovery is complete.

When you execute the drmsqlrecover command with the -undo option specified, the database is recovered in the standby status (read-only and usable). When you execute with the -loading option specified, the database becomes loading status and not accessible. Note that the -undo option cannot be used unless the UNDO file storage directory has been set using the drmsqlinit command.

7.3.4.3 Recovering a SQL Server Database Using the drmsqlrecovertool Dialog Box

The drmsqlrecovertool command is useful for performing recovery and checking the status of a database at the same time.

The following command illustrates execution of the drmsqlrecovertool command to recover the SQL Server database:

1. Specify an instance name to execute the drmsqlrecovertool command.

   ```
   DBServer > drmsqlrecovertool default
   ```

   When you execute the command, the drmsqlrecovertool dialog box is displayed as shown in Figure 7.18.

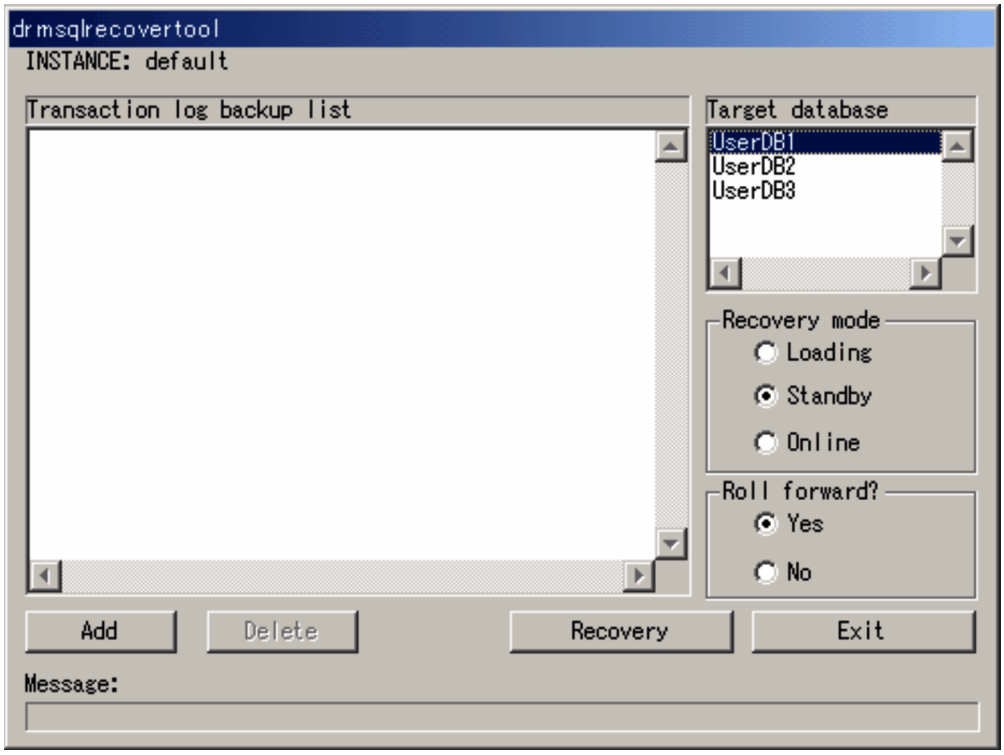


Figure 7.18 The `drmsqlrecovertool` Dialog Box

2. Select a recovery destination database in **Target database**.
3. Check the status of the database after recovery.

   For **Recovery mode**, select the status of the recovered database.

   Standby: Select this to perform recovery in the standby status (read-only and usable). Note that you cannot select **Standby** unless the UNDO file storage directory is set using the `drmsqlinit` command.

   Loading: Select this to perform recovery in the loading status.

   Online: Select this to make the database fully accessible.
4. Select whether to perform roll-forward.

   To apply transaction logs (a roll-forward), select **Yes** for **Roll forward?**. If no transaction log is displayed in **Transaction log backup list**, **Roll forward?** can be set to either **Yes** or **No**.
5. Click the **Recovery** button.

   If **Online** is selected for **Recovery mode**, `*` is appended to the file name displayed in **Target database**, and recovery is complete.

7.4 Backing Up a SQL Server Database Using a User Script

By using a backup command that specifies a user script, you can use a single operation to back up a SQL Server database to tape from the primary volume via the secondary volume.

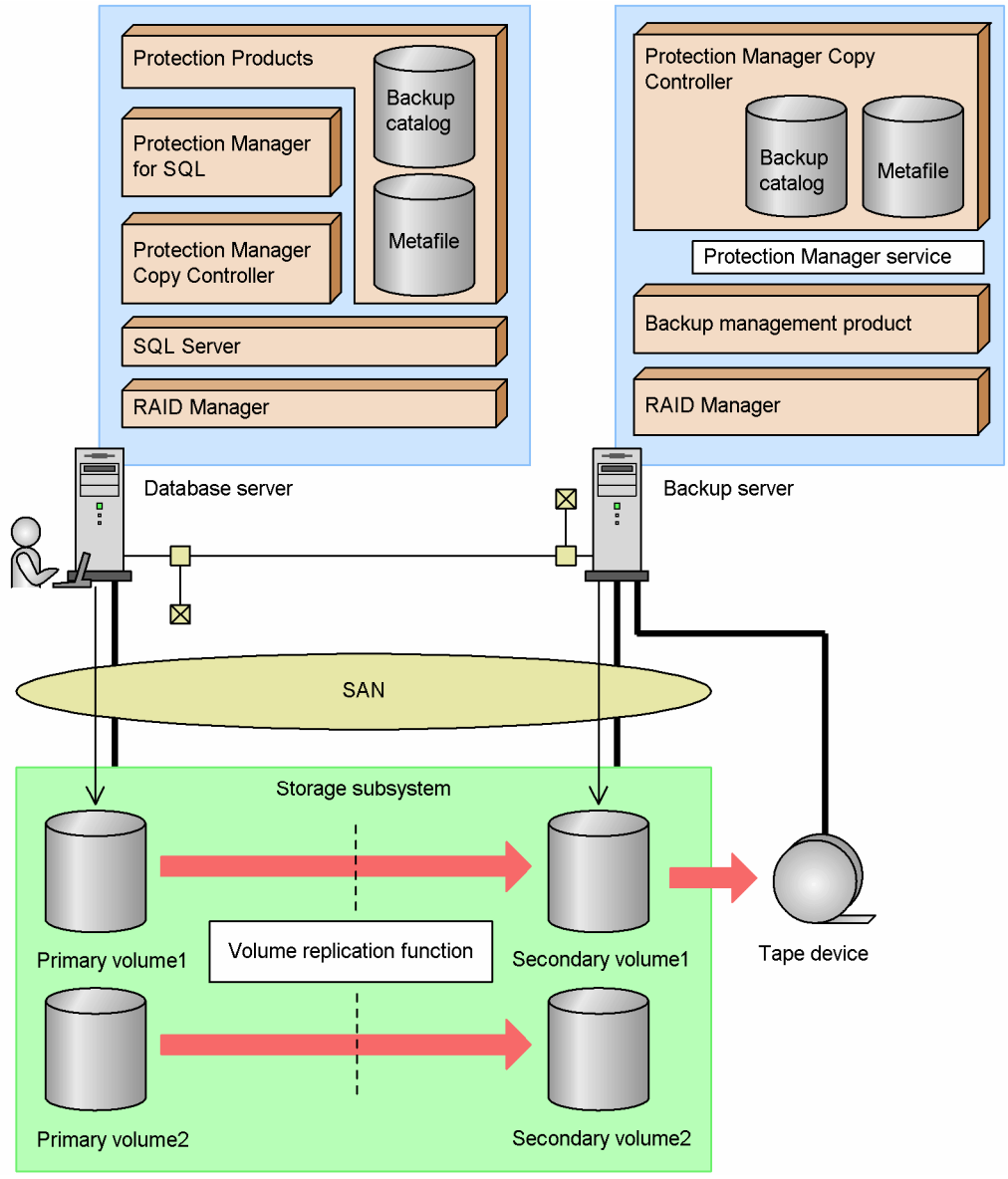


Figure 7.19 System Configuration for Backup of a SQL Server Database to Tape

7.4.1 User Script Backup Processing Flow

In this example, the processing flow of the `drmsqlbackup` command that specifies a user script is as follows:

- The cache of the secondary volume is cleared (this is a step in the user pre-processing section).

- An online backup of SQL Server's `DEFAULT` instance is performed.
- Once the backup from the primary volume to the secondary volume has been completed, the backup catalog is transferred to the backup server.
- On the backup server, the secondary volume is mounted on the `G` drive.
- The secondary volume is backed up to tape by using `NTBACKUP`, and then is unmounted (this is a step in the user post-processing section).

The `drmsqlbackup` command terminates without waiting for the tape backup operation to terminate.

Note: By not specifying the VDI metafile storage location in `drmsqlinit`, the VDI metafile is written to the primary volume and can be backed up at the same time as the data files.

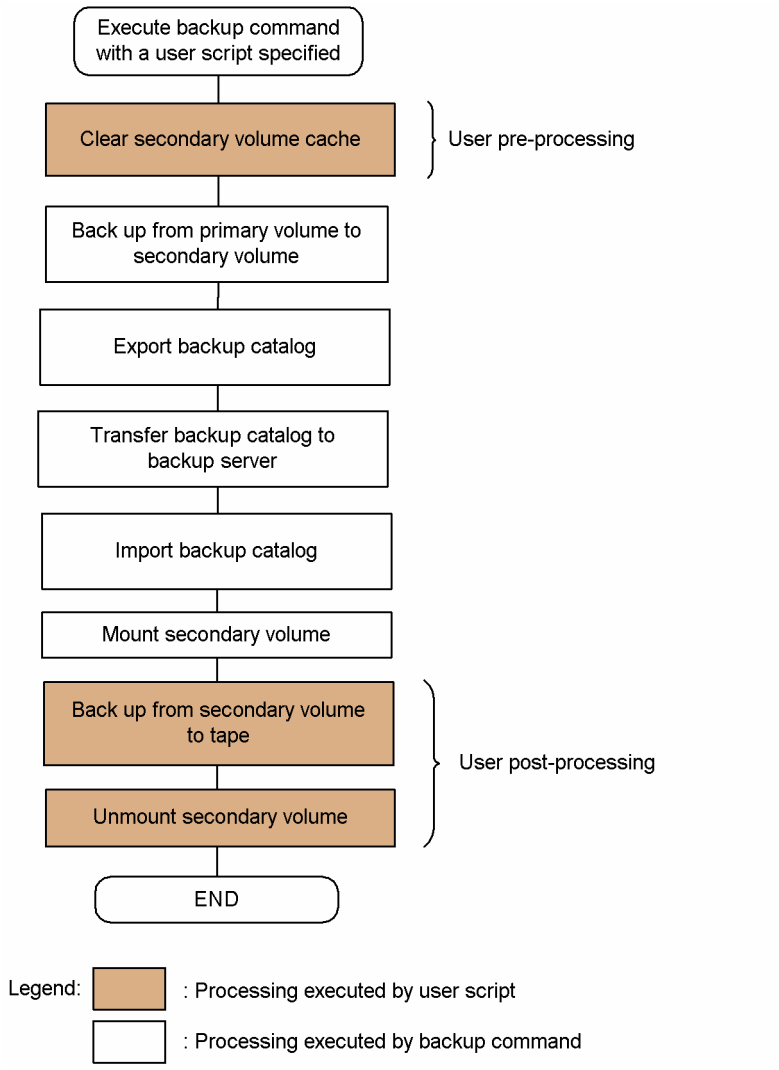


Figure 7.20 Processing Flow

7.4.2 User Script Example

Table 7.5 shows the user script that is created for the example.

Table 7.5 User Script Creation Example

Script text	Explanation
`LOCAL_BACKUP=YES ...(1)`	(1) `YES` is required.
`#Pre-processing section`	
`[PRE_PROC] ...(2)`	(2) Start of the user pre-processing section
`#Clearing of the secondary volume cache (mounting and unmounting)` `[CMD] ...(3)`	(3) Mounting of the first volume
`CMDLINE="C:\program files\hitachi\drm\bin\drmmount.exe" -copy_group vg01,vol01` `LOCATION=REMOTE ...(4)`	(4) Execute on a remote server.
`[CMD] ...(5)`	(5) Mounting of the second volume
`CMDLINE="C:\program files\hitachi\drm\bin\drmmount.exe" -copy_group vg01,vol02` `LOCATION=REMOTE`	
`[CMD] ...(6)`	(6) Unmounting of the first volume
`CMDLINE="C:\program files\hitachi\drm\bin\drmumount.exe" -copy_group vg01,vol01` `LOCATION=REMOTE`	
`[CMD] ...(7)`	(7) Unmounting of the second volume
`CMDLINE="C:\program files\hitachi\drm\bin\drmumount.exe" -copy_group vg01,vol02` `LOCATION=REMOTE`	
`#Post-processing section`	
`[POST_PROC] ...(8)`	(8) Start of the user post-processing section
`#Tape backup of the secondary volume`	
`[CMD]`	
`CMDLINE=C:\tmp\tapebackup.bat`	
`TIMEOUT=NOWAIT ...(9)`	(9) Execute the next command without waiting for the command to terminate.
`END_CODE=TERMINATE_NZ ...(10)`	(10) Treat a command return value that is a value other than 0 as an error.
`LOCATION=REMOTE`	
`PARENT_STAT=NORMAL ...(11)`	(11) Execute only when the backup command is normal.

The following is an example of backing up the secondary volume to tape using `tapebackup.bat`:

```
rem Use NTBACKUP to execute the job Job1,and perform a copy backup of G:\ to tape Tape1
rem Set the backup comment passed to the DRMENV_COMMENT environment variable as the backup job explanation
"C:\Windows\system32\ntbackup.exe" backup G:\ /j "Job1" /a /t "Tape1" /D "%DRMENV_COMMENT%" /m copy
IF NOT "%errorlevel%"=="0" GOTO ERROR

rem After the tape backup, specify the backup ID imported to the backup server, and unmount the secondary
volume
"C:\Program Files\Hitachi\DRM\bin\drmumount.exe" %DRMENV_R_BACKUPID%
IF NOT "%errorlevel%"=="0" GOTO ERROR
exit 0
:ERROR
exit 1
```

Note: A line that begins with the characters `rem` is a comment line.

7.4.3 Backup Execution Example

An example of using a user script is shown below. If you specified `LOCATION=REMOTE` in the user script, you must specify the `-s` option on the command line.

```
PROMPT>drmsqlbackup DEFAULT -script C:\tmp\script.txt -s BKHOST -auto_import
-auto_mount G: -comment TEST1
```

7.5 Operation Using SQL Server Transaction Logs

Backing up a SQL Server database entails either backing up all databases, or backing up transaction log files.

When formulating a plan for backing up a SQL Server database, if the database is fairly large so that the time required to perform backup is significant, or if the database is updated regularly, we recommend a combination of database backup and transaction log backup.

There are several precautions to keep in mind when backing up transaction logs. For details, see sections 7.1.7, 7.1.8, and 7.1.9.

To recover the database from the last backup data, or backups prior to the last backup because of a problem in the last backup data, see section 7.5.8.

7.5.1 Configuring to Use the SQL Server Transaction Log

The examples in this section assume the system configuration shown in Figure 7.21.

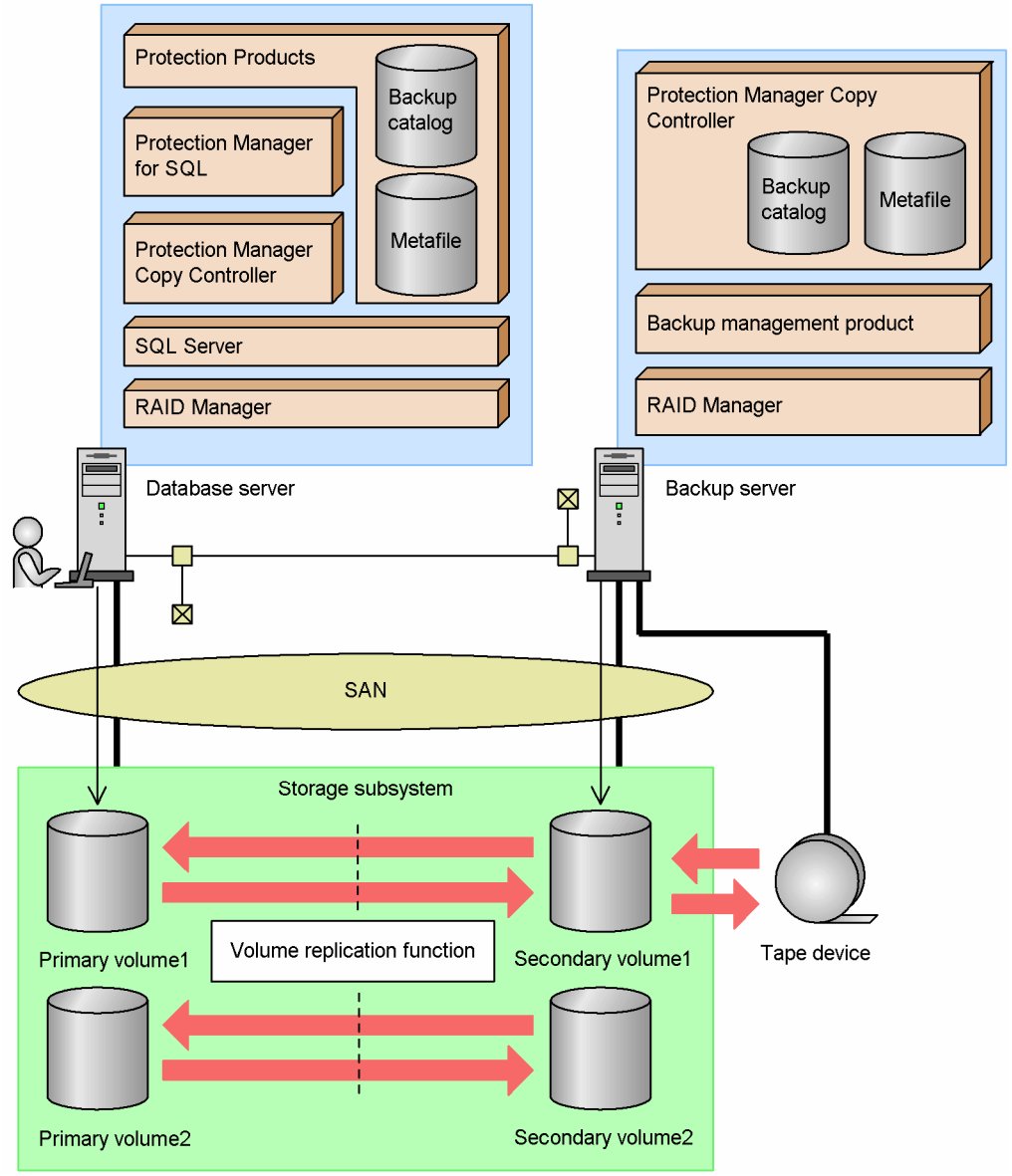


Figure 7.21 System Configuration for Operations Using the SQL Server Transaction Log

Protection Manager allows you to use a cluster configuration for database servers. If a failover cluster configuration is used for the database servers, the active server can be switched to the standby server when a failure occurs on the active server.

The examples in this section assume the following conditions:

- The primary volume is NTFS-formatted.
- The primary volume and the secondary volume are pair defined in the database server and the backup server.
- The backup management product is installed on the backup server.
- The parameters of the backup management product have been registered by using the `drmtapeinit` command.
- The FTP service on the backup server is started, and the system is set up so a user who has logged onto the OS of the database server can log onto the FTP server and transfer files to the FTP server. The FTP user ID is `admin`, and the FTP user password is `password`.
- Data for which an online back up was performed from the primary volume to the secondary volume was backed up to tape.
- The instance `DEFAULT` exists on the database server (server name: `DBServer`), and the service is started.
- On the database server and the backup server, the temporary directory for extended commands and the VDI metafile storage directory are created.
- The secondary volume is not always mounted. It is mounted on the E drive (drive character `E:`) only while it is operating.
- The primary volume configured as a pair with the secondary volume to be backed up will not be backed up until the secondary volume is backed up onto tape.
- The `cscript //H:Cscript` command is executed from the command prompt and the host parameters are updated.

7.5.2 Example of Operation Using a Transaction Log Backup

This section explains an example of operation using a combination of database backup and transaction log backup. In Figure 7.22, database data is deleted accidentally after transaction log backup, then the transaction log is applied, and the database is restored to the same state at “point C”. The operation ID is `Operation_A`.

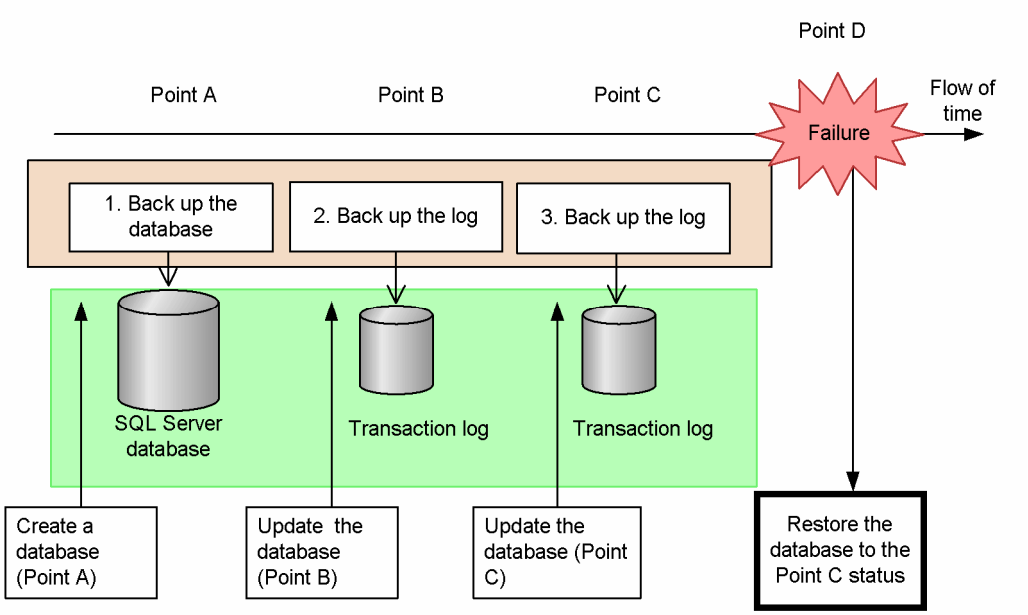


Figure 7.22 Operation Using Transaction Log Backups

This example is based on backup being obtained as follows:

1. Back up the SQL Server database at point A.

   For details about how the database is backed up at point A, see section 7.5.3.

2. Back up the SQL Server database transaction log at point B and point C.

   For details about how transaction logs are backed up, see section 7.5.4.

Procedures for restoration and recovery are as follows after failure occurs at point D.

1. Restore the data in the SQL Server database backed up at point A. The database is recovered to point A.

   For details about how to restore the SQL Server database backed up at point A, see section 7.5.5. For details about the order of applying transaction logs, see section 7.1.9.

2. Perform recovery by applying the transaction logs backed up at point B and point C.

   When you perform recovery by applying the transaction logs backed up at point B (point C), the database reverts to that at point B (point C).

   For details about how to perform recovery by using the `drmsqlrecovertool` dialog box and applying the transaction logs, see section 7.5.6.1.

   When you apply multiple transaction logs for the `drmsqlrecover` command, you can apply multiple transaction logs at one time by using the transaction log batch file. For details about how to perform recovery using a transaction log list file, see section 7.5.6.2.

7.5.3 Backing up a SQL Server Database

Assuming that the database is in the state as that at “point A”, obtain a backup of the SQL Server database.

7.5.3.1 Resynchronizing Copy Groups

When usual operation is in the split state, the copy group is first resynchronized, after which the data is backed up.

When usual operation is in the pair state, it is unnecessary to resynchronize the copy group before backing up the data. After the data is backed up to tape, resynchronize the copy group to restore data to the initial state.

To resynchronize the copy group:

1. For the copy pair configuration of multiple generations, check the copy group to be used next.

   To check the copy group used next, execute the `drmsqlcat` command on the backup server.

   If the copy group is not locked, the least recently backed up copy group is overwritten. (You can use the `drmcgctl` command to check the lock status of a copy group.)

2. Clear the cache of the secondary volume.

   Before performing backup, clear the system cache of the backup server.

   To clear the system cache, execute `EX_DRM_CACHE_PURGE` on the backup server at the local site to mount, and then unmount the secondary volume.

```
BKServer > EX_DRM_CACHE_PURGE Operation_A -cg_file C:\temp\CGLIST.txt
```

   In this example, the caches of the secondary volumes belonging to copy groups with names `VG01,vol11` and `VG01,vol12` are cleared. To specify multiple copy group names, you can create a copy-group list file ahead of time, and simply specify the file. In this example, copy groups `VG01,vol11` and `VG01,vol12` are defined in the copy-group list file `CGLIST.txt`.

3. Resynchronize the copy group.

   If you are using a Thunder 9200 Series, and ShadowImage with a Thunder 9500V Series or a TagmaStore AMS/WMS Series, execute `EX_DRM_RESYNC` to resynchronize the copy group. You can facilitate backup processing by resynchronizing the copy group in advance.

```
DBServer > EX_DRM_RESYNC Operation_A -cg_file C:\temp\CGLIST.txt -copy_size 7
```

7.5.3.2 Backing up a SQL Server Database

To back up a SQL Server database:

1. Back up the SQL Server database from the primary volume to the secondary volume.

```
DBServer > EX_DRM_SQL_BACKUP Operation_A
```

2. Export the backup information to temporary files.

```
DBServer > EX_DRM_DB_EXPORT Operation_A
```

3. Save the metafile created when backing up in the temporary folder.

```
DBServer > EX_DRM_SQLFILE_PACK Operation_A
```

4. Transfer the metafile and the temporary file to the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

5. Import the temporary file transferred from the database server in the backup catalog of the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

6. Expand the metafile transferred from the database server to the backup server.

```
BKServer > EX_DRM_SQLFILE_EXTRACT Operation_A
```

7. Back up the data of the secondary volume to tape.

   To back up, execute the `EX_DRM_TAPE_BACKUP` on the backup server. In this case, set the drive characters of the secondary volume `E:`.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   When backup is performed, information about related backup operations are registered in the backup catalog.

7.5.4 Backing up Transaction Logs

This section explains how to back up the transaction logs at point B and point C.

To back up the transaction logs, execute:

`EX_DRM_SQL_TLOG_BACKUP`.

```
DBServer > EX_DRM_SQL_TLOG_BACKUP Operation_A
```

7.5.5 Restoring SQL Server Databases

Restore the SQL Server database backed up at “point A”. To restore the SQL Server database, execute:

`EX_DRM_SQL_RESTORE`.

```
DBServer > EX_DRM_SQL_RESTORE Operation_A -resync -undo
```

When restoration is complete, make sure that the database has reverted to the status at “point A”.

If the `-undo` option was not specified for `EX_DRM_SQL_RESTORE`, the database is displayed as loading, and you cannot check its contents

If the `-undo` option was specified, the database becomes read-only, allowing you to check the contents.

Note that the `-undo` option cannot be used unless the UNDO file storage directory has been set using the `drmsqlinit` command.

7.5.6 Applying drmsqlrecovertool to Restore a SQL Server Database

This section explains how to perform recovery by applying the transaction logs backed up at “point B” and “point C”.

To apply the transaction logs, recover by using the `drmsqlrecovertool` command or the `drmsqlrecover` command.

7.5.6.1 Using drmsqlrecovertool to Recover a SQL Server Database

To perform recovery by applying the transaction logs:

1. Start the drmsqlrecovertool dialog box.

```
DBServer > drmsqlrecovertool DEFAULT
```

2. Select the database to be recovered.

3. Click the **ADD** button, and select the transaction log obtained at point B and point C.

   The transaction log backup file is named *database-name_time-of-transaction-log-backup_log-sequence-number*.bk.

4. Select **Standby** for **Recovery mode** and **Yes** for **Roll forward?**, and click the **Recovery** button.

   The transaction log obtained at point B and point C is applied, and the database is restored to the status at point C.

   At this point, the status of the database after recovery changes based on the selection for **Recovery mode**. Select **Standby** to perform recovery with the standby status (read-only and usable), **Loading** to perform recovery with the loading status, or **Online** to perform recovery with full access to the database.

   If **Online** was selected, allowing full access to the database, any existing transaction logs can no longer be applied. To apply multiple transaction logs in several separate batches, select either **Loading** or **Standby** for **Recovery mode**. Note that **Standby** cannot be selected unless the UNDO file storage directory has been set using the `drmsqlinit` command.

   After recovering databases in the standby status or loading status, open the drmsqlrecovertool dialog box again, select **Online** for **Recovery mode**, and click the **Recovery** button to allow full access.

7.5.6.2 Using the drmsqlrecover Command to Recover a SQL Server Database

When using the `drmsqlrecover` command to recover multiple databases, you can use a transaction log list file to apply multiple transaction logs at once.

To perform recovery using a transaction log list file:

1. Create a list file for backed up transaction logs.

```
DBServer > drmsqllogbackup DEFAULT -v > c:\temp\SQLTXLOG.txt
```

   A transaction log list file named `SQLTXLOG.txt` is created in the `c:\temp` directory. This file contains the contents of the backed up transaction logs.

   The following shows an example of a transaction log list file.

```
# ORIGINAL-ID:0000000080   BACKUP-ID:0000000080
[DB03]
l:\mssql\log\DB03_20040811115351_0001.bk
l:\mssql\log\DB03_20040811115431_0002.bk
# ORIGINAL-ID:0000000080   BACKUP-ID:0000000080
[DB02]
l:\mssql\log\DB02_20040811115351_0001.bk
l:\mssql\log\DB02_20040811115431_0002.bk
# ORIGINAL-ID:0000000080   BACKUP-ID:0000000080
[DB01]
l:\mssql\log\DB01_20040811115351_0001.bk
l:\mssql\log\DB01_20040811115431_0002.bk
```

2. Perform recovery by applying the transaction log batch.

   To perform recovery by applying a transaction log list file, execute the `drmsqlrecover` command with the `-transact_log_list` option specified.

```
DBServer > drmsqlrecover DEFAULT -transact_log_list c:\temp\SQLTXLOG.txt
```

7.5.7 Example of Operation Using Transaction Log Backup

If a failure occurs, you can restore the database up to the point of the last backup by using the data and transaction logs backed up during regular operation before the failure occurred. However, when you use only backups under normal operation, since operations performed after the last backup is truncated, such operations cannot be restored.

To restore a database to its state immediately before failure, you need to obtain the transaction logs from the last backup until the point immediately before failure. This section explains an example in which transaction logs are obtained without truncation, and the database is restored to the state immediately before failure (Figure 7.23). The operation ID is `Operation_A`.

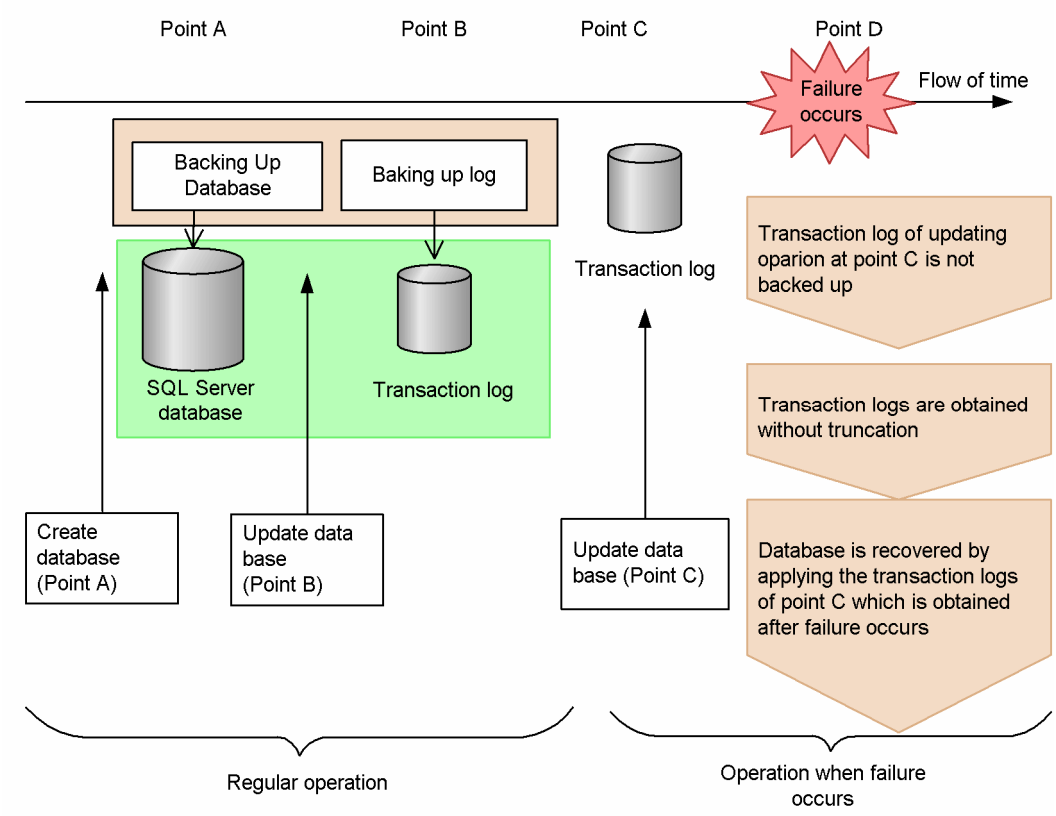


Figure 7.23 Operation Using SQL Server Transaction Log Backup (Restoring a Database)

To back up a SQL Server database and transaction logs during regular operation:

1. Back up the SQL Server database at point A.

   For details on how to back up the database at point A, see section 7.5.3.

2. Back up the SQL Server database transaction logs at point B.

   For details on how to back up the transaction logs, see section 7.5.4.

3. Backup is performed without truncating the transaction logs after failure occurs at point D.

   To obtain the transaction logs after failure, obtain transaction logs without truncating those subsequent to point C. To obtain transaction logs without truncating, execute the `EX_DRM_SQL_TLOG_BACKUP` with the `-no_truncate` option specified.

   ```
   DBServer > EX_DRM_SQL_TLOG_BACKUP Operation_A -no_truncate
   ```

   When the `-no_truncate` option is specified, transaction logs are obtained from after transaction log backup acquisition at point B, until failure occurred.

To perform restoration or recovery after obtaining transaction logs:

1. Restore the data in the SQL Server database backed up at point A.

   The database is recovered to point A. For details on how to restore the SQL Server database backed up at point A, see section 7.5.5.

2. Perform recovery by applying the transaction logs backed up at point B.

   When applying the transaction logs backed up at point B performs recovery, the database is reverted to the state at point B.

   For details on how to perform recovery the `drmsqlrecovertool` dialog by applying the transaction logs, see section 7.5.6.1.

   When you apply multiple transaction logs for the `drmsqlrecover` command, you can apply multiple transaction logs by using the transaction logs batch. For details about how to perform recovery using a transaction log list file, see section 7.5.6.2.

3. Perform recovery by applying transaction logs obtained with the `-no_truncate` option specified.

   When applying performs recovery transaction logs obtained by specifying the `-no_truncate` option, transactions are applied from transaction log backup acquisition at point B until the time of failure, and the database is reverted to the state at point C.

   For details about how to perform recovery by using the drmsqlrecovertool dialog box to apply transaction logs, see section 7.5.6.1.

   When you apply multiple transaction logs for the `drmsqlrecover` command, you can apply multiple transaction logs by using the transaction logs batch. For details about how to perform recovery using a transaction log list file, see section 7.5.6.2.

7.5.8 Applying Transaction Logs to Recover Databases from Backup at a Particular Time

When you apply transaction logs to recover the database from backup at a particular time, the transaction logs must be applied in the order of backup. This section describes the order of applying transaction logs when using the following procedures:

- Recovering the SQL Server database from the last backup data
- Recovering the SQL Server database from backup data prior to the last backup data

7.5.8.1 Recovering the SQL Server Database From the Last Backup Data

When you recover the SQL Server database from the last backup data, restore a previous backup from the primary volume to the secondary volume and then apply transaction logs.

Figure 7.24 shows an example of applying transaction log backups after restoring a previous backup from the primary volume to the secondary volume.

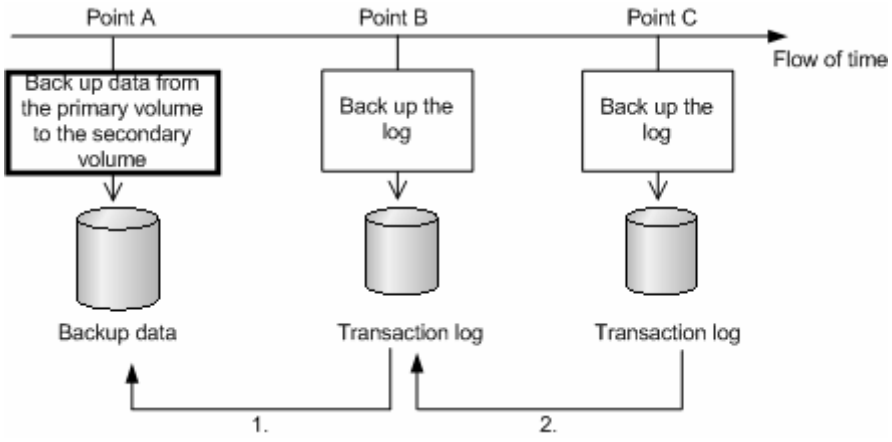


Figure 7.24 Order of Applying Transaction Log Backups (1)

Transaction log backups at point B can be applied after the previous volume backup (point A) is restored.

Transaction log backups at point C can be applied after the previous transaction log backup (point B) is applied.

7.5.8.2 Recovering the SQL Server Database From Backup Data Prior to the Last Backup Data

You can recover the SQL Server database from backup data prior to the last backup datawhen there is a problem in the latest backup data. For example, when you recover the SQL Server database from the second-last backup data, apply the transaction logs of the second-last backup from the primary volume to the secondary volume and then apply transaction log backup.

Figure 7.25 shows an example of applying a transaction log backup after the last transaction log obtained for the second-to-last backup from the primary volume to the secondary volume is applied. By applying the backups as shown in the example, even if a problem exists with the latest backup data, restoration can be performed to the most recent state, using the second-to-last backup data.

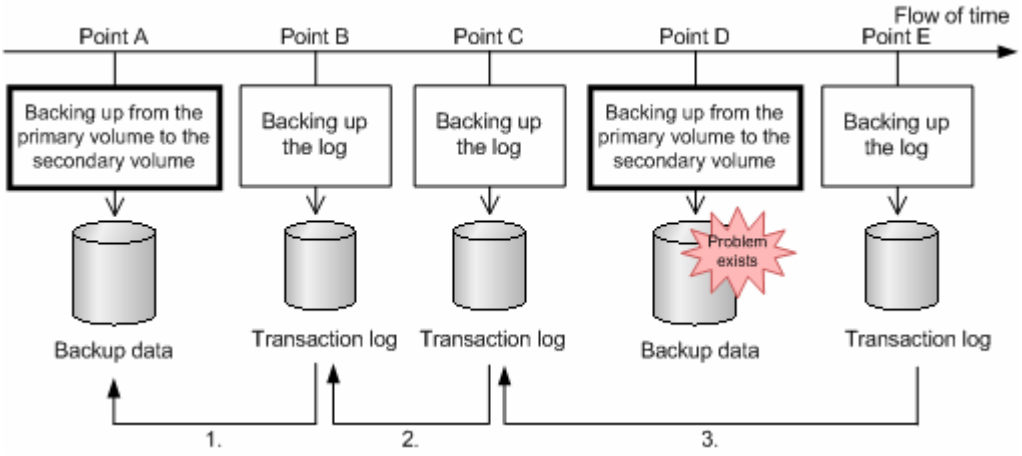


Figure 7.25 Order of Applying Transaction Log Backups (2)

1. Transaction log backups at point B can be applied after the previous volume backup (point A) is restored.
2. Transaction log backups at point C can be applied after the previous transaction log backup (point B) is applied.
3. The transaction log backup at point E can be applied after the last transaction log obtained for the backup from the primary volume to the secondary volume is applied (point C).

7.6 Backing up and Restoring Transaction Log Backup Files

Before you can recover a SQL Server database, you need to back up the transaction logs.

Two methods are available to save transaction logs that have previously been backed up, to tape or other storage media:

- Backing up pertinent files directly using a backup management product
- Backing up the transaction logs to the primary volume, and then saving the contents of the primary volume to tape or other storage media. This method does not place a heavy load on the LAN.

The rest of this section explains the procedure that is used when transaction logs are to be backed up to the primary volume, to back up transaction logs to the primary volume, and then save the primary volume contents to tape or other storage media.

7.6.1 Process Flow for Backing Up Transaction Log Backup Files

The examples in this section assume the system configuration shown in Figure 7.26.

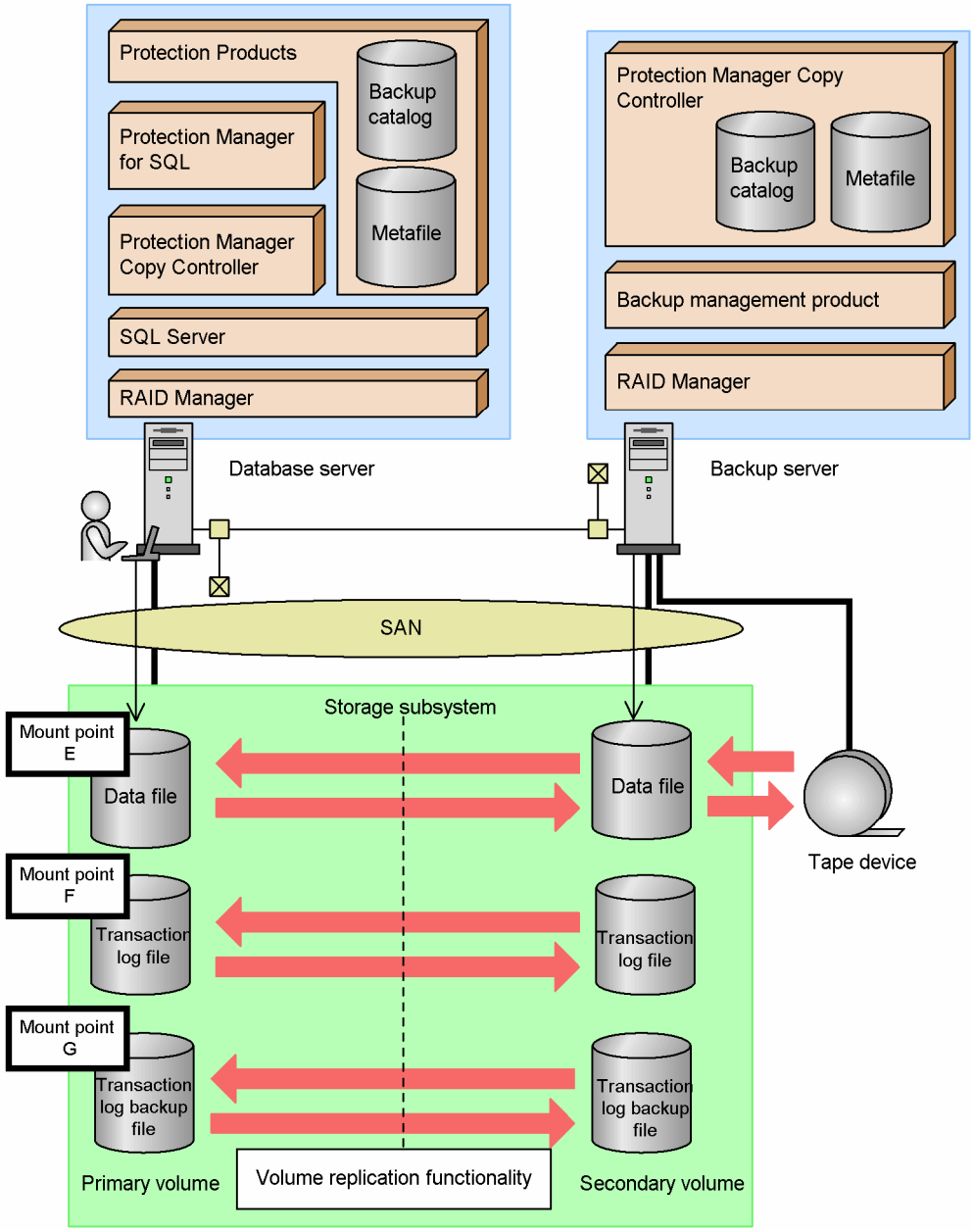


Figure 7.26 Backing up SQL Server Transaction Log Backup Files

The system prerequisites for this example are as follows:

- A pair is defined for each of the volumes on which the E, F, and G drives are mounted.
- SQL Server data files are placed on the E drive and SQL Server transaction log files are placed on the F drive.
- The `drmsqlinit` command has been used to place the transaction log backup file storage directory in `G:\Logbackup`.
- The operation definition file used to back up SQL Server data files is defined as `SQL1`. Database DB1 has been backed up.

  Contents of `_SQL1.dat`:

  ```
  BACKUP_OBJECT=MSSQL
  DB_SERVER_NAME=FILESERV1
  INSTANCE_NAME=DEFAULT
  TARGET_NAME=DB1
  FTP_HOME_DIR=C:\FTP_ROOT
  FTP_SUB_DIR=script
  SET_DRM_HOSTNAME=1
  ```

- The operation definition file used to back up the file system is named `FS1`. The backup target is the G drive that contains the transaction log backup file storage directory.

  Contents of `_FS1.dat`:

  ```
  BACKUP_OBJECT=FILESYSTEM
  DB_SERVER_NAME=FILESERV1
  INSTANCE_NAME=G:
  TARGET_NAME=G:\Logbackup
  FTP_HOME_DIR=C:\FTP_ROOT
  FTP_SUB_DIR=script
  SET_DRM_HOSTNAME=1
  ```

7.6.2 Backing up Transaction Logs

1. Back up the transaction logs.

   To back up transaction logs, execute `EX_DRM_SQL_TLOG_BACKUP` on the database server.

```
DBServer > EX_DRM_SQL_TLOG_BACKUP SQL1 -target DB1
DBServer >
```

2. Check the backup ID that identifies the start point for the transaction logs.

   To check the backup ID that identifies the start point for the transaction logs, execute the `drmsqllogbackup` command with the *`instance-name`* `-v BACKUP-ID` option specified.

```
DBServer > drmsqllogbackup SQL1 -v BACKUP-ID
ORIGINAL-ID   BACKUP_ID    DB
0000000002    0000000002   DB1
DBServer >
```

3. Check the contents of the transaction logs that have been backed up.

   To check the contents of the transaction logs that have been backed up, execute the `drmsqllogbackup` command with the *`instance-name`* `-target` *`database-name`* `-v` option specified.

```
DBServer > drmsqllogbackup SQL1 -target DB1 -v
# ORIGINAL-ID: 0000000002 BACKUP-ID: 0000000002
[DB1]
G:\LogBackup\DB1_20021106010100_0001.bk
G:\LogBackup\DB1_20021106010110_0002.bk
DBServer >
```

4. Back up the backed up transaction log file to the secondary volume.

   This step backs up the transaction log file online. To do this, execute `EX_DRM_FS_BACKUP`.

```
DBServer > EX_DRM_FS_BACKUP FS1
DBServer >
```

   When executes this command, places backup information with a new backup ID (the next unused ID) in the backup catalog on the database server. It is assumed here that this new backup ID is `0000000003`.

5. Check the backup execution results.

   To confirm that backup to the secondary volumes finished successfully, execute the `drmsqlcat` command.

```
DBServer> drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000003 BACKUP-MODE: ONLINE INSTANCE: G: ORIGINAL-ID:0000000003
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: FILESERV1
T FILE          FS          DG        DEVICE              COPY-GROUP
F -             G:          -         Harddisk1           FS001,DDrive
F -             -           -         -                   FS001,DDrive
```

6. Export the backup information to temporary files.

   To export the information pertaining to the backup operations to temporary files, execute `EX_DRM_DB_EXPORT`.

```
DBServer> EX_DRM_DB_EXPORT FS1
DBServer>
```

7. Transfer the exported temporary file from the database server to the backup server.

```
DBServer> EX_DRM_FTP_PUT FS1 -server BKServer -user admin -password password
DBServer>
```

8. Import the temporary file, which was transferred from the database server, into the backup catalog of the backup server.

```
DBServer> EX_DRM_DB_IMPORT FS1
DBServer>
```

9. Check the import results.

   To confirm that the import finished successfully, execute the `drmfscat` command on the backup server.

```
BKServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000001 BACKUP-MODE: ONLINE INSTANCE: G: ORIGINAL-ID:0000000003
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: FILESERV1
T FILE  FS  DG  DEVICE     COPY-GROUP
F -     G:  -   Harddisk1  FS001,DDrive
F -     -   -   -          FS001,DDrive
BKServer >
```

10. Back up the data of the secondary volume to tape.

    To back up the data of the secondary volume, execute `EX_DRM_TAPE_BACKUP` on the backup server. Specify the `G:` drive as the mount point.

```
BKServer > EX_DRM_TAPE_BACKUP FS1 -mount_pt G:
BKServer >
```

    When this backup operation is performed, information on this operation is entered into the backup catalog, with a new backup ID `0000000004`.

11. Check the backup to tape execution results.

    To confirm that backup to tape finished successfully, execute the `drmtapecat` command.

```
BKServer > drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000004  FILESYSTEM       2002/06/01 10:20:00  2002/09/01 10:20:00  MEDIA1
BKServer >
```

7.6.3 Restoring Transaction Logs

1. Check the backup ID of the backup data you want to use for restoration.

   To check the backup ID, execute the `drmtapecat` command on the backup server.

```
BKServer > drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000004  FILESYSTEM       2002/06/01 10:20:00  2002/09/01 10:20:00  MEDIA1
BKServer >
```

2. Restore data from its backup tape to the secondary volume.

   To restore data, execute `EX_DRM_TAPE_RESTORE` on the backup server.

```
BKServer > EX_DRM_TAPE_RESTORE FS1 -backup_id 0000000004 -mount_pt G:
BKServer >
```

   When restore is performed, information about the restore operation is registered in the backup catalog on the backup server with new backup ID `0000000005`.

3. Check that correct data is restored from the tape to the secondary volume.

   To check that backup information with the date of the backup from the primary volume to the secondary volume is restored on the backup server, execute the `drmfscat` command on the backup server.

```
BKServer > drmfscat G:
INSTANCE: G:
BACKUP-ID:0000000005 BACKUP-MODE: ONLINE INSTANCE: G: ORIGINAL-ID:0000000003
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: FILESERV1
T FILE  FS  DG  DEVICE     COPY-GROUP
F -     G:  -   Harddisk1  FS001,DDrive
F -     -   -   -          FS001,DDrive
BKServer >
```

4. Export the backup information to temporary files.

   To restore data from the secondary volume to the primary volume, the backup information about the restore operation from tape to the secondary volume needs to be copied to the database server. To export the backup information to temporary files in the temporary directory for extended commands, execute `EX_DRM_DB_EXPORT`.

```
BKServer > EX_DRM_DB_EXPORT FS1
BKServer >
```

5. On the database server, receive the temporary files.

   To receive the temporary files collectively from the backup server, execute `EX_DRM_FTP_GET` on the database server. In this example, the user ID to log on to the FTP server is `admin` and the login password is `password`. The temporary files are stored in the temporary directory for extended commands, on the database server.

```
DBServer > EX_DRM_FTP_GET FS1 -server BKServer -user admin -password password
DBServer >
```

6. Import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server.

   To import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server, execute `EX_DRM_DB_IMPORT` on the database server.

```
DBServer > EX_DRM_DB_IMPORT FS1
DBServer >
```

7. Restore data in the secondary volume to the primary volume.

   To restore the data and resynchronize the primary and secondary volumes, execute `EX_DRM_FS_RESTORE` on the database server.

```
DBServer > EX_DRM_FS_RESTORE FS1 -resync
DBServer >
```

7.7 Backing up and Restoring Multiple Generations by Locking a Copy Group

You can manage backups of multiple generations by using multiple copy groups for each backup purpose.

In Protection Manager, backup is usually performed to an automatically determined copy group, but you can also use the `drmcgctl` command to lock the contents of a particular copy group, and continue backup using other copy groups only.

The following kinds of operation become possible when generation management is performed by locking particular copy groups:

- After a monthly backup is obtained for a volume, the copy group is locked, and daily backups are obtained for the remaining two volumes.

  This allows you to keep the monthly backup until the copy group is unlocked.

- Backups obtained for a volume are kept, while the usual backups are obtained for the other two volumes.

7.7.1 Using Lock Functionality and Generation Management

Figure 7.27 shows an example of operation in which a particular copy group is locked.

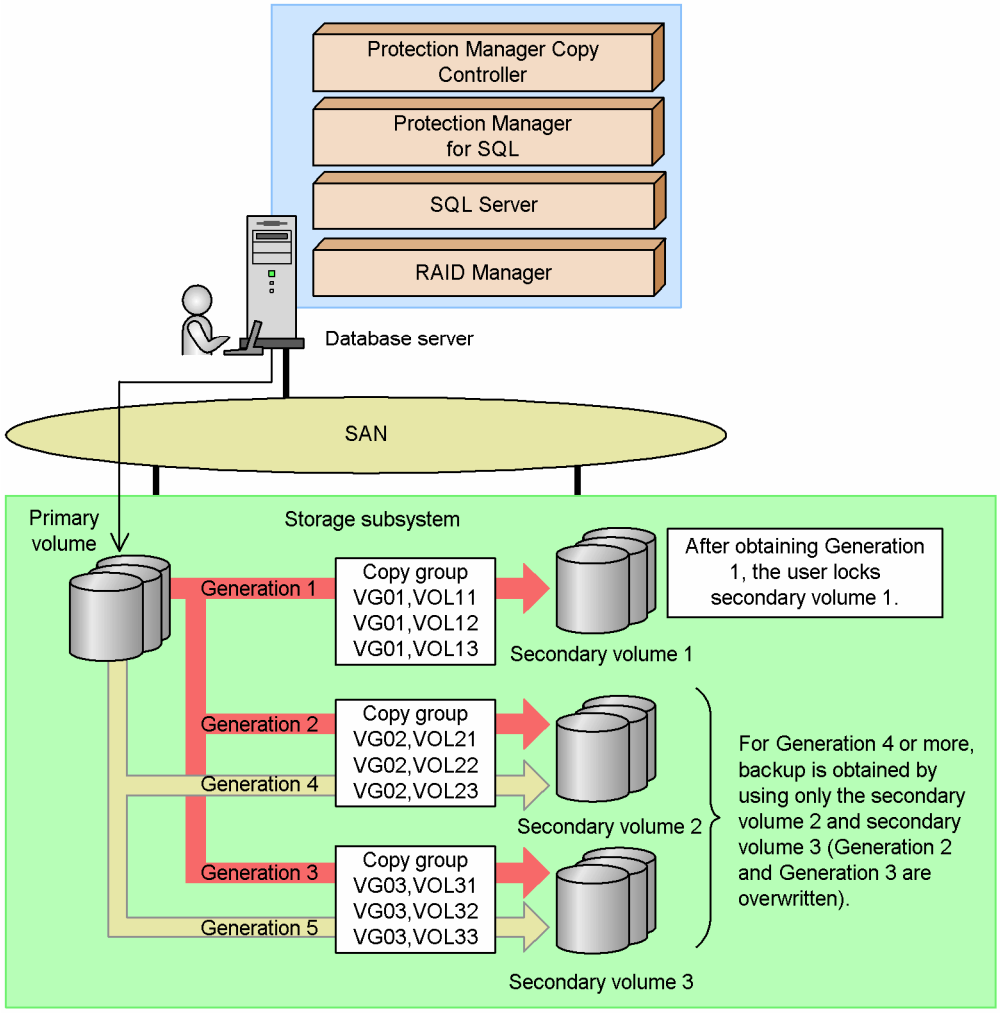


Figure 7.27 Example of Operation in Which a Particular Copy Group is Locked

1. Back up the primary volume to secondary volume 1 (generation 1), and then lock copy group 1 (generation 1).

   For details about the procedures from backing up generation 1 to locking copy group 1 (generation 1), see section 7.7.2.

2. Use secondary volume 2 and secondary volume 3 to perform backup operation.

   Use secondary volume 2 and secondary volume 3 to obtain backups for generations 2 and later.

3. Unlock the copy group (generation 1).

   When the copy group (generation 1) no longer needs to be locked, or to restore the contents of secondary volume 1 to the primary volume, unlock the copy group (generation 1).

For details about how to unlock the copy group (generation 1), see section 7.7.3.

7.7.2 Locking a Copy Group

This section explains how to lock a copy group after a generation 1 backup is obtained.

For the following, `Operation_A` is used as the operation ID.

To lock the copy group:

1. Checking the secondary volume status.

   Display the list of copy groups and make sure that none are locked.

   To check the lock status of a copy group, execute the `drmcgctl` command with no parameters. If `UNLOCKED` is displayed in the **LOCK STATUS** column, the copy group is not locked.

```
DBServer > drmcgctl
COPY GROUP LOCK STATUS BACKUP-ID
VG01,VOL11 UNLOCKED    0000000001
VG01,VOL12 UNLOCKED    0000000001
VG01,VOL13 UNLOCKED    0000000001
VG02,VOL21 UNLOCKED    0000000002
VG02,VOL22 UNLOCKED    0000000002
VG02,VOL23 UNLOCKED    0000000002
VG03,VOL31 UNLOCKED    0000000003
VG03,VOL32 UNLOCKED    0000000003
VG03,VOL33 UNLOCKED    0000000003
DBServer >
```

2. Obtain a backup (generation 1) from the primary volume to the secondary volume.

```
DBServer > EX_DRM_SQL_BACKUP Operation_A
```

   The primary volume is backed up to secondary volume 1.

3. Check that the data was backed up correctly.

   To check whether the backup was performed properly, execute the `drmsqlcat` command and `drmappcat` command on the database server.

```
DBServer > drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000004 BACKUP-MODE: ON-LINE INSTANCE: default
:
:
DBServer > drmappcat
BACKUP-ID  BACKUP-MODE BACKUP-OBJECT START-TIME          END-TIME
0000000004 ONLINE      MSSQL         2004/08/11 14:04:10 2004/08/11 14:04:22
0000000002 ONLINE      MSSQL         2004/08/05 14:13:44 2004/08/05 14:13:55
0000000003 ONLINE      MSSQL         2004/08/05 14:16:24 2004/08/05 14:16:35
DBServer >
```

   In this example, the results of the `drmsqlcat` command indicate that the most recent backup ID is `0000000004`. The results of the `drmappcat` command also indicate that the backup ID `0000000001`, displayed as the result of the `drmcgctl` command in the step 1, has been changed to `0000000004`.

4. Lock secondary volume 1.

   To keep the backup data (generation 1), lock the copy group (generation 1) by executing the `drmcgctl` command with the `-mode lock` option specified. You can specify the item to be locked by backup ID or copy group.

   At this point, execute the `drmcgctl` command, specifying `0000000004` as the backup ID.

```
DBServer > drmcgctl -backup_id 0000000004 -mode lock
KAVX0001-I The drmcgctl command will now start.
KAVX0302-I The copy group(s) associated with the specified backup ID were locked.
KAVX0002-I The drmcgctl command will now end.
```

   Note that the lock is only valid for the server on which it was performed. For a multiple server configuration, perform locking appropriate to the operation.

5. Checking the secondary volume status.

   Display the list of copy groups, and check the status of the locked copy group.

```
DBServer > drmcgctl
COPY GROUP LOCK STATUS BACKUP-ID
VG01,VOL11 LOCKED      0000000004
VG01,VOL12 LOCKED      0000000004
VG01,VOL13 LOCKED      0000000004
VG02,VOL21 UNLOCKED    0000000002
VG02,VOL22 UNLOCKED    0000000002
VG02,VOL23 UNLOCKED    0000000002
VG03,VOL31 UNLOCKED    0000000003
VG03,VOL32 UNLOCKED    0000000003
VG03,VOL33 UNLOCKED    0000000003
DBServer >
```

   Copy groups for which `LOCKED` is displayed in the **LOCK STATUS** column are locked.

   When the copy group (generation 1) is locked, all subsequent backups are obtained using secondary volume 2 and secondary volume 3.

7.7.3 Unlocking a Copy Group

When the copy group (generation 1) no longer needs to be locked, or when the contents of secondary volume 1 are restored to the primary volume, the copy group can be unlocked.

To unlock the copy group:

1. Unlock the copy group.

   To unlock the copy group, execute the `drmcgctl` command with the `-mode unlock` option specified. You can use the backup ID or copy group to specify the item to be unlocked.

   At this point, execute the `drmcgctl` command, specifying `0000000004` as the backup ID.

```
DBServer > drmcgctl -backup_id 0000000004 -mode unlock
KAVX0001-I The drmcgctl command will now start.
KAVX0303-I The copy group(s) associated with the specified backup ID were unlocked.
KAVX0002-I The drmcgctl command will now end.
```

2. Check the copy group status.

   Display the copy group list and check that the copy group has been unlocked.

```
DBServer > drmcgctl
COPY GROUP LOCK STATUS BACKUP-ID
VG01,VOL11 UNLOCKED    0000000004
VG01,VOL12 UNLOCKED    0000000004
VG01,VOL13 UNLOCKED    0000000004
VG02,VOL21 UNLOCKED    0000000007
VG02,VOL22 UNLOCKED    0000000007
VG02,VOL23 UNLOCKED    0000000007
VG03,VOL31 UNLOCKED    0000000008
VG03,VOL32 UNLOCKED    0000000008
VG03,VOL33 UNLOCKED    0000000008
DBServer >
```

   Unlocked copy groups have `UNLOCKED` displayed in the **LOCK STATUS** column.

   When the next backup is performed, the secondary volume for which data is stored for the backup ID `0000000004` will be backed up with backup ID `0000000009`.

7.8 Backing up and Restoring Between Remote Sites (Remote Copying)

This section explains the use of the volume replication function that is used to replicate volumes between TrueCopy, UR, or other subsystems. It includes an example of backing up a SQL Server database from a primary volume in the local site, to a secondary volume in the remote site, and restoring the SQL Server database from the secondary volume in the remote site to the primary volume in the local site.

The following conditions are assumed:

- A SQL Server database will be backed up to a remote site.
- The SQL Server database that is backed up to the remote site will be restored to the local site (if the local site is normally active).
- The SQL Server database that is backed up to the remote site will be restored to a remote site (if the local site has encountered a problem).
- The SQL Server database will be restored from the remote site to the local site (when the local site recovers from the problem).

The examples in this section assume the system configuration shown in Figure 7.28.

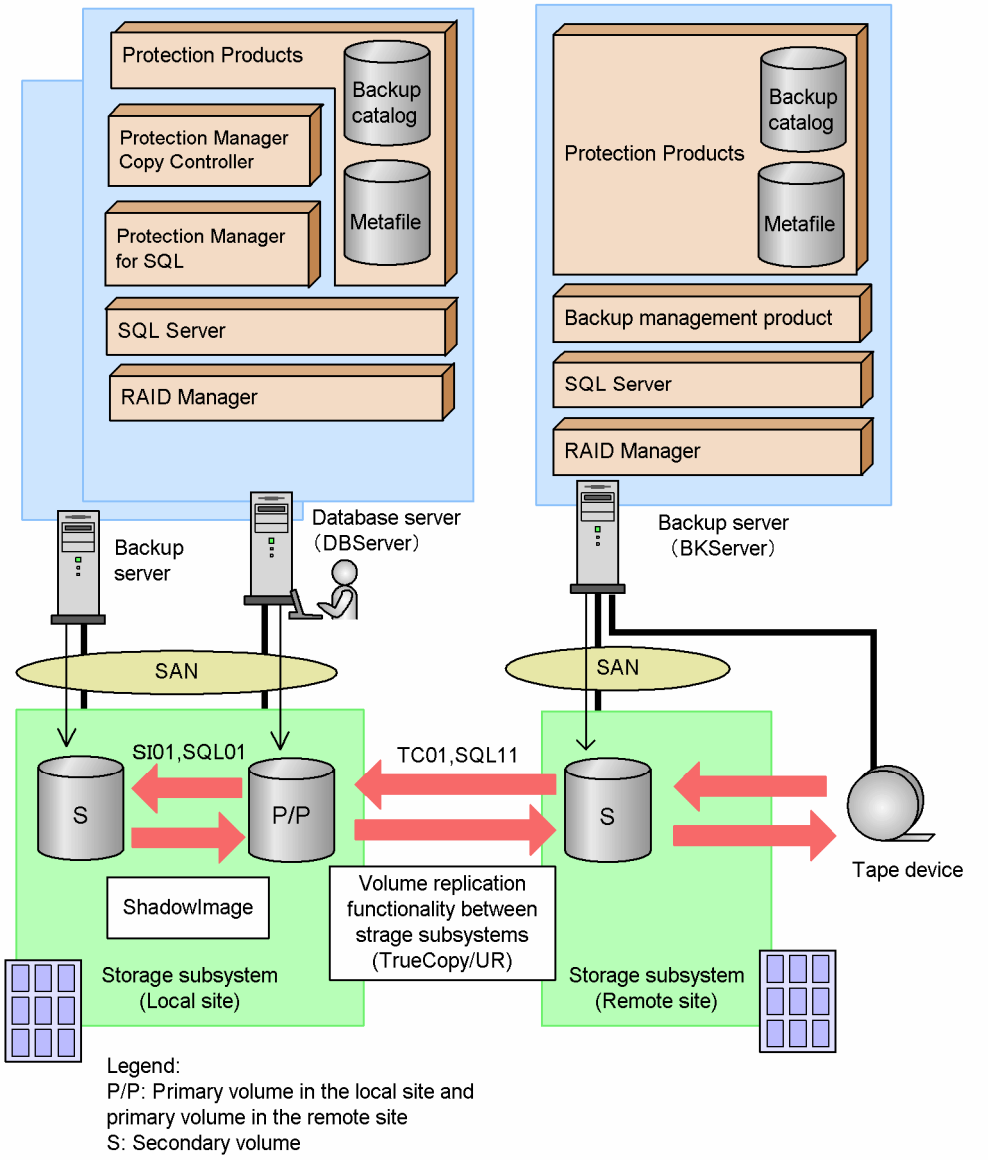


Figure 7.28 System Configuration Example Used When SQL Server Databases Are Backed up and Restored Between Remote Sites

The system prerequisites for this example are as follows:

- The system includes a local-site database server (server name: `DBServer`) and a remote-site backup server (server name: `BKServer`) to which a tape device is attached.
- FTP service is active on the remote-site backup server and is configured to ensure that any user who has logged in to the operating system running on each file server can log in to the FTP server and transfer files. The FTP user ID is `admin`, and the FTP user password is `password`.
- SQL Server is installed on the remote-site backup server (`BKServer`) to ensure that, when a problem occurs in the local-site storage subsystem, the backup server can be used as a database server.
- The local-site primary volume is mounted at the `G:` drive on `DBServer`.
- The copy group named `TC01,SQL11` contains a TrueCopy (usual operation is in the pair state).
- The SQL Server data files and transaction log files are included in the TrueCopy or UR volume (usual operation is in the pair state) named `TC01,SQL11`.
- The copy group named `SI01,SQL01` includes a ShadowImage pair.
- The system is configured to ensure that the generation identifier of the remote-site backup destination will be `remote_0`.
- The remote-site volume is not usually mounted. When necessary, it will be mounted at the `G:` drive on the backup server.
- The SQL Server data files and transaction log files are stored in the same directory structures on the local and remote sites.
- Use `Operation_A` as the operation ID.

7.8.1 Backing up a SQL Server Database to a Remote Site

The examples provided in the following sections explain the operation used to back up a file system to a remote site. Procedures at the local site and remote site are explained separately.

7.8.1.1 Operation at the Local Site

1. Check the volume to be backed up and its generation identifier.

   To check the information, execute the `drmsqldisplay` command on the database server.

```
DBServer > drmsqldisplay default -cf
INSTANCE: default
T DB       OBJECT    FILE              FS   DG   DEVICE      COPY-GROUP COPY-FUNC   GEN-NAME
D UserDB1   DATAFILE   G:\UserDB1_Data.MDF  G:   -    Harddisk1    SI01,SQL01 ShadowImage
local_0
- -         -          -                    -    -    -            SI01,SQL01 ShadowImage
local_0
T UserDB1   TRANSACT   G:\UserDB1_Log.LDF   G:   -    Harddisk1    SI01,SQL01 ShadowImage
local_0
- -         -          -                    -    -    -            SI01,SQL01 ShadowImage
local_0
D UserDB1  DATAFILE  G:\UserDB1_Data.MDF G:  -  Harddisk2  TC01,SQL11 TrueCopy   remote_0
- -        -         -                   -   -   -          TC01,SQL11 TrueCopy   remote_0
T UserDB1  TRANSACT  G:\UserDB1_Log.LDF  G:  -  Harddisk2  TC01,SQL11 TrueCopy   remote_0
- -        -         -                   -   -   -          TC01,SQL11 TrueCopy   remote_0
DBServer >
```

2. Back up the database to the remote-site secondary volume.

   To back up the database to the remote-site secondary volume, execute `EX_DRM_SQL_BACKUP`, in which `remote_0` is specified as the backup destination generation identifier, on the database server.

```
DBServer > EX_DRM_SQL_BACKUP Operation_A -rc remote_0
```

3. Check that backup was performed properly.

   To check the backup information, execute `drmsqlcat` on the database server.

```
DBServer > drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000001  BACKUP-MODE: ONLINE  INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 2004/08/05 10:00:00  END-TIME: 2004/08/05 10:03:00  HOSTNAME: DBServer
T DB       OBJECT    FILE                                  FS   DG   DEVICE      COPY-GROUP
M UserDB1   METAFILE   F:\MSSQL\meta\0000000131_0000000005.dmp -    -    -           -
D UserDB1   DATAFILE  G:\UserDB1_Data.MDF                   G:   -    Harddisk1   TC01,SQL11
- -         -         -                                     -    -    -           TC01,SQL11
T UserDB1   TRANSACT  G:\UserDB1_Log.LDF                    G:   -    Harddisk1   TC01,SQL11
DBServer >
```

4. Export the backup information to temporary files.

   To perform backup from the secondary volume to tape, you need to copy information about the backup operations from the primary volume to the secondary volume, to the backup server. To export the information pertaining to the backup operations to temporary files, execute `EX_DRM_DB_EXPORT`. The temporary files are stored in the temporary directory for extended commands, on the database server.

```
DBServer > EX_DRM_DB_EXPORT Operation_A
```

5. Transfer the temporary files to the backup server.

   Collectively transfer the temporary files from the database server to the backup server. The temporary files are stored in the temporary directory for extended commands, on the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

7.8.1.2 Operation at the Remote Site

1. Import the temporary files that have been transferred from the local-site database server, to the backup catalog on the backup server.

   To import the temporary files, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

2. Check that the import operation has been performed correctly.

   To confirm that backup information with the date of the backup from the primary volume to the secondary volume has been correctly imported to the backup server, execute the `drmsqlcat` command on the backup server.

```
BKServer > drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000001  BACKUP-MODE: ONLINE  INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 2004/08/05 10:00:00  END-TIME: 2004/08/05 10:03:00  HOSTNAME: DBServer
T DB       OBJECT    FILE                                    FS   DG   DEVICE      COPY-GROUP
M UserDB1   METAFILE   G:\MSSQL\meta\0000000001_0000000005.dmp -    -    -           -
D UserDB1   DATAFILE  G:\UserDB1_Data.MDF                     G:   -   Harddisk1   SI01,SQL01
- -        -         -                                       -    -    -           SI01,SQL01
T UserDB1   TRANSACT   G:\UserDB1_Log.LDF                    G:   -   Harddisk1   SI01,SQL01
BKServer >
```

3. Back up the data of the secondary volume to tape.

   To back up the data of the secondary volume, execute `EX_DRM_TAPE_BACKUP` on the backup server.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A
```

   When backup is performed in this example, information on this backup operation is entered in the backup catalog, with a new backup ID `0000000002`.

4. Confirm that data has been correctly backed up to tape.

   To confirm that backup information with the date of the backup from the primary volume to the secondary volume exists, execute `drmtapecat` on the backup server.

```
BKServer > drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000002  MSSQL            2004/08/05 10:00:00  2004/11/05 10:00:00  MEDIA1
BKServer >
```

5. Clear the cache of the secondary volume.

   After the tape backup operation finishes, clear the system cache on the backup server.

   To clear the system cache, on the backup server, execute `EX_DRM_CACHE_PURGE` to mount and then unmount the secondary volume.

```
BKServer > EX_DRM_CACHE_PURGE Operation_A
```

7.8.1.3 Operation at the Local Site

Resynchronize the copy group. To resynchronize the copy group, execute `EX_DRM_RESYNC` on the database server.

```
DBServer > EX_DRM_RESYNC Operation_A
```

7.8.2 Restoring a SQL Server Database to the Local Site

The example provided in the following sections explain the operation used to restore a SQL Server database to the local site if the local site is normally active and is free from any problem. Procedures at the local site and remote site are explained separately.

7.8.2.1 Operation at the Remote Site

1. Split the TrueCopy pair (usual operation is in the pair state).

```
BKServer > pairsplit -g TC01 -d SQL11 -rw
BKServer >
```

2. Check the backup ID of the backup data you want to use for restoration.

   To check the backup ID, execute the `drmtapecat` command on the backup server.

```
BKServer >drmtapecat
BACKUP-ID ORIGINAL-ID BACKUP-OBJECT INSTANCE SNAPSHOT TIME   EXPIRATION TIME  BACKUP-MEDIA
0000000002 0000000001 MSSQL         default  2004/08/05 10:00:00 2004/11/05 10:00:00  -
:
:
BKServer >
```

3. Restore data from its backup tape to the secondary volume.

   To restore data, execute `EX_DRM_TAPE_RESTORE` on the backup server.

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002
```

   When restore is performed, information about the restore operation is registered in the backup catalog on the backup server with new backup ID `0000000003`.

4. Check that correct data is restored from the tape to the secondary volume.

   To confirm that backup information with the date of the backup from the primary volume to the secondary volume has been restored to the backup server, execute the `drmsqlcat` command on the backup server.

```
BKServer > drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000003  BACKUP-MODE: ONLINE  INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 2004/08/05 10:00:00  END-TIME: 2004/08/05 10:03:00  HOSTNAME: DBServer
T DB       OBJECT    FILE                                  FS   DG   DEVICE      COPY-GROUP
M UserDB1   METAFILE   F:\MSSQL\meta\0000000131_0000000005.dmp -    -    -           -
D UserDB1   DATAFILE  G:\UserDB1_Data.MDF                   G:   -   Harddisk1   SI01,SQL01
- -        -         -                                     -    -    -           SI01,SQL01
T UserDB1   TRANSACT  G:\UserDB1_Log.LDF                   G:   -   Harddisk1   SI01,SQL01
BKServer >
```

5. Export the backup information to temporary files.

   Before you can restore a SQL Server database from the secondary volume to the primary volume, you need to copy backup information on the tape-to-secondary-volume restore operation to the file server.

```
BKServer > EX_DRM_DB_EXPORT Operation_A
```

7.8.2.2 Operation at the Local Site

1. In the local site database server, obtain the temporary files.

   To receive the temporary files collectively from the backup server, execute `EX_DRM_FTP_GET` on the database server. The temporary files are stored in the temporary directory for extended commands, on the database server.

```
DBServer > EX_DRM_FTP_GET Operation_A -server BKServer -user admin -password password
```

2. Import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server.

   To import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server, execute `EX_DRM_DB_IMPORT` on the database server.

```
DBServer >EX_DRM_DB_IMPORT Operation_A
```

3. Check that the import operation has been performed correctly.

   To confirm that backup information with the date of the backup from the primary volume to the secondary volume has been imported to the database server, execute the `drmsqlcat` command on the database server.

```
DBServer > drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000002  BACKUP-MODE: ONLINE  INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 2004/08/05 10:00:00  END-TIME: 2004/08/05 10:03:00  HOSTNAME: DBServer
T DB        OBJECT     FILE                    FS   DG   DEVICE       COPY-GROUP
M UserDB1   METAFILE   G:\META_0000000005.dmp -    -    -            -
D UserDB1   DATAFILE   G:\UserDB1_Data.MDF    G:   -    Harddisk1    SI01,SQL01
- -          -          -                       -    -    -            SI01,SQL01
T UserDB1   TRANSACT   G:\UserDB1_Log.LDF     G:   -    Harddisk1    SI01,SQL01
DBServer >
```

4. Restore data in the secondary volume to the primary volume.

   To restore the data and resynchronize the primary and secondary volumes, execute `EX_DRM_SQL_RESTORE` on the database server.

```
DBServer > EX_DRM_SQL_RESTORE Operation_A -resync
```

5. Recover the SQL Server database.

   To recover the SQL Server database, execute the `drmsqlrecover` or `drmsqlrecovertool` command on the database server at the local site.

6. Resynchronize the copy group.

   To resynchronize the copy group, execute `EX_DRM_RESYNC` on the database server.

```
DBServer > EX_DRM_RESYNC Operation_A
```

7.8.3 Restoring a SQL Server Database to a Remote Site

The example provided below explains the operation used to restore a SQL Server database in a remote site if the local site has encountered a problem.

7.8.3.1 Operation at the Remote Site

1. Split the TrueCopy pair (usual operation is in the pair state).

   Change the pair status from `PAIR` to `SMPL`.

```
BKServer > pairsplit -g TC01 -d SQL11 -S
BKServer >
```

2. Check the backup ID of the backup you want to use for restoration.

   To check the backup ID, execute the `drmtapecat` command on the backup server.

```
BKServer >drmtapecat
BACKUP-ID ORIGINAL-ID BACKUP-OBJECT INSTANCE SNAPSHOT TIME       EXPIRATION TIME
BACKUP-MEDIA
0000000002 0000000001 MSSQL         default  2004/08/05 10:00:00 2004/11/05 10:00:00 -
:
:
BKServer >
```

3. Restore data from its backup tape to the secondary volume.

   To restore data, execute `EX_DRM_TAPE_RESTORE` on the backup server.

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002
```

   When restore is performed, information about the restore operation is registered in the backup catalog on the backup server with new backup ID `0000000003`.

4. Check that correct data is restored from the tape to the secondary volume.

   To check that backup information with the date of the backup from the primary volume to the secondary volume is restored on the backup server, execute the `drmsqlcat` command on the backup server.

```
BKServer > drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000003  BACKUP-MODE: ONLINE  INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 2004/08/05 10:00:00  END-TIME: 2004/08/05 10:03:00  HOSTNAME: DBServer
T DB        OBJECT     FILE                   FS   DG   DEVICE       COPY-GROUP
M UserDB1   METAFILE   G:\META_0000000005.dmp -    -    -            -
D UserDB1   DATAFILE   G:\UserDB1_Data.MDF    G:   -    Harddisk1    SI01,SQL01
- -          -          -                      -    -    -            SI01,SQL01
T UserDB1   TRANSACT   G:\UserDB1_Log.LDF     G:   -    Harddisk1    SI01,SQL01
BKServer >
```

5. Mount the restored volume by using the disk management function.

   Mount is performed to the `G:` drive.

6. Restart SQL Server.

7. Create a backup ID file.

   To create a backup ID file, execute `EX_DRM_BACKUPID_SET` with a backup ID.

```
BKServer > EX_DRM_BACKUPID_SET Operation_A -backup_id 0000000003
```

8. Perform restore operation in the remote site.

   Since the operation here does not include copy group resynchronization, specify the `-no_resync` option.

```
BKServer > EX_DRM_SQL_RESTORE Operation_A -no_resync -nochk_host
```

9. Recover the SQL Server database.

   To recover the SQL Server database, execute the `drmsqlrecover` or `drmsqlrecovertool` command on the database server at the local site.

10. If a ShadowImage pair is deployed in the remote site, continue operation in the remote site by, for example, placing a backup into the pair volume.

7.8.4 Recovering Data from the Remote Site to the Local Site

Assuming that you have restored the SQL Server database onto the remote site and then the local site has recovered from the problem, the procedure below explains how to recover the restored database from the remote site onto the local site.

7.8.4.1 Operation at the Local Site

If a SQL server is running on the database server, stop the SQL Server. Use a disk management function to unmount the volume mounted by using the disk management function.

7.8.4.2 Operation at the Remote Site

1. Stop SQL Server.
2. Use the disk management function to unmount the volume that was mounted using the disk management function.

   The `G:` drive is unmounted.

3. To create a backup catalog, create a pair the primary volume of which should reside in the local site.

```
BKServer > paircreate -g TC01 -d SQL11 -vl -f never -c 15
BKServer >
```

7.8.4.3 Operation at the Local Site

1. Swap the primary volume and the secondary volume.

```
DBServer > pairresync –g TC01 –d SQL11 –swaps
DBServer >
```

2. Resynchronize the copy group.

   To resynchronize the copy group, execute `EX_DRM_RESYNC` on the database server.

```
DBServer > EX_DRM_RESYNC Operation_A
```

3. Restart SQL Server.

7.9 Backing up and Restoring in a Multi-Target Configuration

Using extended commands of Protection Manager for SQL, you can execute a simultaneous backup (a backup in a multi-target configuration) of data to a secondary volume at a local site in conjunction with a backup to a secondary volume at a remote site, in a combined system configuration of ShadowImage and TrueCopy.

Use basic commands, not extended commands, when backing up data only to a secondary volume in a remote site without executing local backup (when backing up in a cascade configuration). For examples of cascade configuration backup using basic commands, see section 7.12.

This section gives examples of processing data in a multi-target configuration using extended commands. The examples in this section assume the system configuration shown in Figure 7.29.

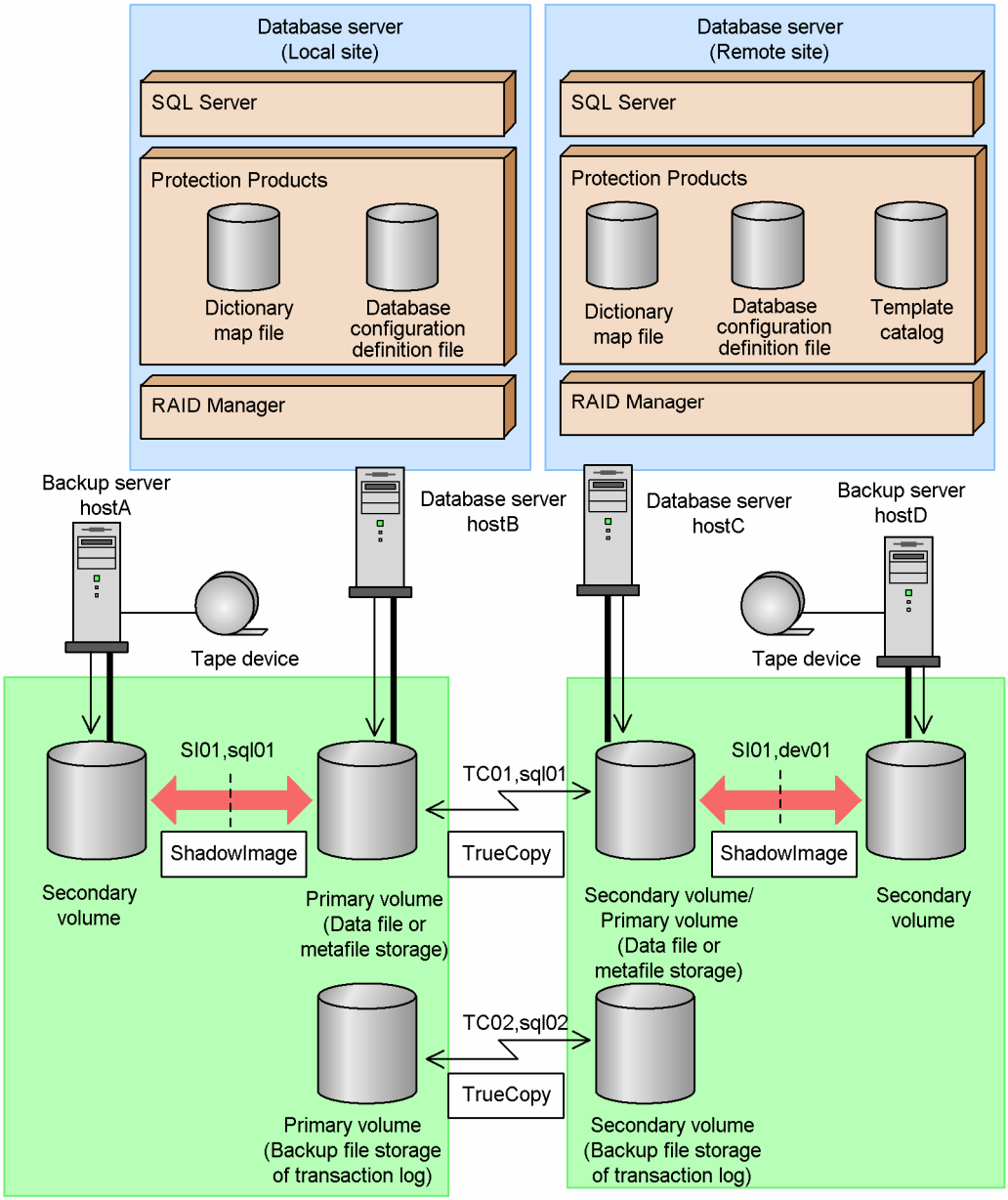


Figure 7.29 System Configuration for Backing up a SQL Server Database to Tape and Restoring from Tape in a Multi-Target Configuration

This example assumes the following conditions:

- The primary volume is NTFS-formatted.
- Both of the primary volumes in the local site and in the remote site are defined as a pair by TrueCopy. In each site, the primary volume and the secondary volume are also defined as a pair by ShadowImage.
- The system is prepared to operate in a multi-target configuration. For details about preparations for operations in a multi-target configuration, see section 7.12.1.
- Both the local site and the remote site have the same directory configuration for a SQL Server data files and a transaction log file.
- The SQL Server database for which backup operations are to be performed and database ID must have the same name both at the local and remote sites.
- The data files of the SQL Server database and the directory for storing the VDI metafile are in the same volume.
- The backup management product is installed on the backup server.
- The parameters of the backup management product have been registered by using the `drmtapeinit` command.
- The FTP service on the backup server is started, and the system is set up so a user who has logged onto the OS of the database server can log onto the FTP server and transfer files to the FTP server.
- Data for which an online back up was performed from the primary volume to the secondary volume was backed up to tape.
- Two SQL Server instances (`INSTANCE_1` and `INSTANCE_2`) exist, and services are running on database server (`hostB`).
- The secondary volume is not always mounted. It is mounted on the E drive (drive character `E:`) only while it is operating.
- The primary volume configured as a pair with the secondary volume to be backed up will not be backed up until the secondary volume is backed up onto tape.
- The `cscript //H:Cscript` command is executed from the command prompt and the host parameters are updated.

This example uses the following operation definition file, which is deployed in servers hostA and hostB, and has the name _opid_SQL.dat corresponding to the operation ID opid_SQL:

Contents of _opid_SQL.dat:

```
BACKUP_OBJECT=MSSQL
DB_SERVER_NAME=hostB
INSTANCE_NAME=DEFAULT
TARGET_NAME=USER_DB1
FTP_HOME_DIR=C:\FTP_ROOT
FTP_SUB_DIR=script
SET_DRM_HOSTNAME=0
```

This example uses the following host environment settings files. A host environment settings file is deployed on both server hostA and hostB.

The host environment settings file used for hostA:

```
HOST_ROLE=BK
MAX_LOG_LINES=1000
```

The host environment settings file used for hostB:

```
HOST_ROLE=DB
MAX_LOG_LINES=1000
```

This example prepares the following user script file named C:\Uscrip.txt:

Contents of C:\Uscrip.txt:

```
LOCAL_BACKUP=YES
[RESYNC_PROC]
# Target pair volume: SI01,dev01
[CMD]
CMDLINE=C:\HORCM\etc\pairresync.exe -g SI01 -d dev01 -FBC 0
ENV=HORCMINST=0
ENV=HORCC_MRCF=
END_CODE=TERMINATE_NZ
TIMEOUT=10
[CMD]
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g SI01 -d dev01 -s pair -FBC 0
END_CODE=TERMINATE_NZ
TIMEOUT=0
[SPLIT_PROC]
# Target pair volume: SI01,dev01
[CMD]
CMDLINE=C:\HORCM\etc\pairsplit.exe -g SI01 -d dev01 -FBC 0
END_CODE=TERMINATE_NZ
TIMEOUT=10
[CMD]
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g SI01 -d dev01 -s psus -FBC 0
END_CODE=TERMINATE_NZ
TIMEOUT=0
[FINISH_PROC]
#do nothing
```

7.9.1 Backing up a SQL Server Database in a Multi-Target Configuration

This section describes procedures used to back up a SQL Server database to a secondary volume at a local site while simultaneously backing it up to a secondary volume at a remote site (a backup in a multi-target configuration). The system uses `opid_SQL` as the operation ID.

To back up a SQL Server database in a multi-target configuration:

1. Clear the cache of the secondary volume.

   Before performing backup, clear the system cache of the backup server.

   To clear the system cache, execute `EX_DRM_CACHE_PURGE` on the backup server at the local site to mount and then unmount the secondary volume. In this example, the copy group name is set to `SI01,sql01`.

```
hostA > EX_DRM_CACHE_PURGE opid_SQL -cg SI01,sql01
```

2. Resynchronize the copy groups.

   To resynchronize the copy group, execute `EX_DRM_RESYNC` on the database server at the local site. Resynchronizing the copy groups before you perform backup enhances the speed of backup processing.

```
hostB > EX_DRM_RESYNC opid_SQL -cg SI01,sql01 -copy_size 7
```

   In both the following cases, the copy groups need to be resynchronized:

   - When using ShadowImage in Thunder 9500V Series or TagmaStore AMS/WMS Series
   - When using Thunder 9200

3. Back up the SQL Server database onto the secondary volume.

   Perform an online backup of the SQL Server database. To do this, execute `EX_DRM_SQL_BACKUP`. Specify the operation ID `opid_SQL` as the argument. Additionally, specify the user script file to be executed.

```
hostB > EX_DRM_SQL_BACKUP opid_SQL -script C:\Uscrip.txt
```

4. Export the backup information at the local site to temporary files.

   To back up data from the secondary volume to tape, copy the backup information to the backup server. To export the backup information to temporary files, execute `EX_DRM_DB_EXPORT`. These temporary files are stored in the temporary directory for extended commands, on the database server.

```
hostB > EX_DRM_DB_EXPORT opid_SQL
```

5. Transmit the temporary files to the backup server at the local site.

   Collectively transmit the temporary files from the database server to the backup server at the local site. To transmit the temporary files, execute `EX_DRM_FTP_PUT` on the database server. In this example, the user ID to log on to the FTP server is `admin` and the login password is `password`. The temporary files are stored in the temporary directory for extended commands, on the backup server.

```
hostB > EX_DRM_FTP_PUT opid_SQL -server hostA -user admin -password password
```

6. Import the temporary files, which were transmitted from the database server, into the backup catalog on the backup server at the local site.

   To import the temporary files, execute `EX_DRM_DB_IMPORT` on the backup server.

```
hostA > EX_DRM_DB_IMPORT opid_SQL
```

7. Back up the data of the secondary volume to tape.

   To back up the data, execute `EX_DRM_TAPE_BACKUP` on the backup server at the local site. In this example, the drive letter of the secondary volume is assumed to be `E:`.

```
hostA > EX_DRM_TAPE_BACKUP opid_SQL –mount_pt E:
```

   When this backup is executed, the backup information is registered, with a new backup ID of `0000000002`, in the backup catalog.

7.9.2 Restoring a SQL Server Database in a Multi-Target Configuration

This section provides an example of recovering a SQL Server database by restoring data backed up to tape at a local site. This example describes restoration from a local site. For details about performing restoration from a remote site, see section 7.12.5. The example uses `opid_SQL` as the operation ID.

To restore a SQL Server database from tape in a multi-target configuration:

1. Check the backup ID of the backup you want to use for restoration.

   To check the backup ID of the backup data to be used for restoration, execute the `drmtapecat` command from the backup server of the local site.

```
hostA> drmtapecat
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000002  MSSQL            2004/02/04 13:01:35  2004/05/04 13:01:35  MEDIA1
hostA>
```

2. Restore data from its backup tape to the secondary volume.

   To restore data, execute `EX_DRM_TAPE_RESTORE` on the backup server at the local site.

```
hostA > EX_DRM_TAPE_RESTORE opid_SQL -backup_id 0000000002
```

3. Export the backup information to temporary files.

   To restore data from the secondary volume to the primary volume, the backup information about the restore operation from tape to the secondary volume needs to be copied to the database server. To export the backup information to temporary files in the temporary directory for extended commands, execute `EX_DRM_DB_EXPORT` on the backup server at the local site.

```
hostA > EX_DRM_DB_EXPORT opid_SQL
```

4. Receive the temporary files on the database server.

   To receive the temporary files of the backup server on the database server, execute `EX_DRM_FTP_GET` on the database server at the local site. In this example, the user ID to log on to the FTP server is `admin` and the login password is `password`. The temporary files are stored in the directory for extended commands, on the database server.

```
hostB > EX_DRM_FTP_GET opid_SQL -server hostA -user admin -password password
```

5. Import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server.

   To import the temporary files, which were transmitted from the backup server, into the backup catalog on the database server, execute `EX_DRM_DB_IMPORT` on the database server at the local site.

```
hostB > EX_DRM_DB_IMPORT opid_SQL
```

6. Restore the data of the secondary volume to the primary volume.

   Resynchronize the primary volume and secondary volume and then restore the data. To restore data, execute `EX_DRM_SQL_RESTORE` on the backup server at the local site.

```
hostB > EX_DRM_SQL_RESTORE opid_SQL -resync
```

7. Recover the SQL Server database.

   To recover the SQL Server database, execute the `drmsqlrecover` or `drmsqlrecovertool` command on the database server at the local site.

   The following command illustrates execution of the `drmsqlrecover` command to recover the SQL Server database:

```
hostB > drmsqlrecover DEFAULT
KAVX0001-I The drmsqlrecover command will now start.
KAVX0002-I The drmsqlrecover command will now end.
hostB >
```

   The following command illustrates execution of the `drmsqlrecovertool` command to recover the SQL Server database:

```
hostB > drmsqlrecovertool DEFAULT
hostB >
```

   When the command is executed, the drmsqlrecovertool dialog box is displayed. For details about the drmsqlrecovertool dialog box, see the description of the `drmsqlrecovertool` command in the manual *HiCommand Protection Manager Command Reference*.

7.10 Performing Restoration and Recovery on a Host Other Than the Backup Host

This section explains by way of an example how to restore and recover SQL Server database data from a host other than the host that was used to make the backup. This explanation assumes the system configurations shown in Figure 7.30 and Figure 7.31. Operations are usually performed on database server 1, but when an error occurs on database server 1, the operations are switched to database server 2.

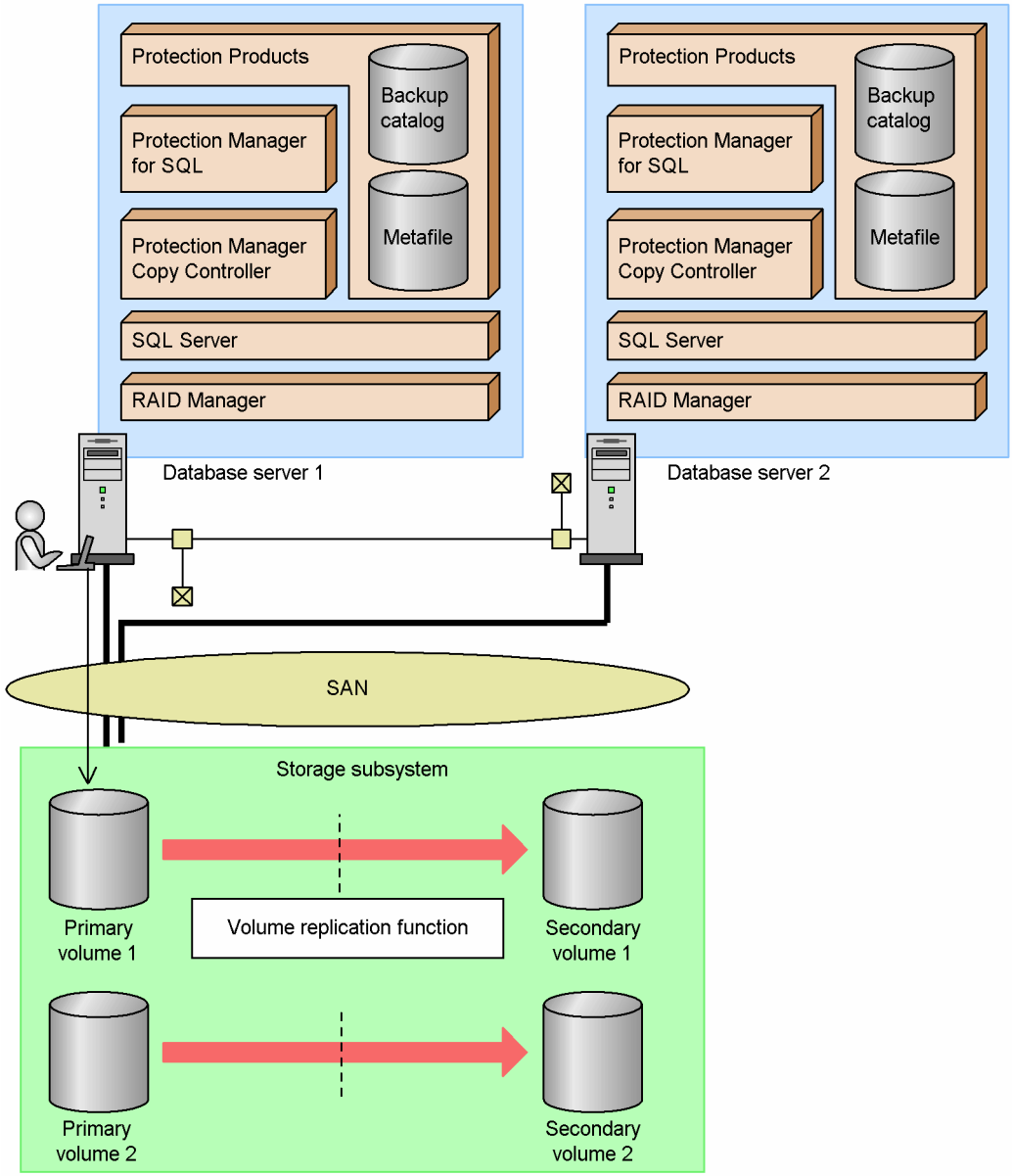


Figure 7.30 System Configuration for Restoring and Recovering from a Host not Used for Backup

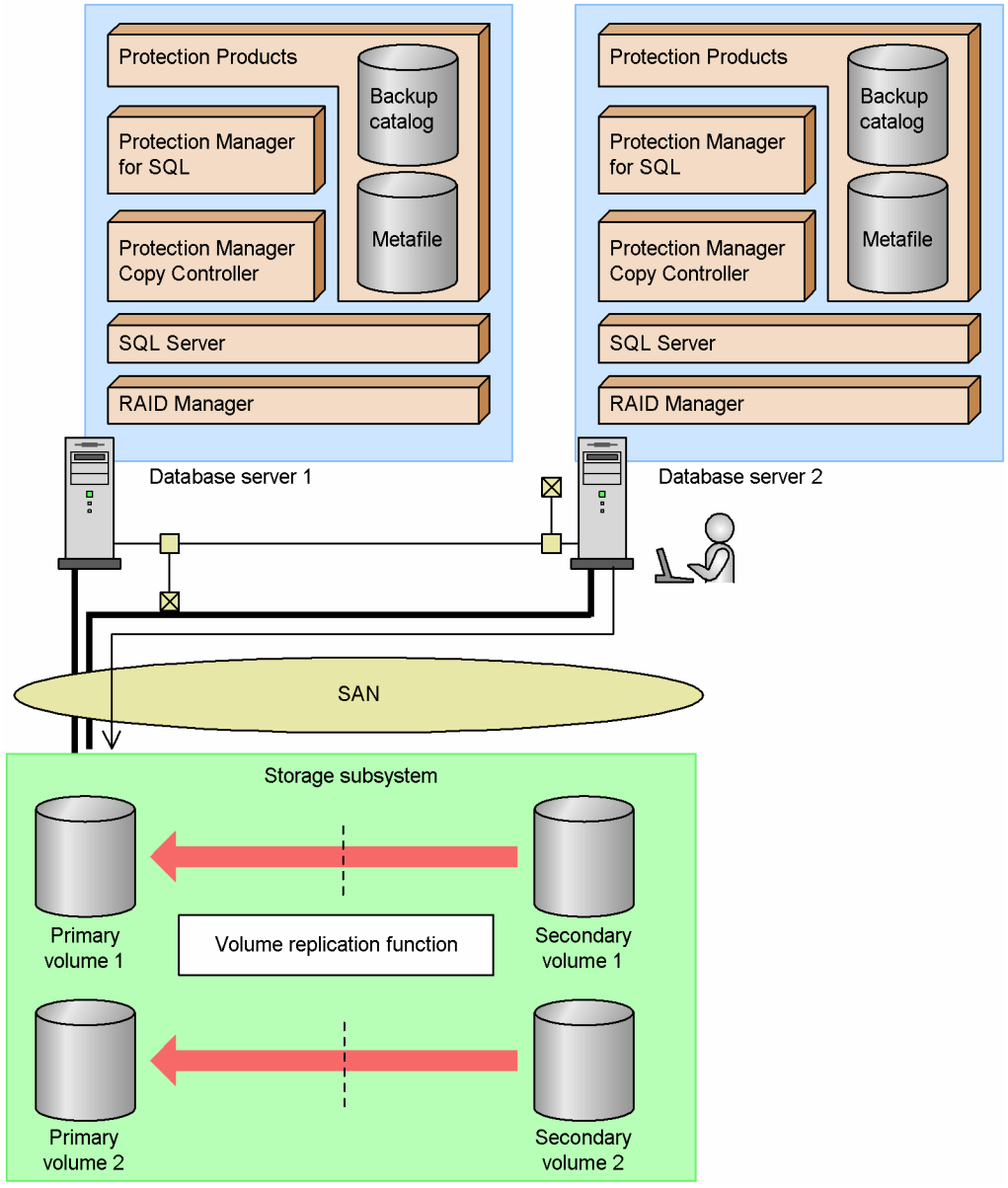


Figure 7.31 Restoring and Recovering from a Host not Used for Backup (When Restoring)

The following conditions are assumed:

- The primary volumes are formatted with NTFS, and are mounted on the D and E drives. Both volumes are used for the SQL Server database. A volume used for storing data files cannot also be used to store logs.
- The primary and secondary volumes are mirrored (as set by the volume replication function and RAID Manager).
- A metafile and backup catalog are on the server to be restored.
- The metafile pathnames are the same on both database server 1 and database server 2.
- The system databases (master, model, and msdb) are not specified when restoring.
- The `drmsqlinit` command has been executed on the server to be restored.
- An instance has been created on the server to be restored that is the same as the one on database server 1.
- For restoring, the primary volumes are mounted on database server 2.

To restore and recover SQL Server database data from a host that was not used to perform the backup:

1. Back up the SQL Server database to the secondary volumes.

   To perform an online backup of the SQL Server database, execute the `drmsqlbackup` command.

```
PROMPT> drmsqlbackup SQL1
KAVX0001-I The drmsqlbackup command will now start.
KAVX1017-I SQL Server information is being collected now.
KAVX1207-I Backing up the database...
KAVX0040-I The backup ID has been assigned as follows:
           Backup ID = 0000000001
KAVX0002-I The drmsqlbackup command will now end.
PROMPT>
```

   When you execute this command, the backup information is registered into the backup catalog on database server 1 under a new backup ID (the next unused ID). The following assumes that the new backup ID is `0000000001`.

2. Check the backup execution results.

   To confirm that backup to the secondary volumes finished successfully, execute the `drmsqlcat` command.

```
PROMPT> drmsqlcat SQL1
INSTANCE: SQL1
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: SQL1 ORIGINAL-ID: 0000000001
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: SQL1
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M  DB1  METAFILE  C:\METADIR\Meta1 C:  -   -          -
D  DB1  DATAFILE  D:\SQL\data1     D:  -   Harddisk1  SQL001,B001
-  -    -         -                -   -   -          SQL001,B001
T  DB1  TRANSACT  E:\SQL\tlog      E:  -   Harddisk2  SQL002,B001
-  -    -         -                -   -   -          SQL002,B001
PROMPT>
```

3. Export the backup information to a temporary file.

   You must copy to database server 2 the information pertaining to the operations performed in backing up the primary volumes to the secondary volumes. To export the information pertaining to the backup operations to the temporary file `C:\temp\EX-FILE1.drm`, execute the `drmdbexport` command.

```
PROMPT> drmdbexport 0000000001 -f C:\temp\EX-FILE1.drm
KAVX0001-I The drmdbexport command will now start.
KAVX0002-I The drmdbexport command will now end.
PROMPT>
```

4. Transfer the exported temporary file from database server 1 to database server 2.

   Use the `ftp` (file transfer protocol) command to transfer the exported temporary file `C:\temp\EX-FILE1.drm` from database server 1 to database server 2.

```
PROMPT> ftp database-server2-name
ftp> Username: (logon-name)
ftp> password: (password)
ftp> binary
ftp> put C:\temp\EX-FILE1.drm
ftp> quit
PROMPT>
```

5. Import the temporary file, which was transferred from database server 1, into the backup catalog on database server 2.

   To import the temporary file transferred from database server 1 `C:\temp\EX-FILE1.drm` into the backup catalog on database server 2, execute the `drmdbimport` command on database server 2.

```
PROMPT> drmdbimport -f C:\temp\EX-FILE1.drm
KAVX0001-I The drmdbimport command will now start.
KAVX0418-I Backup information from the specified file was imported into the backup catalog.
           Imported backup ID = 0000000001
KAVX0002-I The drmdbimport command will now end.
PROMPT>
```

6. Check the import results.

   To confirm that the import finished successfully, execute the `drmsqlcat` command on database server 2.

```
PROMPT> drmsqlcat SQL1
INSTANCE: SQL1
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: SQL1 ORIGINAL-ID: 0000000001
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: SQL1
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M  DB1  METAFILE  C:\METADIR\Meta1 C:  -   -          -
D  DB1  DATAFILE  D:\SQL\data1     D:  -   Harddisk1  SQL001,B001
-  -    -         -                -   -   -          SQL001,B001
T  DB1  TRANSACT  E:\SQL\tlog      E:  -   Harddisk2  SQL002,B001
-  -    -         -                -   -   -          SQL002,B001
PROMPT>
```

7. Restore the data on the secondary volumes to the primary volumes.

   To restore the data and resynchronize the primary and secondary volumes, execute the `drmsqlrestore` command on database server 2.

```
PROMPT> drmsqlrestore 0000000001 -resync -nochk_host
KAVX0001-I The drmsqlrestore command will now start.
KAVX1316-I Performing restore pre-processing...
KAVX1000-I Logged on to the SQL Server system.
KAVX1300-I Restore processing will now start.
KAVX1000-I Logged on to the SQL Server system.
KAVX1301-I Restore processing has finished.
KAVX1318-I Performing restore post-processing...
KAVX1001-I Logged off from the SQL Server system.
KAVX0002-I The drmsqlrestore command will now end.
PROMPT>
```

8. Recover the SQL Server database.

   To recover the SQL Server database, execute the `drmsqlrecover` or `drmsqlrecovertool` command on database server 2.

   - The following example uses the `drmsqlrecover` command:

     ```
     PROMPT> drmsqlrecover SQL1
     KAVX0001-I The drmsqlrecover command will now start.
     KAVX1017-I SQL Server information is being collected now.
     KAVX1302-I Recovery processing will now start.
     KAVX1303-I Recovery processing has finished.
     KAVX0002-I The drmsqlrecover command will now end.
     PROMPT>
     ```

   - The following example uses the `drmsqlrecovertool` command:

     ```
     PROMPT> drmsqlrecovertool SQL1
     PROMPT>
     ```

     When you execute this command, the drmsqlrecovertool dialog box appears. For details on this dialog box, see the description of the `drmsqlrecovertool` command in the *HiCommand Protection Manager Command Reference,* MK-94HC072.

7.11 Using the Log-Shipping Function of a SQL Server Database

7.11.1 Before Using the Log-Shipping Function

When a backup target is the SQL Server database, the log-shipping functionality of Microsoft SQL Server 2000 Enterprise Edition and Microsoft SQL Server 2005 can be used. Using the log-shipping functionality, you can continuously copy backed-up transaction logs to a distribution server so that it can be used as an operation server. For details on the log-shipping functionality, see the SQL Server manual.

This section explains an example of synchronizing the database on a source server and the database on a distribution server and setting log-shipping operations on the source server, before using the log-shipping functionality of Microsoft SQL Server 2000 Enterprise Edition or Microsoft SQL Server 2005. This explanation assumes the system configuration shown in the Figure 7.32.

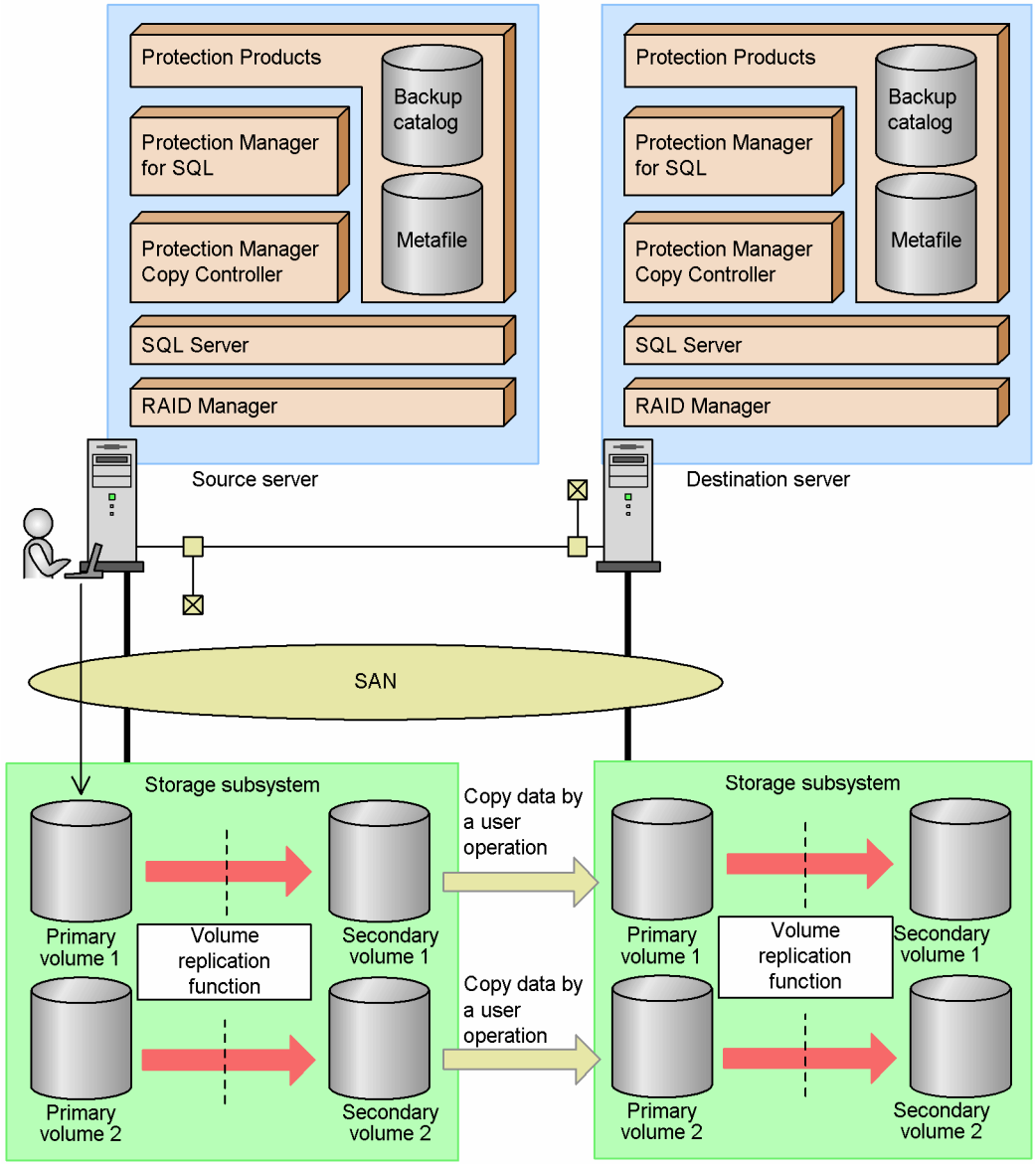


Figure 7.32 System Configuration for Log-Shipping Operations

The following conditions are assumed:

- The primary volumes on the source server and the volumes on the distribution target server are formatted with NTFS, and are mounted on the D and E drives, respectively.
- The primary and secondary volumes on the source server are mirrored (as set by the volume replication function and RAID Manager).
- The secondary volumes on the source server are formatted with NTFS, and are not mounted.
- The `drmsqlinit` command has been executed on the distribution target server.
- The same instance has been created on both the source server and the distribution target server.
- The FTP server is running.
- The system databases (master, model, and msdb) are not specified when restoring.

The following explains the procedure for synchronizing the database on the source server and the database on the distribution server and setting log-shipping operations on the source server, for using the log-shipping functionality.

To set up the log-shipping functionality:

1. Back up the SQL Server database to the secondary volumes.

   To perform an online backup of the SQL Server database, execute the `drmsqlbackup` command on the source server.

```
PROMPT> drmsqlbackup SQL1
KAVX0001-I The drmsqlbackup command will now start.
KAVX1017-I SQL Server information is being collected now.
KAVX1207-I Backing up the database...
KAVX0040-I The backup ID has been assigned as follows:
           Backup ID = 0000000001
KAVX0002-I The drmsqlbackup command will now end.
PROMPT>
```

   When you execute this command, the backup information is registered in the backup catalog on the source server under a new backup ID (the next unused ID). The following assumes that the new backup ID is `0000000001`.

2. Check the backup execution results.

   To confirm that backup to the secondary volumes finished successfully, execute the `drmsqlcat` command on the source server.

```
PROMPT> drmsqlcat SQL1
INSTANCE: SQL1
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: SQL1 ORIGINAL-ID: 0000000001
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: SQL1
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M  DB1  METAFILE  C:\METADIR\Meta1 C:  -   -          -
D  DB1  DATAFILE  D:\SQL\data1     D:  -   Harddisk1  SQL001,B001
-  -    -         -                -   -   -          SQL001,B001
T  DB1  TRANSACT  E:\SQL\tlog      E:  -   Harddisk2  SQL002,B001
-  -    -         -                -   -   -          SQL002,B001
PROMPT>
```

3. Export the backup information to a temporary file.

   You must copy, to the distribution target server, the information pertaining to the operations performed in backing up the primary volumes to the secondary volumes. To export the information pertaining to the backup operations to the temporary file `C:\temp\EX-FILE1.drm`, execute the `drmdbexport` command on the source server.

```
PROMPT> drmdbexport 0000000001 -f C:\temp\EX-FILE1.drm
KAVX0001-I The drmdbexport command will now start.
KAVX0002-I The drmdbexport command will now end.
PROMPT>
```

4. Transfer the metafile and the temporary file from the source server to the distribution target server.

   Use the `ftp` (file transfer protocol) command to transfer the metafile `C:\METADIR\Meta1` and the exported temporary file `C:\temp\EX-FILE1.drm` from the source server to the distribution target server.

```
PROMPT> ftp distribution-destination-server-name
ftp> Username: (logon-name)
ftp> password: (password)
ftp> binary
ftp> put C:\temp\EX-FILE1.drm
ftp> put C:\METADIR\Meta1
ftp> quit
PROMPT>
```

5. Copy the backup data.

   Copy to the distribution target server volumes the data that was backed up to the secondary volumes in step 1.

6. Import the temporary file into the backup catalog on the distribution target server.

   To import the temporary file `C:\temp\EX-FILE1.drm` (transferred from the source server in step 4) into the backup catalog on the distribution target server, execute the `drmdbimport` command on the distribution target server.

```
PROMPT> drmdbimport -f C:\temp\EX-FILE1.drm
KAVX0001-I The drmdbimport command will now start.
KAVX0418-I Backup information from the specified file was imported into the backup catalog.
           Imported backup ID = 0000000001
KAVX0002-I The drmdbimport command will now end.
PROMPT>
```

7. Check the import results.

   To confirm that the import finished successfully, execute the `drmsqlcat` command on the distribution target server.

```
PROMPT> drmsqlcat SQL1
INSTANCE: SQL1
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: SQL1 ORIGINAL-ID: 0000000001
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: SQL1
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M  DB1  METAFILE  C:\METADIR\Meta1 C:  -   -          -
D  DB1  DATAFILE  D:\SQL\data1     D:  -   Harddisk1  SQL001,B001
-  -    -         -                -   -   -          SQL001,B001
T  DB1  TRANSACT  E:\SQL\tlog      E:  -   Harddisk2  SQL002,B001
-  -    -         -                -   -   -          SQL002,B001
PROMPT>
```

8. Restore the backup data to the distribution target server volumes.

   Check that the backup data has been copied to the distribution target server volumes, and then restore the backup data from the distribution target server. When restoring, apply the metafile to the backup data without recovering the backup data from the secondary volumes to the primary volumes, and place the database into standby status. To restore in this manner, execute the `drmsqlrestore` command as shown below:

```
PROMPT> drmsqlrestore 0000000001 -no_resync -undo -nochk_host
KAVX0001-I The drmsqlrestore command will now start.
KAVX1316-I Performing restore pre-processing...
KAVX1000-I Logged on to the SQL Server system.
KAVX1300-I Restore processing will now start.
KAVX1000-I Logged on to the SQL Server system.
KAVX1301-I Restore processing has finished.
KAVX1318-I Performing restore post-processing...
KAVX1001-I Logged off from the SQL Server system.
KAVX0002-I The drmsqlrestore command will now end.
PROMPT>
```

9. Set up log-shipping operations on the source server.

   Specify the settings, such as the distribution server and the frequency of log creation, for log-shipping operations on the source server.

   For information on how to set up log-shipping operations, see the SQL Server documentation.

7.11.2 Setting a Distribution Target Server as an Operating Server

This section explains by way of an example how to set a distribution target server to be an operating server when using the log-shipping functionality of Microsoft SQL Server 2000 Enterprise Edition or Microsoft SQL Server 2005, if such is required due to a failure on the source server during log-shipping operations or for some other reason. This explanation assumes the system configuration shown in Figure 7.33.

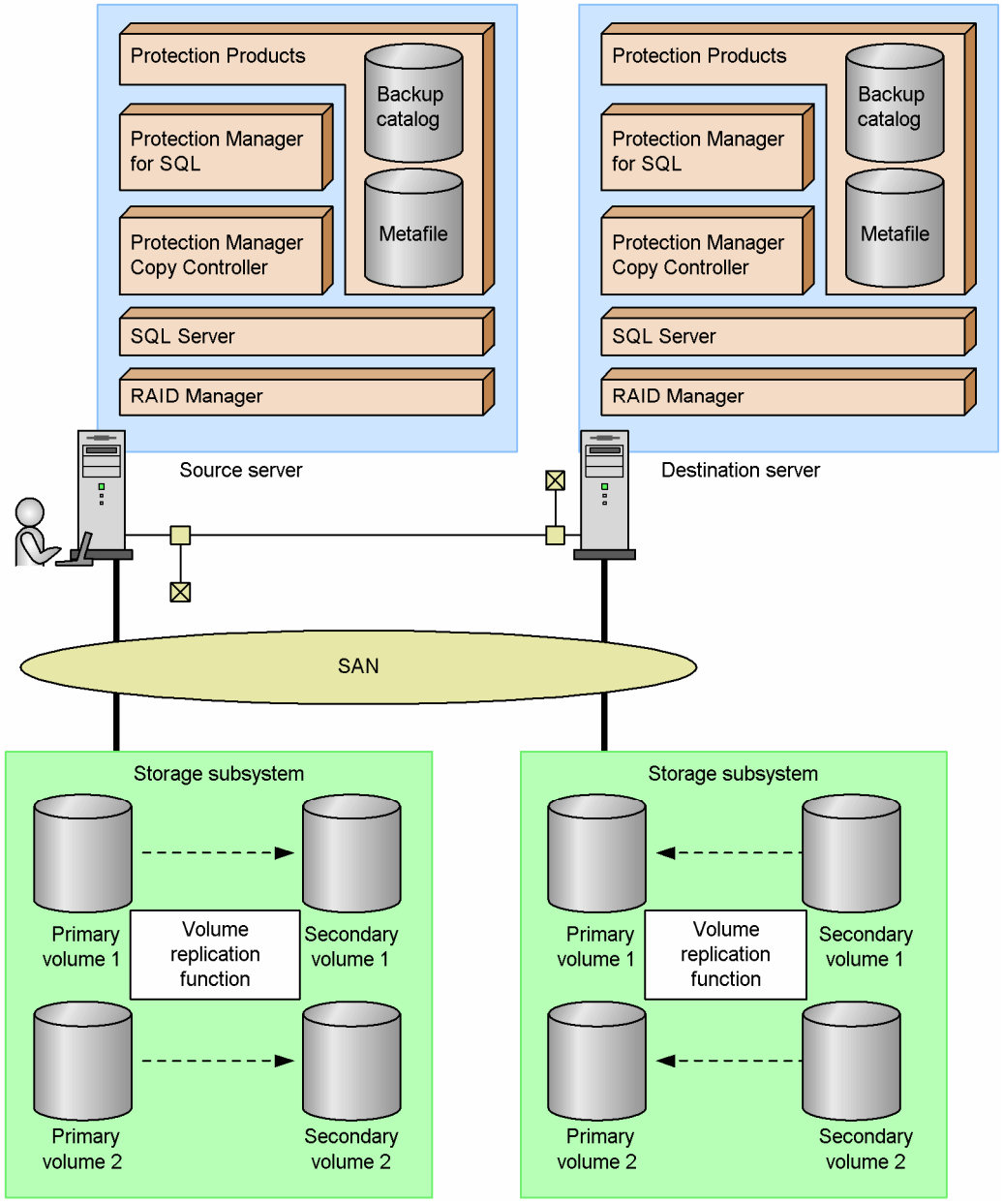


Figure 7.33 System Configuration for Setting a Distribution Target Server to be an Operating Server

The following conditions are assumed:

- The distribution target server volumes are formatted with NTFS, and are mounted on the D and E drives.
- The `drmsqlinit` command has been executed on the distribution target server.
- The system databases (master, model, and msdb) are not specified when restoring.
- The database on the distribution target server is in the standby status.

To set a distribution target server to be an operation server:

1. Recover the SQL Server database.

   To recover the SQL Server database, execute the `drmsqlrecover` command on the distribution target server.

```
PROMPT> drmsqlrecover SQL1
KAVX0001-I The drmsqlrecover command will now start.
KAVX1302-I Recovery processing will now start.
KAVX1303-I Recovery processing has finished.
KAVX0002-I The drmsqlrecover command will now end.
PROMPT>
```

2. Place the standby database online.

   To use a Protection Products command after restoring the database, execute `drmsqldisplay` set to the `-refresh` option.

7.12 Backup and Restore in a Cascade or Multi-Target Configuration

Protection Manager for SQL supports backup to, and restoration from, a secondary volume at a remote site (in a cascade configuration). When backing up data to the secondary volume at the remote site, you can also back up the same data on the secondary volume at the local site (backup in a multi-target configuration).

This section explains how to back up a SQL Server database to, and restore it from, a secondary volume at a remote site (how to perform backup and restoration in a cascade configuration). For backup operation in a cascade configuration, a user-created script file is used.

This example is based on the system configuration shown in Figure 7.34.

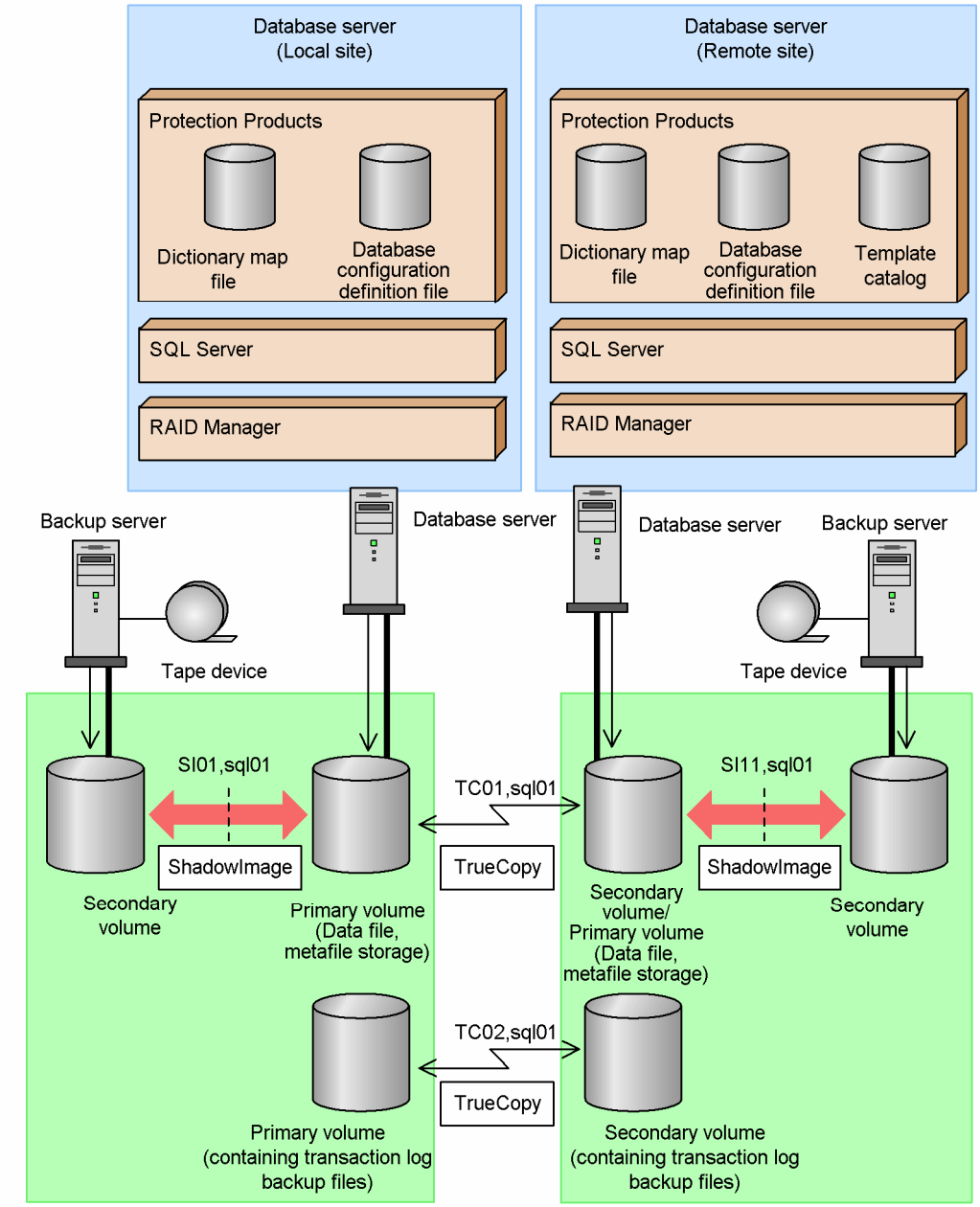


Figure 7.34 Sample Configuration for Backing up to a Secondary Volume at a Remote Site (Cascade Configuration)

The following conditions are assumed:

- The primary volume at the local site and the primary volume at the remote site must be defined as a pair in TrueCopy. The primary and secondary volumes at each site must be defined as a pair in ShadowImage.
- The same RAID Manager instance number is defined at both the local and remote sites.
- The SQL Server data files and transaction log files must have the same directory structure at both the local and remote sites.
- The SQL Server database for which backup operations are to be performed and the database ID must have the same name at both the local and remote sites.
- Both the SQL Server database data files and the VDI metafile storage directory must be stored in the same volume.
- The TrueCopy pair status must be `PAIR` when the target volume contains any SQL server data file, transaction log file, and transaction log backup file.
- When the cluster configuration is used at a remote site or local site, the TrueCopy pair volume must not contain a quorum disk.
- The SQL Server service at the remote site must be stopped in advance when the TrueCopy pair contains data files or transaction log files for the system databases (master, model, and msdb).
- The disk at the remote site must be unmounted in advance when the status of the TrueCopy pair disk is `PAIR`.

7.12.1 Preparing to Backup or Restore in a Cascade or Multi-Target Configuration

Before backup or restoration can be performed in a cascade or multi-target configuration, the following preparation is required.

To prepare for backup or restore operations in a cascade or multi-target configuration:

1. Confirm that the pair status of the TrueCopy pair volume is `SMPL` or `PSUS`.

   To confirm the pair status of the TrueCopy pair volume, execute the RAID Manager `pairdisplay` command at the local site.

```
PROMPT> pairdisplay -g TC01 -CLI
```

2. Check the database ID of the database for which backup operations are to be performed.

   Check the database ID at the local site:

```
PROMPT> osql -E
1> SELECT DB_ID("UserDB1")
2> go
```

3. Stop the SQL Server service and unmount the TrueCopy pair volume.

   Perform operations in step 3 at the local site.

4. After the pair status of the TrueCopy pair volume is changed to `PAIR`, perform takeover.

To perform takeover, execute the RAID Manager command at the remote site.

```
PROMPT> paircreate -g TC01 -vr ( or pairresync -g TC01 )
PROMPT> pairevtwait -g TC01 -t 600 -s pair
PROMPT> horctakeover -g TC01
```

5. Mount the TrueCopy pair volume at the remote site and restart SQL Server.

   The mount point name to mount must be same as the mount point name at the local site. Perform operations in step 5 at the remote site.

6. Attach the database from the mounted disk to check the database ID.

   Check the database ID at the remote site:

```
PROMPT> osql -E
1> SELECT DB_ID("UserDB1")
2> go
```

   If the database ID at the remote site is not the same as the database ID at the local site checked in step 2, set the database ID at the remote site to the same database ID as one at the local site.

7. Stop the SQL Server service and unmount the TrueCopy pair volume.

   Perform operations in step 7 at the remote site.

8. Perform takeover.

   To perform takeover, execute the RAID Manager `horctakeover` command at the remote site:

```
PROMPT> horctakeover -g TC11
```

9. Mount the TrueCopy pair volume at the local site and start the SQL Server service.

10. Create a database configuration definition file.

    To create a database configuration definition file, execute the `drmsqlinit` command on the database server at the local site.

```
PROMPT> drmsqlinit default
KAVX1100-I Enter the name of the storage directory for VDI metafiles:
KAVX1101-I Enter the timeout interval for VDI generation (in seconds, between 0 and 3600):
300
KAVX1102-I Enter the name of the storage directory for undo files: E:\MSSQL\DEFAULT\undo
KAVX1107-I Enter the name of the directory that contains the backup logs:
E:\MSSQL\DEFAULT\tlog
KAVX1103-I SQL Server information has been updated.
```

11. Update the dictionary map file.

    To update the dictionary map file, execute the `drmsqldisplay` command on the database server at the local site.

```
PROMPT> drmsqldisplay -refresh
KAVX0023-I Dictionary map file was updated.
```

12. Create a user script file.

To obtain the information required to create a user script file, execute the `drmsqldisplay` command on the database server at the local site. For details on how to create a user script file, see section 4.16.

```
PROMPT> drmsqldisplay default -remote
INSTANCE: default
T DB       OBJECT    FILE                      L-COPY-GROUP   R-COPY-GROUP   PORT#   TID#
LUN#   MU#   LDEV#   P/S   SERIAL#
D UserDB1   DATAFILE   Z:\UserDB1_Data.MDF         -              -            CL1-A   1
32     -     32      P     116
- -         -          -                          SI11,dev01     -            CL1-A   1
60     0     60      S     116
- -         -          -                          TC11,dev11     -            CL1-A   1
32     0     32      S     600
- -         -          -                          SI01,sql01     -            CL1-A   1
60     0     60      S     116
- -         -          -                          TC01,sql01     -            CL1-A   1
32     0     32      S     600
- -         -          -                          TC01,sql01     TC01,sql01#0   -       -
-      0     60      C     600
T UserDB1   TRANSACT   Z:\UserDB1_Log.LDF          -              -            CL1-A   1
32     -     32      P     116
- -         -          -                          SI01,sql01     -            CL1-A   1
60     0     60      S     116
- -         -          -                          TC01,sql01     -            CL1-A   1
32     0     32      S     600
- -         -          -                          TC01,sql01     TC01,sql01#0   -       -
-      0     60      C     600
PROMPT>
```

13. Stop the SQL Server service and unmount the TrueCopy pair volume at the local site.

14. To change the secondary volume at the remote site to the primary volume, perform takeover.

    To perform takeover, execute the RAID Manager command `horctakeover` at the local site.

```
PROMPT> horctakeover -g TC01 -d sql01
PROMPT> pairsplit -g TC01 -d sql01 -rw
```

15. Mount the TrueCopy pair volume at the remote site and start the SQL Server service.

16. Create a database configuration definition file at the remote site.

    To create a database configuration definition file, execute the `drmsqlinit` command on the database server at the remote site. Specify the same information you specified for the local site in step 1.

```
PROMPT> drmsqlinit default
KAVX1100-I Enter the name of the storage directory for VDI metafiles:
KAVX1101-I Enter the timeout interval for VDI generation (in seconds, between 0 and 3600):
KAVX1102-I Enter the name of the storage directory for undo files: E:\MSSQL\DEFAULT\undo
KAVX1107-I Enter the name of the storage directory for backup-log files:
E:\MSSQL\DEFAULT\tlog
KAVX1103-I SQL Server information has been updated
```

17. Update the dictionary map file at the remote site.

    To update the dictionary map file, execute the `drmsqldisplay` command on the database server at the remote site.

```
PROMPT> drmsqldisplay -refresh
KAVX0023-I Dictionary map file was updated.
```

18. Create a template catalog at the remote site.

    To create a template catalog, execute the `drmsqlbackup` command on the database server at the remote site. The `drmsqlbackup` command does not perform backup when it is used to create a template catalog.

```
PROMPT> drmsqlbackup default -template
KAVX0001-I The drmsqlbackup command will now start.
KAVX1017-I SQL Server information is being collected now.
KAVX0085-I Now making backup information...
KAVX0086-I The backup information has been assigned as follows:
           Backup ID = 0000000001
KAVX0002-I The drmsqlbackup command will now end.
```

    If you plan to perform backup at the local site by using the `drmsqlbackup` command with the `-target` or `-f` option specified, create a template catalog according to the option setting.

```
PROMPT> drmsqlbackup default -target UserDB1 -template
KAVX0001-I The drmsqlbackup command will now start.
KAVX1017-I SQL Server information is being collected now.
KAVX0085-I Now making backup information...
KAVX0086-I The backup information has been assigned as follows:
           Backup ID = 0000000002
KAVX0002-I The drmsqlbackup command will now end.
```

    You can `also` delete the template catalog:

```
PROMPT> drmappcat 0000000001 -delete
KAVX0424-I The backup information corresponding to the specified backup ID has been
deleted.
           Backup ID = 0000000001
```

19. Detach the SQL Server database.

    Use SQL Server Enterprise Manager or an equivalent tool to detach the user database at the remote site you want to back up. If the backup targets include the system databases (master, model, and msdb), stop the SQL Server service at the remote site and unmount the TrueCopy pair volume.

20. Export the template catalog at the remote site.

    To export the template catalog, execute the `drmdbexport` command on the database server at the remote site. Save the exported catalog file in a file for storage because it will be required for restoration.

```
PROMPT> drmdbexport 0000000002 -f c:\templateCat.drm
KAVX0001-I The drmdbexport command will now start.
KAVX0002-I The drmdbexport command will now end.
```

21. To change the secondary volume at the local site to the primary volume, perform takeover.

    To perform takeover, execute the RAID Manager command `horctakeover` at the local site.

```
PROMPT> pairresync -g TC01 -d sql01
PROMPT> horctakeover -g TC01 -d sql01
```

22. Mount the TrueCopy pair volume at the local site and start the SQL Server service.

7.12.2 Creating a User Script to Perform Backup in a Cascade Configuration

To perform backup in a cascade configuration, you need to create a user script file. This section consists of the following topics:

- Outline of user script files
- Coding rules for user script files
- Sample user script files

7.12.2.1 Outline of User Script Files

If the `drmsqlbackup` command is executed with the `-script` option specified, the command reads the user-specified script file and, at the following points in time, executes the commands defined in the script file:

- After ShadowImage pair resynchronization
- While databases are frozen
- After databases are thawed

Figure 7.35 shows points in time at which a user script is executed.

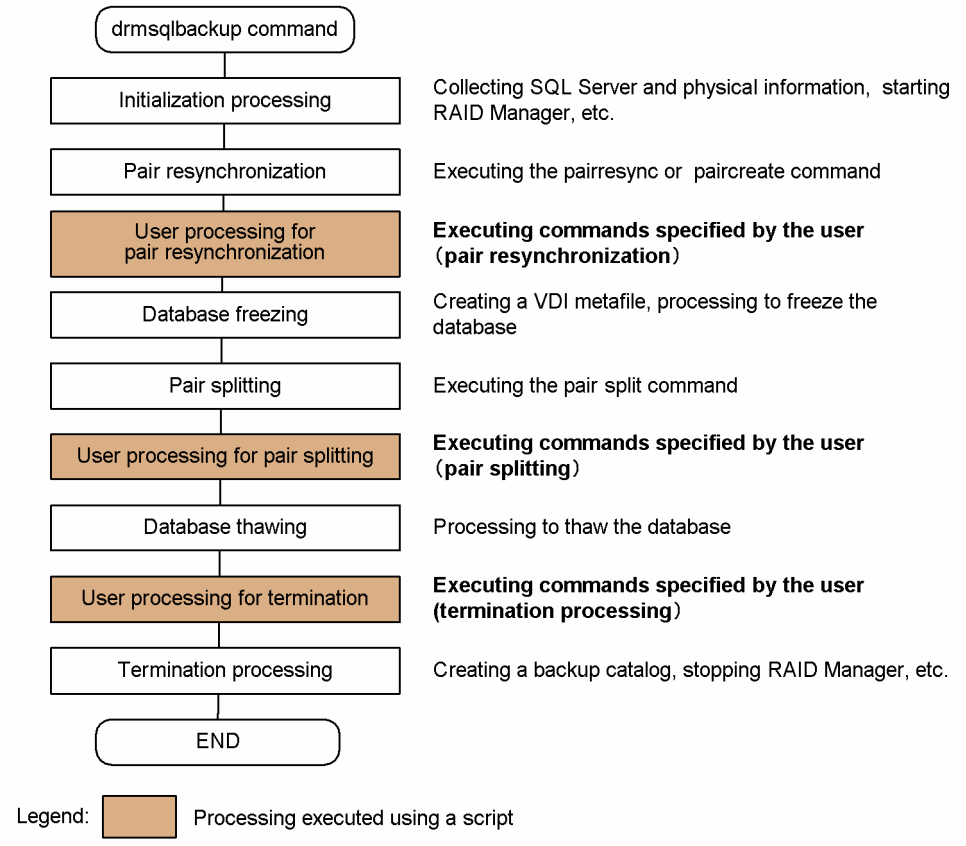


Figure 7.35 Points in Time at Which a User Script is Executed

The following are notes on the user script file:

- If a command defined in the user script file does not actually exist, it will not be executed.
- If the user script file includes the specification that local backup should not be performed (that is, `NO` is assigned to the `LOCAL_BACKUP` item), the `drmsqlbackup` command will simply execute the user script, without performing pair resynchronization, pair splitting, or backup catalog creation at the local site's secondary volume.
- When a command specified in the user script file executes, the `drmsqlbackup` command waits until the command specified in the user script file finishes or times out.
- If you set `TIMEOUT` to `0`, the `drmsqlbackup` command does not finish unless the command executed by the user script file finishes. Keep this in mind when you specify the timeout setting.
- When you want to stop an executing command (for example, when a command executed by a user script file does not respond), enter **Ctrl+C** from the keyboard, or use the Windows Task Manager to terminate the command processing. (The command processing is executed by the user script file, and not by the `drmsqlbackup` command.)

7.12.2.2 Coding Rules for User Script Files

For details about the coding rules for user script files, see section 4.16.2.

Table 7.6 Coding Rules for User Script Files

Applicable items	Rules
Entire user script file	▪ The character code set must be ASCII. ▪ Codes smaller than `0x20` cannot be used, except for `CR` (`0x0d`), `LF` (`0x0a`), and `TAB` (`0x09`). ▪ The linefeed code must be `LF` (`0x0a`) or `CR+LF` (`0x0d` and `0x0a`). ▪ One line must not exceed 8 KB.
Item and section names	▪ One line must contain only one item. ▪ Item and section names are case-insensitive. ▪ Item and section names must consist only of one-byte characters. Two-byte characters cannot be used. ▪ Item names must be separated by a one-byte equal sign (=). One-byte space or tab characters can be inserted before and after each one-byte equal sign.
Item values	▪ The value of an item must be specified after the one-byte equal sign (=) that follows the item name. Do not enter a linefeed code between the name and value of an item. ▪ The item value begins after the one-byte equal sign (=) that follows the item name and ends before the linefeed code. ▪ If an item name is specified alone (without a value), it will cause an error.
Comments	▪ A line that begins with a one-byte hash mark (#) is regarded as a comment line. ▪ Even if a one-byte hash mark (#) is entered in the middle part of a line, the part of the line after the hash mark will not be regarded as a comment. ▪ If the part of the line before the hash mark (#) consists only of tab and one-byte space characters, the line will be regarded as a comment line.
Blank lines	▪ You can enter blank lines. ▪ A line that consists only of tab or one-byte space characters is regarded as a blank line.

Table 7.7 Entries in User Script Files (continues on following pages)

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
LOCAL_BACKUP	Specifies whether to perform local backup: ▪ If you want to perform local backup, specify YES. ▪ If you do not want to perform local backup, specify NO. ▪ If local backup is specified, enter the entry before other entries. ▪ Specify the value by using one-byte character.	Not allowed	Not allowed
[RESYNC_PROC][2]	User processing section: Indicates the beginning of the "user processing for pair resynchronization". In this section, define commands to be executed as part of user processing for pair resynchronization.	Not allowed	Allowed[3]
[SPLIT_PROC][2]	User processing section: Indicates the beginning of the "user processing for pair splitting". In this section, define commands to be executed as part of user processing for pair splitting.	Not allowed	Allowed[3]
[FINISH_PROC][2]	User processing section: Indicates the beginning of the "user processing for termination". In this section, define commands to be executed as part of user processing for termination.	Not allowed	Allowed[3]
[CMD]	Indicates the beginning of a command definition section. ▪ This item must be specified within a user processing section. ▪ If two or more command definitions are included in your code, they will be executed individually in the order in which they are coded. In this case, the next command starts executing after the previously executed command finishes or times out. (Therefore, only one command executes at a time, always.) ▪ The item specified in the command definition section is valid only on the command definition section for the target command. The other command definition sections are not affected.	Allowed[4]	Allowed[3]

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
CMDLINE[5]	Specifies a command line to be executed. ▪ If you want to specify an option, separate the command name and the option with a one-byte space character. ▪ One command definition section must contain only one command line. ▪ A maximum of 2048 characters can be specified on a command line.[6] ▪ A path or file name containing one or more spaces must be enclosed in double quotation marks ("). ▪ Command and file names must be specified using absolute path names. ▪ Do not include any environment variable in a command name or command path name (for example, a specification like %SystemRoot%\notepad.exe is not allowed). ▪ If you want to use a shell command such as dir, execute it as a child process of cmd.exe. An example is C:\WINNT\System32\cmd.exe /c dir.	Not allowed	Not allowed
ENV[5]	Specifies the environment variables to be applied for execution of the specified command. ▪ Only one environment variable is allowed per line. ▪ The name and value of an environment variable must be separated by a one-byte equal sign (=). ▪ One environment variable definition can consist of up to 2048 characters.[6] ▪ If you omit the value for the environment variable, the environment variable is deleted. For example, the environment variable ABC is deleted when you specify ENV=ABC=. ▪ The ENV setting is valid only for the target command. Note that the subsequent command does not inherit the setting contents).	Allowed[4]	Allowed
END_CODE[5]	Specifies the action to be performed in response to the return value of the executed command. Specifiable values are as follows: ▪ TERMINATE_NZ Terminates script processing if a non-zero return value returns. ▪ IGNORE Continue processing whatever may be returned for a return value. ▪ If the executed command ends in a timeout, a value other than 0 is assumed to be returned. ▪ Specify the value by using a one-byte character.	Not allowed	Allowed
TIMEOUT[5]	Specifies the command timeout value (in seconds). ▪ The range of specifiable values is from 0 to 86400. ▪ Use one-byte numeric characters only. ▪ If this item is omitted, the timeout value is assumed to be 600 (10 minutes).	Not allowed	Allowed

Entry name	Meaning and specifiable values	Multiple specification[1]	Omission
	▪ No timeout will occur if 0 is specified here.		

[1] The [RESYNC_PROC], [SPLIT_PROC], and [FINISH_PROC] can be defined in any order.

[2] "Multiple specification is not allowed" means that multiple entries cannot be specified in the command definition section (CMD).

[3] This item can be omitted if there is no command to be executed within the user processing section.

[4] This item can be used more than once within a user script file.

[5] Within the [CMD] item, CMDLINE, ENV, END_CODE, and TIMEOUT can be defined in any order.

[6] For these characters, one-byte characters and two-byte characters are not distinguished.

7.12.2.3 Sample User Script Files

This section provides some practical examples of user script files. The remainder of this section explains how to write user script files using examples that assume the following cascade configuration (Figure 7.36):

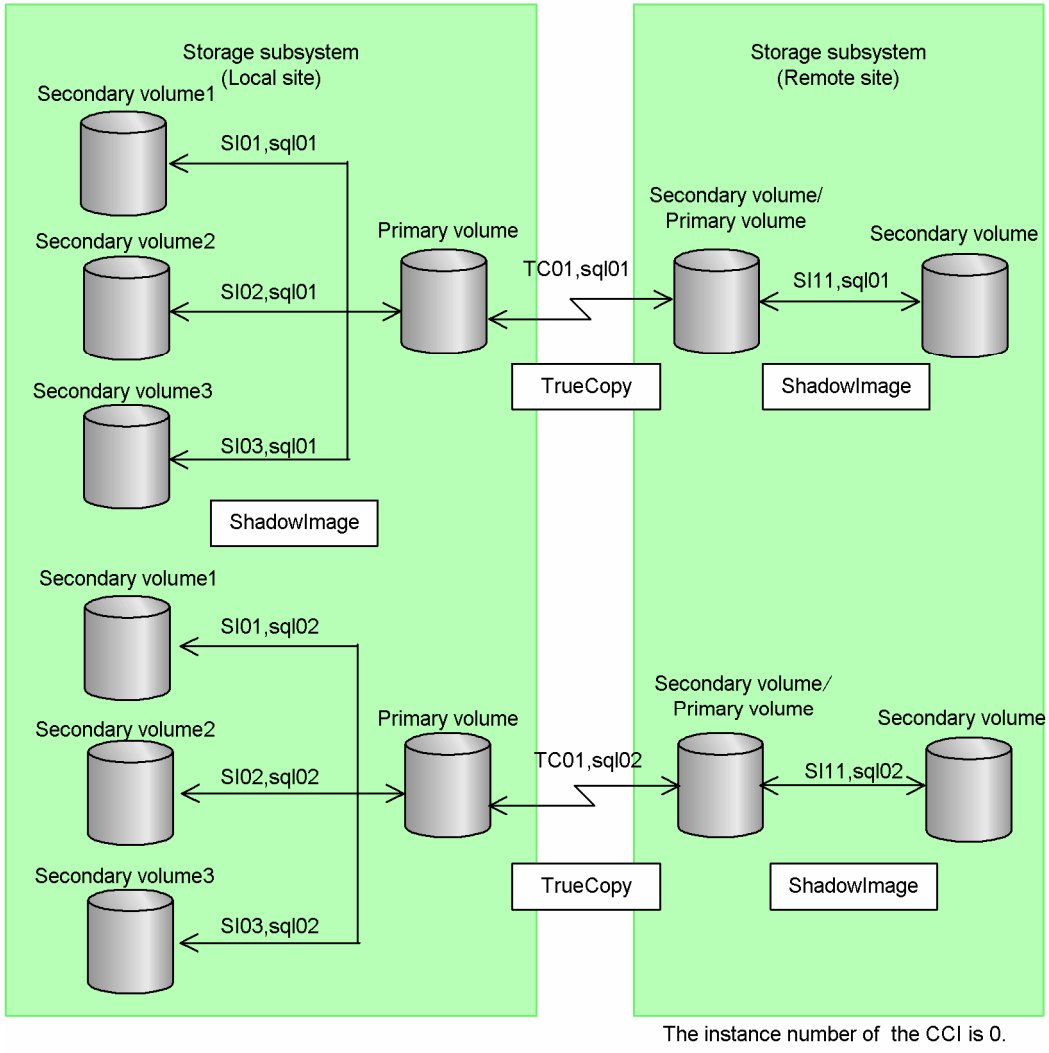


Figure 7.36 Sample Cascade Configuration

If the `drmsqldisplay` command is executed in the above cascade configuration, the command outputs information like this:

```
PROMPT> drmsqldisplay default -remote
INSTANCE: default
T DB  OBJECT  FILE                 L-COPY-GROUP R-COPY-GROUP  PORT#  TID#  LUN#  MU#  LDEV#
P/S  SERIAL#
D UserDB  DATAFILE  O:\UserDB_Data.MDF   -           -           CL2-A  16  49  -   741     P
55014
- -     -  -                  SI01,sql01   -           GL2-B  15  109  0   813     S  55014
- -     -  -                  SI02,sql01   -           CL2-B  15  109  1   813     S  55014
- -     -  -                  SI03,sql01   -           CL2-B  15  109  2   813     S  55014
- -     -  -                 TC01,sql01   TC01,sql01#0   -      -  -  -   114     C  55555
D UserDB  DATAFILE  P:\UserDB_Data.NDF   -           -           CL2-A  16  49  -   741     P
55014
- -     -  -                  SI01,sql02   -           GL2-B  15  109  0   813     S  55014
- -     -  -                  SI02,sql02   -           CL2-B  15  109  1   813     S  55014
- -     -  -                  SI03,sql02   -           CL2-B  15  109  2   813     S  55014
- -     -  -                 TC01,sql02   TC01,sql02#0   -      -  -  -   114     C  55555
T UserDB  TRANSACT  O:\UserDB_Log.LDF   -           -           CL2-A  16  49  -   741     P
55014
- -     -  -                  SI01,sql01   -           GL2-B  15  109  0   813     S  55014
- -     -  -                  SI02,sql01   -           CL2-B  15  109  1   813     S  55014
- -     -  -                  SI03,sql01   -           CL2-B  15  109  2   813     S  55014
- -     -  -                 TC01,sql01   TC01,sql01#0   -      -  -  -   114     C  55555
PROMPT>
```

Provided below are sample scripts for the following two cases:

- Backup operation involving no local backup
- Backup operation involving local backup

For each case, a flowchart of pair operations executed within the script and a sample script are provided in Figure 7.37, Figure 7.28, Table 7.8, and Table 7.9.

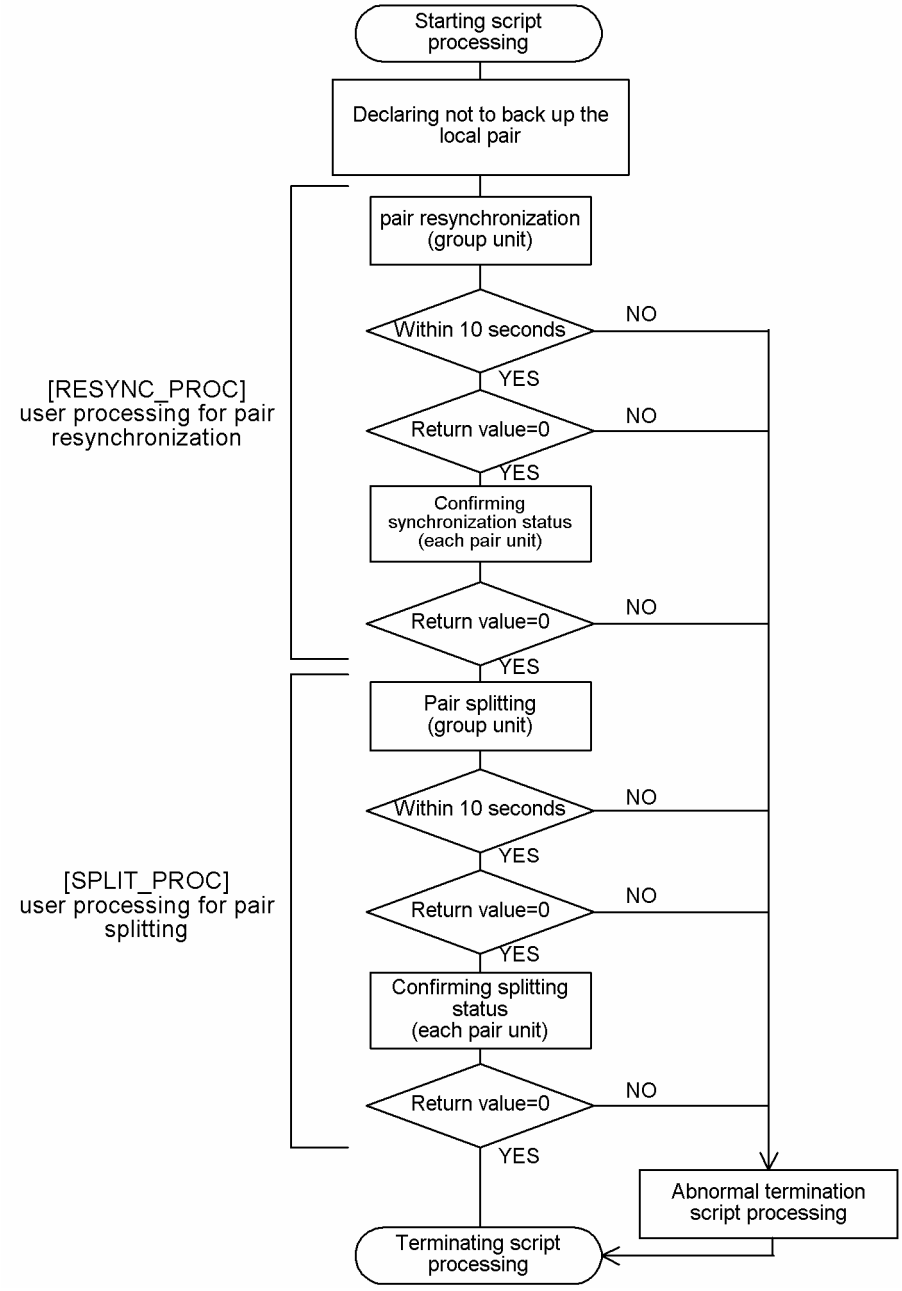


Figure 7.37 Flow of Script-Specified Operations Performed When Local Backup is not Involved

Table 7.8 Example of Writing a Sample Script When Local Backup is not Involved

Script code	Description
LOCAL_BACKUP=NO *...(1)*	(1) Specifies that the database should not be backed up to the local site secondary volume (backed up only to the remote site secondary volume).
[RESYNC_PROC] *...(2)*	(2) Beginning of user processing for pair resynchronization
# Target pair volumes: SI11 dev group	
[CMD]	
CMDLINE=C:\HORCM\etc\pairresync.exe -g TC01 -FBC 0 *...(3)*	(3) Batch resynchronization with a pair group specified
ENV=HORCC_MRCF= *...(4)*	(4) Specifies HORC environment execution.
ENV=HORCMINST=0 *...(5)*	(5) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(6)*	(6) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=10 *...(7)*	(7) If the command does not terminate within 10 seconds, the script will terminate with an error.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql01 -s pair -t 600 -FBC 0 *...(8)*	(8) Checks the sql01 pair status. Waits until the sql01 status becomes PAIR.
ENV=HORCMINST=0 *...(9)*	(9) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(10)*	(10) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(11)*	(11) Waits for command termination.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql02 -s pair -t 600 -FBC 0 *...(12)*	(12) Checks the sql02 pair status. Waits until the sql02 status becomes PAIR.
ENV=HORCMINST=0 *...(13)*	(13) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(14)*	(14) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(15)*	(15) Waits for command termination.
[SPLIT_PROC] *...(16)*	(16) User processing for pair splitting.
# Target pair volumes: SI11 dev group	
[CMD]	
CMDLINE=C:\HORCM\etc\pairsplit.exe -g TC01 -FBC 0 *...(17)*	(17) Batch splitting with a pair group specified
ENV=HORCC_MRCF= *...(18)*	(18) Specifies HORC environment execution.
ENV=HORCMINST=0*...(19)*	(19) Specifies an instance number.

Script code	Description
END_CODE=TERMINATE_NZ *...(20)*	(20) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=10 *...(21)*	(21) If the command does not terminate within 10 seconds, the script will terminate with an error.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql01 -s psus -t 600 -FBC 0 *...(22)*	(22) Checks the sql01 pair status. Waits until the sql01 status becomes PSUS.
ENV=HORCMINST=0 *...(23)*	(23) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(24)*	(24) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(25)*	(25) Waits for command termination.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql02 -s psus -t 600 -FBC 0 *...(26)*	(26) Checks the sql02 pair status. Waits until the sql02 status becomes PSUS.
ENV=HORCMINST=0 *...(27)*	(27) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(28)*	(28) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(29)*	(29) Waits for command termination.
[FINISH_PROC] *...(30)*	(30) User processing for termination
#do nothing	

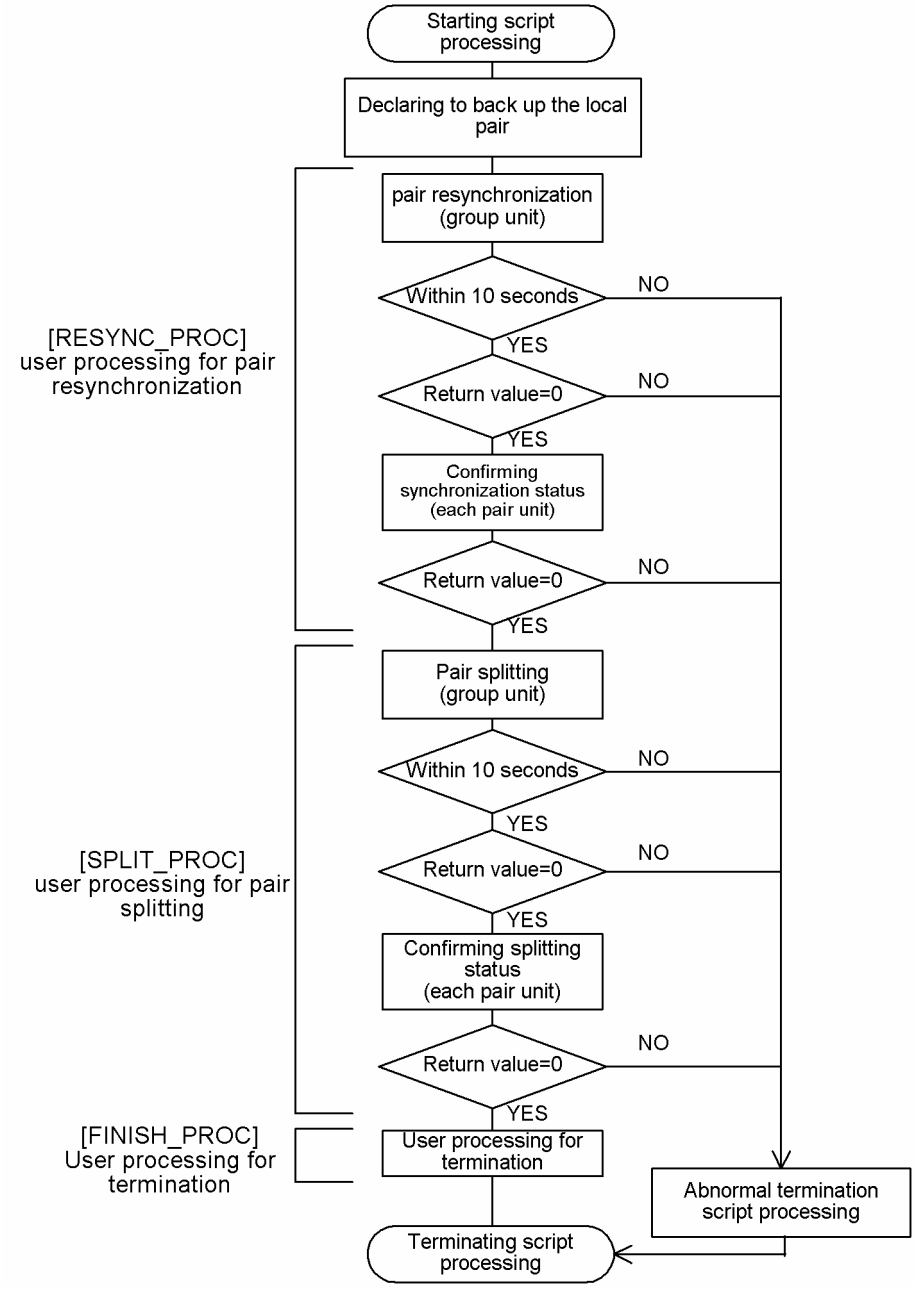


Figure 7.38 Flow of Script-Specified Operations Performed When Local Backup is Involved

Table 7.9 Example of Writing a Sample Script When Local Backup is Involved

Script code	Description
LOCAL_BACKUP=YES *...(1)*	(1) Specifies that the database should also be backed up at the local site secondary volume (backup in a multi-target configuration).
[RESYNC_PROC] *...(2)*	(2) User processing for pair resynchronization
# Target pair volumes: SI11 dev group	
[CMD]	
CMDLINE=C:\HORCM\etc\pairresync.exe -g TC01 -FBC 0 *...(3)*	(3) Batch resynchronization with a pair group specified
ENV=HORCC_MRCF= *...(4)*	(4) Specifies HORC environment execution.
ENV=HORCMINST=0 *...(5)*	(5) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(6)*	(6) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=10 *...(7)*	(7) If the command does not terminate within 10 seconds, the script will terminate with an error.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql01 -s pair -t 600 -FBC 0 *...(8)*	(8) Checks the sql01 pair status. Waits until the sql01 status becomes PAIR.
ENV=HORCMINST=0 *...(9)*	(9) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(10)*	(10) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(11)*	(11) Waits for command termination.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql02 -s pair -t 600 -FBC 0 *...(12)*	(12) Checks the sql02 pair status. Waits until the sql02 status becomes PAIR.
ENV=HORCMINST=0 *...(13)*	(13) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(14)*	(14) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(15)*	(15) Waits for command termination.
[SPLIT_PROC] *...(16)*	(16) User processing for pair splitting.
# Target pair volumes: SI11 dev group	
[CMD]	
CMDLINE=C:\HORCM\etc\pairsplit.exe -g TC01 -FBC 0 *...(17)*	(17) Batch splitting with a pair group specified
ENV=HORCC_MRCF= *...(18)*	(18) Specifies HORC environment execution.
ENV=HORCMINST=0 *...(19)*	(19) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(20)*	(20) If a non-zero return value returns, the script will terminate with an error.

Script code	Description
TIMEOUT=10 *...(21)*	(21) If the command does not terminate within 10 seconds, the script will terminate with an error.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql01 -s psus -t 600 -FBC 0 *...(22)*	(22) Checks the sql01 pair status. Waits until the sql01 status becomes PSUS.
ENV=HORCMINST=0 *...(23)*	(23) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(24)*	(24) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(25)*	(25) Waits for command termination.
[CMD]	
CMDLINE=C:\HORCM\etc\pairevtwait.exe -g TC01 -d sql02 -s psus -t 600 -FBC 0 *...(26)*	(26) Checks the sql02 pair status. Waits until the sql02 status becomes PSUS.
ENV=HORCMINST=0 *...(27)*	(27) Specifies an instance number.
END_CODE=TERMINATE_NZ *...(28)*	(28) If a non-zero return value returns, the script will terminate with an error.
TIMEOUT=0 *...(29)*	(29) Waits for command termination.
[FINISH_PROC] *...(30)*	(30) User processing for termination
[CMD]	
CMDLINE="C:\Program Files\ORIGINAL\TERM.wsf" *...(31)*	(31) Specifies shell script execution.
END_CODE=IGNORE *...(32)*	(32) You should assign IGNORE to END_CODE in the FINISH_PROC item if LOCAL_BACKUP is YES.
#script end	

7.12.3 Backing up a SQL Server Database in Cascade Configuration

This section explains the steps for backing up a SQL Server database at the remote site secondary volume (backup in a cascade configuration). The sample procedure provided here assumes that multi-target backup is not to be performed; that is to say, the database is to be backed up only at the remote site secondary volume, not at the local site's secondary volume. For details on how to perform multi-target backup, see section 7.9.1.

To back up a SQL Server database in a cascade configuration:

1. Prepare the user script file.

   This sample procedure uses the following user script file:

```
LOCAL_BACKUP=NO
[RESYNC_PROC]
[CMD]
CMDLINE=C:\HORCM\etc\pairresync.exe -g TC01 -d sql01 -FBC 0
ENV=HORCMINST=0
END_CODE=TERMINATE_NZ
TIMEOUT=0
[SPLIT_PROC]
[CMD]
CMDLINE=C:\HORCM\etc\pairsplit.exe -g TC01 -d sql01 -FBC 0
ENV=HORCMINST=0
END_CODE=TERMINATE_NZ
TIMEOUT=0
[FINISH_PROC]
```

2. Back up the SQL Server database.

```
PROMPT> drmsqlbackup default -script C:\Uscrip.txt
KAVX0001-I The drmsqlbackup command will now start.
KAVX1017-I SQL Server information is being collected now.
KAVX1207-I Backing up the database...
KAVX0210-I The user script will now start.
             Procedure section = RESYNC_PROC
KAVX0212-I A command from the user script will now execute.
           Command line = C:\HORCM\etc\pairresync.exe -g TC01 -d sql01 -FBC 0
KAVX0213-I A command from the user script has ended.
             End code = 0
KAVX0211-I Execution of the user script has finished.
KAVX0210-I The user script will now start.
             Procedure section = SPLIT_PROC
KAVX0212-I A command from the user script will now execute.
           Command line = C:\HORCM\etc\pairsplit.exe -g TC01 -d sql01 -FBC 0
KAVX0213-I A command from the user script has ended.
             End code = 0
KAVX0211-I Execution of the user script has finished.
KAVX0002-I The drmsqlbackup command will now end.
```

3. Check the backup execution results.

   To check the backup execution results, execute the `drmsqlcat` command on the database server at the local site. If you perform backup with `NO` assigned to `LOCAL_BACKUP` in the user script file, local site backup information will not be displayed.

```
PROMPT> drmsqlcat default -template
KAVX0024-E The specified backup data does not exist.
```

4. Export the backup catalog to a temporary file.

   To export the backup catalog to the temporary file `C:\tmp\EX-FILE1.drm`, execute the `drmdbexport` command. This step must be performed on the database server at the remote site.

```
PROMPT> drmdbexport 0000000001 -f C:\tmp\EX-FILE1.drm
KAVX0001-I The drmdbexport command will now start.
KAVX0002-I The drmdbexport command will now end.
```

5. Transfer the exported temporary file from the database server to the backup server.

   To transfer the exported file `C:\tmp\EX-FILE1.drm` from the database server to the backup server, execute the ftp (file transfer protocol) command on the database server at the remote site. It is assumed here that the FTP root directory is `C:\FTP_ROOT`. The temporary file is named `C:\FTP_ROOT\EX-FILE1.drm` and transferred to the backup server.

```
PROMPT> ftp backup-server-name
ftp> Username: login-name
ftp> password: password
ftp> binary
ftp> put C:\temp\EX-FILE1.drm
ftp> quit
PROMPT>
```

6. Import the temporary file, which was transferred from the database server, into the backup catalog of the backup server.

   Import the temporary file `C:\FTP_ROOT\EX-FILE1.drm`, which was transferred from the database server, into the backup catalog of the backup server. To import the file, execute the `drmdbimport` command on the backup server at the remote site. A backup ID that is not being used on the server is assigned. In this example, a backup ID to be assigned is assumed to be `0000000002`.

```
PROMPT> drmdbimport -f C:\FTP_ROOT\EX-FILE1.drm
KAVX0001-I The drmcgctl command will now start.
KAVX0418-I Backup information from the specified file was imported into the backup catalog.
           Imported backup ID = 0000000002
KAVX0002-I The drmcgctl command will now end.
PROMPT>
```

7. Check the import results.

   Confirm that the import finished successfully. To confirm this, execute the `drmsqlcat` command on the backup server at the remote site.

```
PROMPT> drmsqlcat default -template
INSTANCE: default
BACKUP-ID: 0000000002  BACKUP-MODE: ONLINE  INSTANCE: default  PROMPT> drmsqlcat default
INSTANCE: default
BACKUP-ID: 0000000002  BACKUP-MODE: ONLINE  INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 2004/02/04 13:01:35  END-TIME: 2004/02/04 13:02:33  HOSTNAME: flora4
T DB        OBJECT     FILE                                              FS   DG   DEVICE
COPY-GROUP
M UserDB1   METAFILE   O:\META_0000000001.dmp   -    -    -          -
D UserDB1   DATAFILE   O:\UserDB1_Data.MDF                            O:   -    Harddisk4
SI01,sql01
- -       -        -                                          -   -   -         SI01,sql01
T UserDB1   TRANSACT   O:\UserDB1_Log.LDF                             O:   -    Harddisk4
SI01,sql01
- -       -        -                                          -   -   -         SI01,sql01
PROMPT>
```

8. Mount the secondary volume.

   This step must be performed on the backup server at the remote site. It is assumed here that the mount point is "`E:`".

```
PROMPT> drmmount 0000000002 -mount_pt E:
KAVX0001-I The drmmount command will now start.
KAVX0400-I Mounting will now start.
           Mount point = E:
KAVX0401-I Mounting has finished.
           Mount point = E:
KAVX0002-I The drmmount command will now end.
PROMPT>
```

9. Back up the secondary volume to tape.

   To perform backup, execute the `drmmediabackup` command on the backup server at the remote site. At the same time, lock the copy group to ensure that the secondary volume where the SQL Server database has been backed up cannot be manipulated from the backup server.

```
PROMPT> drmmediabackup 0000000002
KAVX0001-I The drmmediabackup command will now start.
KAVX0402-I Backup processing will now start.
To see more details on backup progress, use the monitoring method of the backup management
product.
           Source = E:
KAVX0403-I Backup processing has finished.
           Source = E:
KAVX0040-I The backup ID has been assigned as follows:
           Backup ID = 0000000003
KAVX0002-I The drmmediabackup command will now end.
```

   When backup is performed, information pertaining to the backup operations is registered in the backup catalog, using the new backup ID `0000000003`.

10. Unmount the secondary volume.

    To unmount the secondary volume, specify the backup ID used when mounting `0000000002`. This step must be performed on the backup server at the remote site.

```
PROMPT> drmumount 0000000002
KAVX0001-I The drmumount command will now start.
KAVX0406-I Unmounting will now start.
           Mount point = E:
KAVX0407-I Unmounting has finished.
           Mount point = E:
```

11. Check the backup to tape execution results.

    Confirm that backup to tape finished successfully. To confirm this, execute the `drmtapecat` command on the backup server at the remote site.

```
PROMPT> drmtapecat 0000000003
BACKUP-ID   BACKUP-OBJECT  SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000003  MSSQL            2004/02/04 13:01:35  2004/05/04 13:01:35  MEDIA1
PROMPT>
```

7.12.4 Backing up Transaction Logs in Cascade Configuration (No Backup Catalog)

This section explains the steps for backing up transaction logs in a cascade configuration when the backup catalog does not exist.

Back up the transaction logs by executing the `drmsqllogbackup` command on the database server at the local site.

```
PROMPT> drmsqllogbackup default -no_cat
KAVX0001-I The drmsqllogbackup command will now start.
KAVX0002-I The drmsqllogbackup command will now end.
```

7.12.5 Restoring a SQL Server Database in Cascade Configuration

This section explains the steps for restoring SQL Server database data, which was backed up from the remote site secondary volume to tape, to the local site (restoration in a cascade configuration).

To restore a SQL Server database in a cascade configuration:

1. Stop the SQL Server service at the local site.
2. Unmount the secondary volume at the local site.
3. Perform takeover.

   If the local site primary volume can communicate with the remote site secondary volume (for example when the current and standby systems are intentionally switched), use the `horctakeover` command to perform takeover. This step must be performed at the remote site.

```
PROMPT> horctakeover -g TC01 -d sql01
```

4. Split the TrueCopy pair (usual operation is in the pair state).

To split a TrueCopy pair whose usual operation is in the pair state, execute the `pairsplit` command at the remote site.

```
PROMPT> pairsplit -g TC01 -d sql01 -rw
```

5. To restore data from the tape, mount the secondary volume.

   To mount the secondary volume, specify backup ID `0000000002`.

   The following assumes that the mount point for the secondary volume is `E:`. This step must be performed on the backup server at the remote site.

```
PROMPT> drmmount 0000000002 -mount_pt E:
KAVX0001-I The drmmount command will now start.
KAVX0400-I Mounting will now start.
           Mount point = E:
KAVX0401-I Mounting has finished.
           Mount point = E:
KAVX0002-I The drmmount command will now end.
PROMPT>
```

6. Restore the backup data from the tape to the secondary volume.

   To perform the restoration, execute the `drmmediarestore` command on the backup server at the remote site.

```
PROMPT> drmmediarestore 0000000002
KAVX0001-I The drmmediarestore command will now start.
KAVX0404-I Restoration will now start.
To see more details on restoration progress, use the monitoring method of the backup
management product.
           Source = E:
           Target = E:
KAVX0405-I Restoration has finished.
           Source = E:
           Target = E:
KAVX0040-I The backup ID has been assigned as follows:
           Backup ID = 0000000003
KAVX0002-I The drmmediarestore command will now end.
PROMPT>
```

   When the restoration is performed, information pertaining to the restoration operations is registered in the backup catalog at the remote site backup server, using a new backup ID `0000000003`. At this point, since the status of the secondary volume will be updated, the backup ID `0000000001`, which was used to manage the status of the secondary volume before the update, is deleted from the backup catalog on the backup server.

7. Unmount the secondary volume.

   To unmount the secondary volume, specify the backup ID that was used when mounting `0000000002`. This step must be performed on the backup server at the remote site.

```
PROMPT> drmumount 0000000002
KAVX0001-I The drmumount command will now start.
KAVX0406-I Unmounting will now start.
           Mount point = E:
KAVX0407-I Unmounting has finished.
           Mount point = E:
KAVX0002-I The drmumount command will now end.
PROMPT>
```

8. Check the execution results of the restoration from the tape to the secondary volume.

To confirm that the restoration finished successfully, execute the `drmsqlcat` command on the backup server at the remote site.

```
PROMPT> drmsqlcat default -template
INSTANCE: default
BACKUP-ID: 0000000003 BACKUP-MODE: ONLINE INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 04.02.04 13:01:35 END-TIME: 04.02.04 13:02:33 HOSTNAME: flora4
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M UserDB1   METAFILE   O:\MSSQL\0000000002_0000000001.dmp   -    -    -          -
D UserDB1   DATAFILE   O:\UserDB1_Data.MDF                          O:   -    Harddisk4
SI01,sql01
- -       -        -                                        -   -   -         SI01,sql01
T UserDB1   TRANSACT   O:\UserDB1_Log.LDF                           O:   -    Harddisk4
SI01,sql01
- -       -        -                                        -   -   -         SI01,sql01
PROMPT>
```

9. Export the backup catalog to a temporary file.

   To perform restoration from a secondary volume to a primary volume, you must copy the backup catalog that contains information pertaining to the operations performed during the restoration to the secondary volume. Copy this backup catalog to the database server on the remote site. To export the backup information to the FTP root folder, execute the `drmdbexport` command. The following assumes that the FTP root folder is `C:\FTP_ROOT`, and that the backup information is exported to `C:\FTP_ROOT\EX-FILE1.drm`.

```
PROMPT> drmdbexport 0000000003 -f C:\FTP_ROOT\EX-FILE1.drm
KAVX0001-I The drmdbexport command will now start.
KAVX0002-I The drmdbexport command will now end.
```

10. From the database server, obtain the temporary file that was exported on the backup server.

    Obtain the temporary file `C:\FTP_ROOT\EX-FILE1.drm`, which was exported on the backup server, from the database server at the remote site. Use the `ftp` (file transfer protocol) command to obtain the temporary file `C:\FTP_ROOT\EX-FILE1.drm` (created in step 9) to the database server. The temporary file will be stored in the current directory. The following assumes that the current directory is `C:\`.

```
PROMPT> ftp database-server-name
ftp> Username: (logon-name)
ftp> password: (password)
ftp> binary
ftp> get EX-FILE1.drm
ftp> quit
PROMPT>
```

11. Import the temporary file into the backup catalog on the database server.

    To import the temporary file `C:\EX-FILE1.drm` (transferred from the backup server in step 10) into the backup catalog on the database server, execute the `drmdbimport` command on the database server of the remote site.

```
PROMPT> drmdbimport -f C:\EX-FILE1.drm
KAVX0001-I The drmdbimport command will now start.
KAVX0418-I Backup information from the specified file was imported into the backup catalog.
           Imported backup ID = 0000000002
KAVX0002-I The drmdbimport command will now end.
PROMPT>
```

At a restoration, backup information is registered in the backup catalog of the database server under the new backup ID `0000000002`. Also, since the fact that a restoration was performed from tape to the secondary volume is reflected in the backup catalog, the backup ID `0000000001`, which was used to manage the status of the secondary volume before the update, is deleted from the backup catalog on the database server.

12. Check the backup catalog for the copy group to be used for restoration.

    Execute the `drmsqlcat` command on the database server at the remote site.

```
PROMPT> drmsqlcat default -template
INSTANCE: default
BACKUP-ID: 0000000002 BACKUP-MODE: ONLINE INSTANCE: default  ORIGINAL-ID: 0000000001
START-TIME: 04.02.04 13:01:35 END-TIME: 04.02.04 13:02:33 HOSTNAME: flora4
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M UserDB1   METAFILE   O:\MSSQL\0000000002_0000000001.dmp   -    -    -          -
D UserDB1   DATAFILE   O:\UserDB1_Data.MDF                          O:   -    Harddisk4
SI12,sql01
- -       -        -                                        -    -    -         SI12,sql01
T UserDB1   TRANSACT   O:\UserDB1_Log.LDF                           O:   -    Harddisk4
SI12,sql01
- -       -        -                                        -    -    -         SI12,sql01
```

13. On the remote site, mount the primary volume.

14. On the remote site, start the SQL Server service.

15. Restore the data on the secondary volume to the primary volume.

    To perform the restoration, resynchronize the primary and secondary volumes. To perform restoration, execute the `drmsqlrestore` command on the database server at the remote site. If you want to execute the command with the template catalog backup ID specified, specify the `-template` option.

```
PROMPT> drmsqlrestore 0000000003 -resync -template
KAVX0001-I The drmsqlrestore command will now start.
KAVX1316-I Performing restore pre-processing...
KAVX1000-I Logged on to the SQL Server system.
KAVX1300-I Restore processing will now start.
KAVX1000-I Logged on to the SQL Server system.
KAVX1301-I Restore processing has finished.
KAVX1318-I Performing restore post-processing..
KAVX1001-I Logged off from the SQL Server system.
KAVX0002-I The drmsqlrestore command will now end.
```

16. Apply the transaction log file.

    Apply the transaction log backup file to the database to recover the SQL Server database. Execute the `drmsqlrecovertool` command at the remote site database server.

```
PROMPT> drmsqlrecovertool default
```

The drmsqlrecovertool dialog box appears.

17. Recover the SQL Server database.

    Recover the SQL Server database by following the instructions displayed on the dialog box.

    If the past transaction log backup files were not deleted appropriately from the local site, transaction log backup files that are unnecessary for recovery may also exist at the local site. If this occurs, you must select the files required for recovery by referring to the creation date and time and the name of each file before starting recovery.

18. Stop the SQL Server service at the remote site.

19. On the remote site, unmount the primary volume.

20. On the remote site, split the ShadowImage pair and change the status of the TrueCopy pair to `PAIR`.

```
PROMPT> set HORCC_MRCF=1
PROMPT> pairsplit -g SI11 -d sql01
PROMPT> set HORCC_MRCF=
```

    If the primary volume is on the remote site:

```
PROMPT> pairresync -g TC01 -d sql01
```

    If the primary volume is on the local site:

```
PROMPT> pairsplit -g TC01 -d sql01 -S
PROMPT> paircreate -g TC01 -d sql01 -vl -f never
```

21. Perform takeover at the local site to change the local site to the primary volume.

    Execute the `horctakeover` command at the local site.

```
PROMPT> horctakeover -g TC01 -d sql01
```

22. At the local site, mount the primary volume and start the SQL Server service.

When performing this step, note the following point.

A transaction log backup file is associated with the backup catalog that was created for the last complete backup of the SQL Server database. However, if this last backup was performed with `NO` assigned to `LOCAL_BACKUP` in the user script file, no backup ID was assigned. Therefore, when the log backup information is displayed by executing `drmsqllogbackup` with the `-v` option specified, the backup catalog and the transaction log file do not correspond, and the information cannot be used for a recovery. In this situation, use the `drmsqlrecovertool` dialog box to select transaction log backup files that are needed to perform recovery operations and perform the recovery operations.

7.13 Using a Backup Management Product That Cannot Link with Protection Manager

When using a backup management product that is unable to link with Protection Manager, carry out the following procedure to back up a SQL Server database to tape, or restore a SQL Server database from tape. For details on operations from step 1 to step 6 in the backup procedure, see section 7.3.3.

For details on the restoration procedure, see section 7.3.4.

To back up a SQL Server database to tape (when using a backup management product that is unable to link with Protection Manager):

1. When usual operation is in the split state, execute `EX_DRM_CACHE_PURGE` to clear the secondary volume cache.
2. When usual operation is in the split state, execute `EX_DRM_RESYNC` to resynchronize the copy group.
3. Execute `EX_DRM_SQL_BACKUP` to back up the SQL Server database into the secondary volume.
4. Execute `EX_DRM_DB_EXPORT` to export the backup information into the temporary file.
5. Execute `EX_DRM_SQLFILE_PACK` to back up the SQL Server metafile created at the backup to the temporary folder.
6. Execute `EX_DRM_FTP_PUT` to transfer the temporary files to the backup servers.
7. Execute `EX_DRM_DB_IMPORT` to import the temporary files transferred from the database server, into the backup catalog of the backup server.
8. Execute `EX_DRM_SQLFILE_EXTRACT` to deploy, on the backup server, the SQL Server metafile transferred from the database server.
9. Check the backup ID.

   To check the backup ID, open the backup ID record file (*`operation-ID`*`.bid`) in the following directory on the backup server:

   `FTP_HOME_DIR-value``\`*`FTP_SUB_DIR-value`*`\`*`operation-ID`*`\BK\`
   `operation-ID``.bid`
10. Check the storage location of the data files and log files.

    To check the location, execute the `drmmount` command using the backup ID checked in step 9 as the argument.

    The command results show the mounted drive name. The mounted drive stores data files and transaction log files for the SQL Server database. If necessary, execute the `drmsqlcat` command to check the contents of the file to be backed up.

11. Check the storage location of the metafile.

    If you registered the VDI metafile storage directory when executing the `drmsqlinit` command, the metafile (*`original-ID_database-ID`*`.dmp`) is stored in the following directory:

    `FTP_HOME_DIR-value``\`*`FTP_SUB_DIR-value`*`\`*`operation-ID`*`\AP\`*`original-ID_database-ID`*`.dmp`

    If you did not register the VDI metafile storage directory when executing the `drmsqlinit` command, the metafile is stored in the drive containing data files and transaction log files.

12. Use a backup management product to back up the checked data files, transaction log files, and metafile onto tape.

13. Execute the `drmumount` command to unmount the mount point mounted in step 10.

To restore a SQL Server database from a tape (where a backup management product that is unable to link with Protection Manager is used):

1. Execute the `drmsqlcat` command to identify the backup ID for the backup catalog to be restored.

2. Execute the `drmmount` command, and then restore the backup from tape to the storage location checked in backup steps 10 and 11.

3. Execute `EX_DRM_BACKUPID_SET` with the backup ID specified to create a backup ID record file.

4. Carry out the procedures following step 3 in section 7.3.3.2 to restore the backup from the secondary volume to the primary volume.

7.14 Backing up and Restoring a SQL Server Database Using a File Share

Usually, to transmit backup information between the database server and backup server, you can use `EX_DRM_FTP_PUT` or `EX_DRM_FTP_GET` to perform FTP transfer. Instead of FTP, however, file sharing can also be used to copy backup information between servers.

The following sections describe examples of backing up or restoring a SQL Server database by using a file share.

The examples in this section assume the system configuration shown in Figure 7.39.

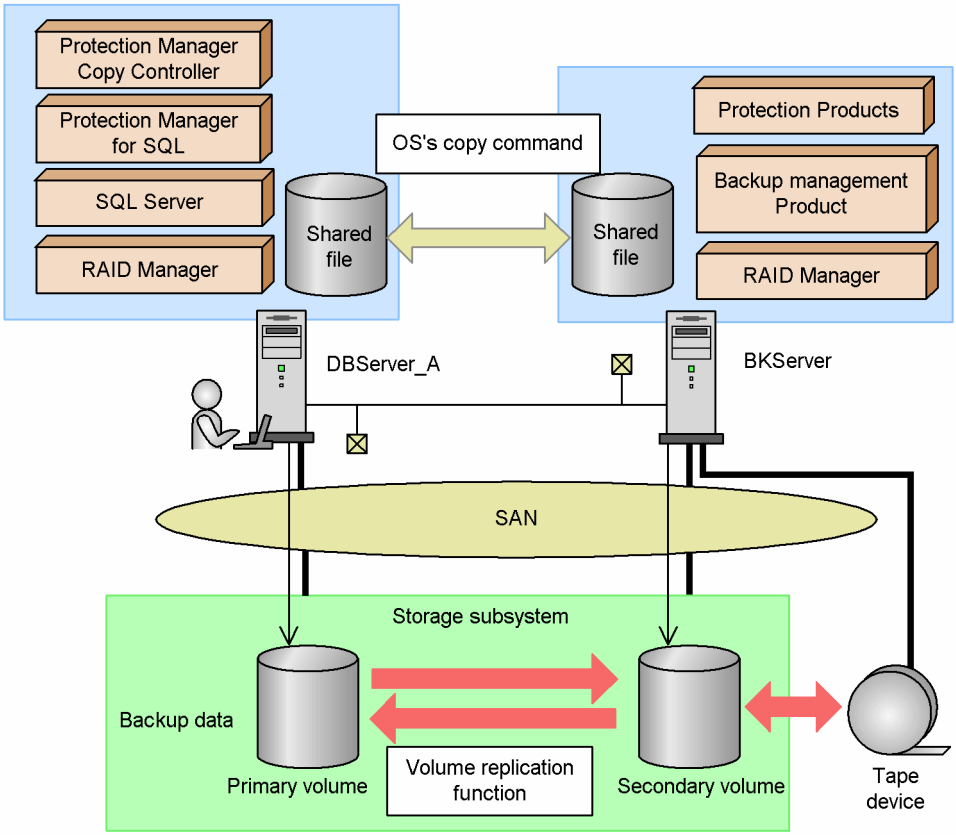


Figure 7.39 System Configuration Using File Sharing to Back up and Restore a SQL Server Database

7.14.1 Preparing to Back up or Restore Using a File Share

This section describes the preparations for backing up or restoring a SQL Server database by using file share.

To prepare for backing up or restoring a SQL Server database by using a file share:

1. Check the temporary directory for extended commands that was created automatically by the check tool for operation definition files.

   The temporary directory for extended commands is created in the following directory:

   In the database server:

   *script_work-at-the-same-level-as-the-dictionary-map-file-storage-directory**operation-ID*\DB

   Example:

   C:\Program Files\HITACHI\DRM\script_work\Operation_A\DB

   In the backup server:

   *directory-specified-for-FTP_HOME_DIR**directory-specified-for-FTP_SUB_DIR**operation-ID*\BK

   Example:

   C:\FTP_ROOT\script\Operation_A\BK

   For details about the temporary directory for extended commands, see section 4.15.9.

2. Use Explorer to share the temporary directory for extended commands that is on the backup server.

   In this case, C:\FTP_ROOT\script\Operation_A\BK is shared.

3. Use Explorer on the database server to allocate the temporary directory for extended commands as a network drive.

   In this case, C:\FTP_ROOT\script\Operation_A\BK is allocated to network drive X:.

4. On the backup server, set the permissions for the automatically generated temporary directory for extended commands, so that users executing extended commands from the database server can read and write to the directory.

7.14.2 Backing up a SQL Server Database Using a File Share

This section describes the procedure for backing up a SQL server database by using a file share. For details about operations for steps 1 through 4, or step 8 and beyond, see section 7.3.3.

The following conditions are assumed:

1. When usual operation is in the split state, execute EX_DRM_CACHE_PURGE to clear the secondary volume cache.
2. When usual operation is in the split state, execute EX_DRM_RESYNC to resynchronize the copy group.

3. Execute EX_DRM_SQL_BACKUP to back up the SQL Server database into the secondary volume.
4. Execute EX_DRM_DB_EXPORT to export the backup information into the temporary file.
5. Execute EX_DRM_SQLFILE_PACK to back up the SQL Server metafile created at the backup to the temporary folder.
6. Delete unnecessary metafiles (those created during the previous backup) stored in the temporary directory for extended commands that is on the backup server.

```
DBServer > del X:\*.dmp
```

7. Copy the temporary files to the temporary directory for extended commands that is on the backup server.

```
DBServer > copy /y C:\Program Files\HITACHI\DRM\script_work\Operation_A\DB X:\
```

8. Execute EX_DRM_DB_IMPORT to import the temporary files copied from the database server, into the backup catalog of the backup server.
9. Execute EX_DRM_SQLFILE_EXTRACT to deploy, on the backup server, the metafile transferred from the database server.
10. When usual operation is in the pair state, execute EX_DRM_CACHE_PURGE to clear the cache of the secondary volume.
11. When usual operation is in the pair state, execute EX_DRM_RESYNC to resynchronize the copy group.

7.14.3 Example of Restoring a Database Using a File Share

This section explains how to use file sharing to restore a SQL Server database to the backup server. For details about the operations for steps 1 through 3, or step 7 and beyond, see section 7.3.4.

To restore a SQL Server database on the backup server by using a file share:

1. Execute the drmtapecat command to check the backup ID of the backup data.
2. Execute EX_DRM_TAPE_RESTORE to restore the backed up data from tape to the secondary volume.
3. Execute EX_DRM_DB_EXPORT to export the backup information to a temporary file.
4. Execute EX_DRM_SQLFILE_PACK to back up the SQL Server metafile created at the restoration to the temporary folder.
5. Delete unnecessary metafiles (those created during the previous backup) stored in the temporary directory for extended commands that is on the database server.

```
DBServer > del C:\Program Files\HITACHI\DRM\script_work\Operation_A\DB\*.dmp
```

6. Copy the temporary files to the temporary directory for extended commands that is on the database server.

```
DBServer > copy /y X:\*.dmp C:\Program Files\HITACHI\DRM\script_work\Operation_A\DB
```

7. Execute `EX_DRM_DB_IMPORT` to import the temporary files transferred from the backup server, into the backup catalog of the database server.
8. Execute `EX_DRM_SQLFILE_EXTRACT` to deploy, on the database server, the SQL Server metafile transferred from the backup server.
9. Execute `EX_DRM_SQL_RESTORE` to restore the secondary volume data to the primary volume.

7.15 Restoring Data to a SQL Server Instance Different from the Backup Source

If you are unable to specify the same SQL Server instance name for the restoration destination database as the one specified as the backup source, you can restore (recover) the data to a SQL Server instance other than the backup source.

7.15.1 Process Flow for Restoring Data to a Different SQL Server Instance

The figure below shows the system configuration for restoring data to a different SQL Server instance.

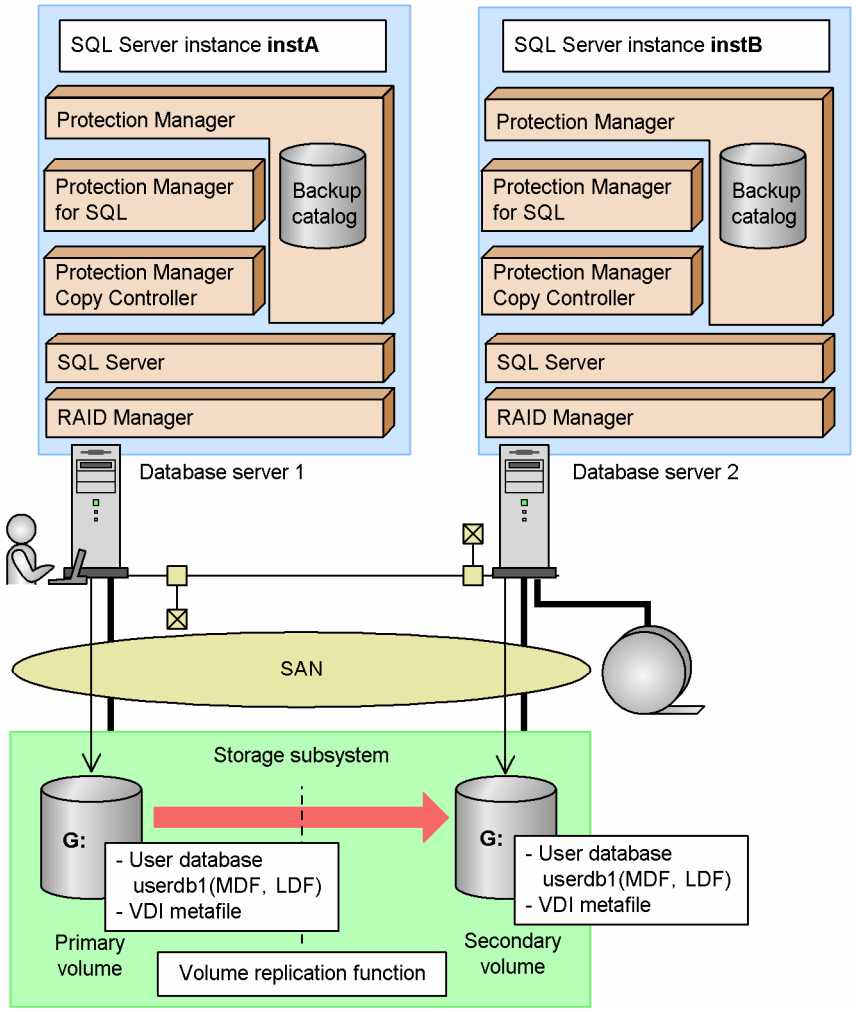


Figure 7.40 System Configuration for Restoring Data to a SQL Server Instance Different from the Backup Source

For example, if only the user database backed up from the SQL Server instance `instA` of database server 1 is restored and recovered to SQL Server instance `instB` of database server 2, the user database from database server 2 becomes available.

System databases such as `master`, `model`, and `msdb` cannot be restored to a different SQL Server instance.

This example assumes the following conditions:

- The same version of SQL Server is installed on both database server 1 and database server 2.
- The database file path configuration is the same both on database server 1 and database server 2.
- The VDI metafile storage directory is not specified by the `drmsqlinit` command on both database server 1 and database server 2.

7.15.2 Operation Flow

A flow of operation for restoring data to a SQL Server instance other than the backup source is shown below.

7.15.2.1 Preparations for Restoration (Recovery)

Make sure that SQL Server instance `instB` has been started on database server 2.

Additionally, execute the `drmsqlinit` command on database server 2 to create a database configuration definition file. In this case, do not specify the VDI metafile storage directory.

7.15.2.2 Operations for Restoration (Recovery)

To check the backup ID of the backup data before performing restoration, specify the name of the backed up SQL Server instance name `isntA` as an argument of the `drmsqlcat` command.

At the time of restoration, the database is restored to SQL Server instance `instB` on database server 2. Specify the following options for the `drmsqlrestore` command.

- `-instance`

  Since you want the database to be restored to a SQL Server instance other than the backup source, specify the restoration destination instance name (in this case, `instB`) using the `-instance` option.

- `-no_resync`

  If the volume does not need to be resynchronized, specify the `-no_resync` option.

- `-nochk_host`

  When restoring data to a server other than the backup source, specify the `-nochk_host` option, which specifies not to perform a host name integrity check for the backup catalog.

At the time of recovery, SQL Server instance `instB` is recovered on database server 2. Specify `instB` as an argument of the `drmsqlrecover` command or `drmsqlrecovertool` command.

7.15.3 Recovery Operation Examples

Examples of operations for performing restoration (recovery) under the following three conditions are described below.

- Example of restoring a SQL Server database from tape and recovering on the secondary volume:

  This example is applicable to the operations described in section 7.8.

- Example of restoring (recovering) a SQL Server database on the log distribution destination:

  This example is applicable to the operations described in sections 7.11.

- Example of restoring (recovering) a SQL Server database on the primary volume:

  This example is applicable to the operations described in sections 7.10 and 7.12.

Each operation example assumes that the SQL Server instance for backup is `instA` and the SQL Server instance for restoration (recovery) is `instB`.

This section describes how to specify the commands and options used for each operation. For details on the examples and procedures for executing these commands, see the *HiCommand Protection Manager Command Reference* manual.

7.15.3.1 Restoring a SQL Server Database from Tape and Recovering on the Secondary Volume

An example of restoring a SQL Server database from tape and recovering on the secondary volume is described below.

To restore (recover) backup data from tape to the secondary volume:

1. Execute `EX_DRM_TAPE_RESTORE` to restore the SQL Server database from tape to the secondary volume.
2. Execute the `drmsqlcat` command with `instA` specified to check the backup ID.
3. Execute the `drmsqlrestore` command with `instB` specified as the `-instance` argument to restore the SQL Server database on the secondary volume.
4. Execute the `drmsqlrecover` command or `drmsqlrecovertool` command with `instB` specified to recover the SQL Server database.

7.15.3.2 Restoring (Recovering) a SQL Server Database on the Log Distribution Destination

An example of restoring (recovering) a SQL Server database on the secondary volume is described below.

To recover on the secondary volume:

1. If the backup catalog was not imported at the time of backup, execute the `drmdbimport` command to import the backup catalog.
2. Execute the `drmsqlcat` command with `instA` specified to check the backup ID.
3. Execute the `drmsqlrestore` command with `instB` specified as the `-instance` argument to restore the SQL Server database on the secondary volume.
4. Execute the `drmsqlrecover` command or `drmsqlrecovertool` command with `instB` specified to recover the SQL Server database.

7.15.3.3 Restoring (Recovering) a SQL Server Database on the Primary Volume

An example of restoring (recovering) a SQL Server database on the primary volume is described below.

To restore (recover) data on the primary volume:

1. If there is no data to restore (recover) on the secondary volume, restore the SQL Server database to the secondary volume by conventional procedures.
2. Execute the `drmsqlrestore` command with `instB` specified as the `-instance` argument to restore the SQL Server database from the secondary volume to the primary volume.
3. Execute the `drmsqlrecover` command or `drmsqlrecovertool` command with `instB` specified to recover the SQL Server database.

7.16 Deleting Backup Data

If the backup data in the secondary volume is no longer necessary, you may delete it. Backup data is deleted after being backed up from the secondary volume to tape, or restored from the secondary volume to the primary volume.

To delete the backup data, execute the `drmresync` command, specifying the backup ID of the data to be deleted:

```
PROMPT> drmresync -backup_id backup-ID
KAVX0001-I The drmresync command will now start.
KAVX0002-I The drmresync command will now end.
PROMPT>
```

When you execute this command, the primary volume and the secondary volume are resynchronized and mirrored. Now the backup information of the specified backup ID is deleted from the backup catalog.

7.17 Making a SQL Server Database Backed Up to a Secondary Volume Available

By backing up a SQL Server database to a secondary volume and then performing restoration and recovery on a server connected to the secondary volume, the database can be made available for use on a server(SQL Server instance) other than the backup source (the server on which the backup command is performed). This operation can be performed using a single backup command that specifies a user script.

Note: This can be performed using the same procedure as explained in 7.15, except that with the procedure in *Section 7.15*, the backup, restore, and recovery commands must be performed in sequence.

This section provides an example using the following operations:

- The backup data of the SQL Server database is obtained in two generations. One generation is used for such operations as data mining on the server connected to the secondary volume, and the other generation is used in the normal manner as a backup in the event of an error.
- The restore and recovery target of this operation can only be a user database. System databases (such as master, model, and msdb) cannot be restored to a SQL Server instance that is different from the backup source.

The example is based on the system configuration shown in the figure below.

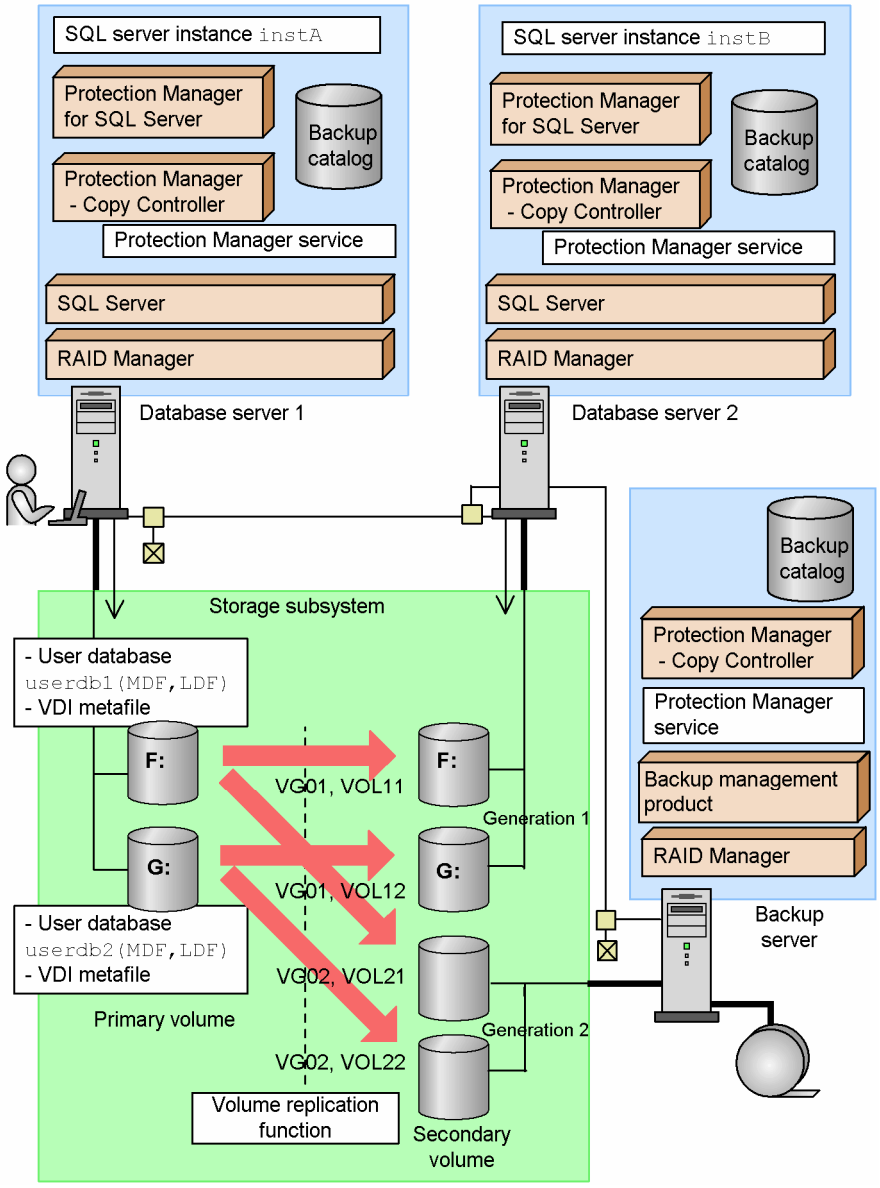


Figure 7.41 Restoring to a SQL Server Instance that is Different from the Backup Source (Using a User Script)

The following system prerequisites apply to this example:

Server configuration

- The same version of SQL Server version is installed in Database Server 1 and Database Server 2.
- The database file path configurations are the same in Database Server 1 and Database Server 2.
- The `drmsqlinit` command is executed on Database Server 2 to create the database configuration definition file.
- The VDI metafile storage directory is not specified in the `drmsqlinit` command for either Database Server 1 or Database Server 2.

Status of the secondary volume used for Generation 1 (data mining)

- The secondary volume is connected to Database Server 2.
- The drive letter of the secondary volume's mount point is the same as that of the corresponding primary volume. (F or G).

Database and copy group configuration

- The databases are `UserDB1` and `UserDB2`, which are composed of data files and log files.

  `UserDB1` (`F:\userdb1_Data.MDF` and `F:\userdb1_Log.LDF`)

  `UserDB2` (`G:\userdb2_Data.MDF` and `G:\userdb2_Log.LDF`)
- The copy groups are composed of the following two generations:

  Generation 1 (`VG01`, `VOL11` and `VG01`, `VOL12`): Used for data mining

  Generation 2 (`VG02`, `VOL21` and `VG02`, `VOL22`): Used for backup

7.17.1 User Script Example

The following two user scripts are used in this example:

7.17.1.1 script1.txt (used for data mining)

```
LOCAL_BACKUP=YES

#Pre-backup user script section
[PRE_PROC]
#The Protection manager service will be started.
[CMD]
CMDLINE=C:\user\detach_databases.bat
END_CODE=TERMINATE_NZ
TIMEOUT=600
LOCATION=REMOTE
PARENT_STAT=NORMAL

#Post-backup user script section
[POST_PROC]
#Normal case
[CMD]
CMDLINE=C:\user\remote_sqlrestore.bat
END_CODE=1
TIMEOUT=600
LOCATION=REMOTE
PARENT_STAT=NORMAL
```

`[PRE_PROC]`: Executes the `detach_databases.bat` batch file. The contents of `detach_databases.bat` detach the database (created in the previous operation) on the secondary volume and unmount the secondary volume.

- `detach_databases.bat`

```
@echo off
@osql –E –S DBServer2\instB –i "C:\user\detach_userdbs.sql"
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
@drmumount –copy_group VG01,VOL11
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
```

```
@drmumount –copy_group VG01,VOL12
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
GOTO END
:ABEND
exit 1
:END
exit 0
```

[POST_PROC]: Executes the remote_sqlrestore.bat batch file. The contents of remote_sqlrestore.bat mount the secondary volume and execute the restore command with the -no_resync specification and the recovery command.

- remote_sqlrestore.bat

```
@echo off
@drmmount –copy_group VG01,VOL11 –mount_pt F:
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
@drmmount –copy_group VG01,VOL12 –mount_pt G:
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
@drmsqlrestore %DRMENV_R_BACKUPID% -no_resync –nochk_host
–instance instB
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
@drmsqlrecover instB
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
GOTO END
:ABEND
exit 1
:END
exit 0
```

7.17.1.2 script2.txt (used for backup)

```
LOCAL_BACKUP=YES

#Post-backup user script section
[POST_PROC]
#Normal case
[CMD]
CMDLINE=C:\user\remote_tapebackup.bat
END_CODE=1
TIMEOUT=NOWAIT
LOCATION=REMOTE
PARENT_STAT=NORMAL
```

[PRE_PROC]: None

[POST_PROC]: Executes the remote_tapebackup.bat batch file. The contents of remote_tapebackup.bat use the NetBackup command to back up the database on the secondary volume to tape and unmount the secondary volume.

- remote_tapebackup.bat

```
@echo off
@bpbackup -k "%DRMENV_COMMENT%" -p ptm_plc -s ptm_schd -i -S BackupServer -w -t
0 -L C:\temp\nbu.log
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
@drmumount %DRMENV_R_BACKUPID%
if NOT "%ERRORLEVEL%"=="0" GOTO ABEND
GOTO END
:ABEND
exit 1
:END
```

```
exit 0
```

7.17.2 Data Mining and Backup Examples

This section shows the examples used for data mining and backup.

- Execute the `drmsqlbackup` command on the DBServer1 server to make a backup (generation 1) on the secondary volume on the DBServer2 server. The data on the secondary volume is restored and recovered on the SQL Server instance `instB` on the DBServer2 server used for data mining.

```
PROMPT> drmsqlbackup instA -rc local_0 -script C:\user\script1.txt -s DBServer2 -auto_import
```

- Execute the `drmsqlbackup` command on the DBServer1 server to make a backup (generation 2) on the secondary volume on the BKServer server for use as a normal backup in case of an error. The data on the secondary volume is backed up to a tape by NetBackup.

```
PROMPT> drmsqlbackup instA -comment "daily backup" -rc local_1 -script C:\user\script2.txt -s
BKServer -auto_import -auto_mount O:
```

7.18 Backing Up and Restoring When Using the SQL Server Replication Configuration

This section explains the system configuration, requirements, and procedure for performing database backup and database restoration and recovery in a configuration that uses the SQL Server replication function.

When the SQL Server replication function is used, the following requirements must be satisfied:

- Of the three replication types (snapshot, transaction, and merge), use `transaction replication`.
- The name of the distribution database is `distribution`, and only one distribution database is created.
- Set the '`sync with backup`' option for both the publication database and the distribution database. For details about how to set this option, see section 7.18.3.

7.18.1 SQL Server System Configuration

The example is based on the system configuration shown in the figure below.

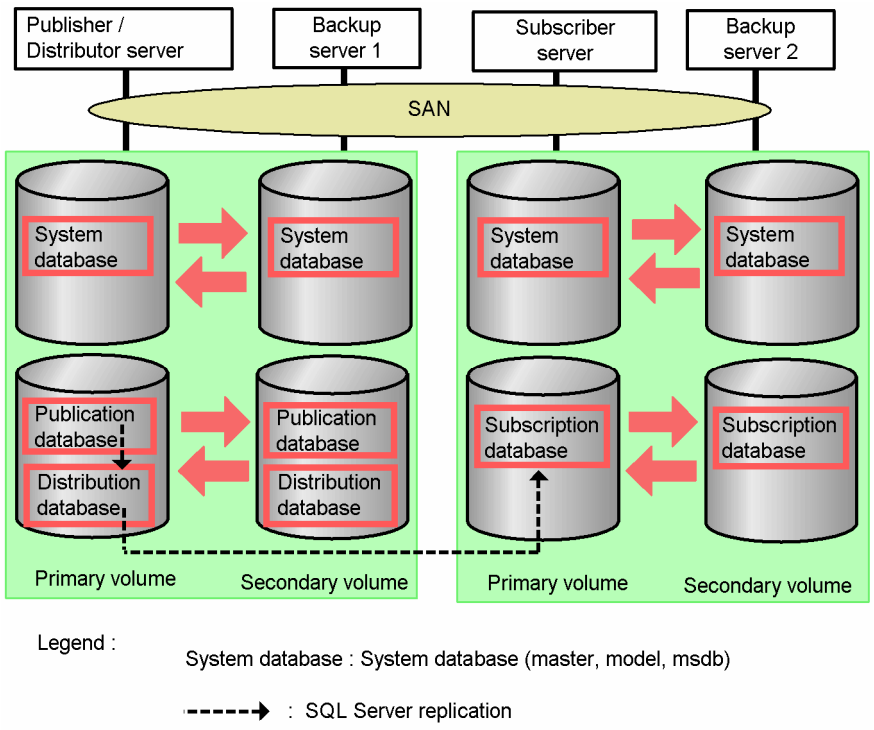


Figure 7.42 Configuration for Using the SQL Server Replication Function

7.18.1.1 Requirements for the Publication and the Distribution Databases

The publication database and the distribution database used in the processing of a single replication must be allocated to satisfy the following requirements:

- Because backups of these two databases must be made at the same time, they must be allocated to the same SQL Server instance.
- Because these two databases must be restored separately from the system databases (`master`, `model`, and `msdb`), they must be allocated to a different volume from the system databases.

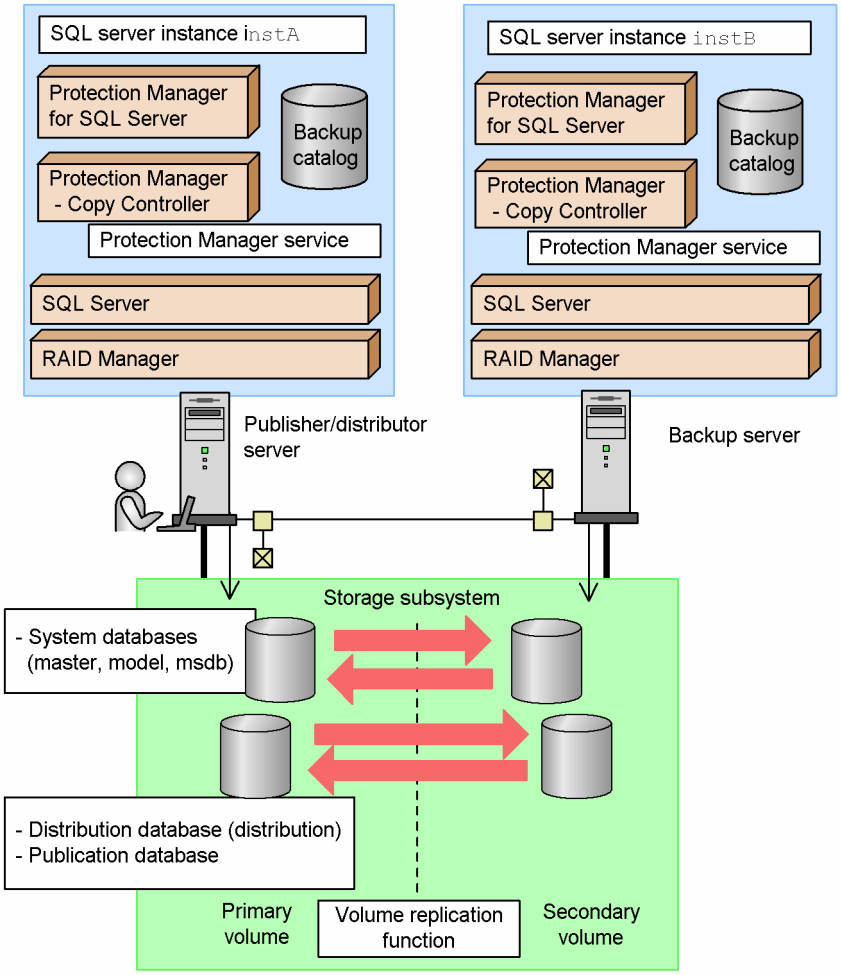


Figure 7.43 Configuration of Backup Server for the Publisher/Distributor Server

7.18.1.2 Requirements for the Subscription Database

Because the subscription database must be restored separately from the system databases (`master`, `model`, and `msdb`), it must be allocated to a different volume from the system databases.

7.18.2 Example When Using SQL Server Replication

The example uses the following database names:

Publication database: `PubUserDB1`

Subscription database: `SubUserDB1`

Distribution database: `distribution`

7.18.2.1 Back Up the Databases

- Back up the publication database

Note: When the publication database is backed up, the distribution database must be backed up at the same time.

On the publisher/distributor server, execute the `drmsqlbackup` command with the publication database and the distribution database specified in the `-target` option or `-f` option.

```
PROMPT > drmsqlbackup instA -target PubUserDB1,distribution
PROMPT >
```

- Back up the subscription database

  Execute the `drmsqlbackup` command on the subscriber server.

```
PROMPT > drmsqlbackup instC
PROMPT >
```

- Back up the databases, including the system databases

  Execute the `drmsqlbackup` command with the `-system` option specified in the publisher/distributor server or the subscriber server.

```
PROMPT > drmsqlbackup instA -system
PROMPT >
```

7.18.2.2 Back Up the Transaction Logs of the Publication Database

Execute the `drmsqllogbackup` command on the publisher/distributor server.

```
PROMPT > drmsqllogbackup instA
PROMPT >
```

7.18.2.3 Prepare for Database Restoration

To restore a database, the following conditions must be satisfied beforehand:

- When the publication database is restored, the distribution database must be restored at the same time.

- When the publication database is restored, this database must be online or have been deleted.
- When the system database is restored, the publication database must be online.

The following preparations are required to restore the databases. However, if only the subscription database is to be restored, perform only step 2 (subscription synchronous stop).

1. Stop the log reader agent.
2. Stop the subscription agent.

   A subscription synchronous stop is performed. If the agent does not stop, the detach processing will fail and an error message will be output.

7.18.2.4 Restore the Databases

- Restore the publication database

  Specify the backup ID used when the publication database and the distribution database were backed up, and execute the `drmsqlrestore` command on the publisher/distributor server.

```
PROMPT > drmsqlrestore 0000000001 -resync
PROMPT >
```

  If you are specifying the backup ID used when backing up the databases, including the system databases, specify the publication database and the distribution database in the `-target` option or `-f` option and execute the `drmsqlrestore` command.

```
PROMPT > drmsqlrestore 0000000005 -resync -target PubUserDB1,distribution
PROMPT >
```

- Restore the publication database and the system databases

  The following procedure must be followed to restore the publication and system databases on the publisher/distributor server.

  a. Restore the system databases

```
PROMPT > drmsqlrestore 0000000005 -resync -target master,model,msdb
PROMPT >
```

  b. Stop the log reader agent.

  c. Stop the subscription agent.

     A subscription synchronous stop is performed.

  Note: If the log reader agent start option is set to `Automatically startup during SQL Server agent startup`, the log reader agent will be started after the `master`, `model`, and `msdb` databases are restored, and an error will occur when the publication database and distribution database are restored.

  d. Restore the publication database and distribution database.

```
PROMPT > drmsqlrestore 0000000005 -resync -target PubUserDB1,distribution
PROMPT >
```

- Restore only the system databases

Specify the backup ID used when the databases, including the system databases, were backed up, and execute the `drmsqlrestore` command on the publisher/distributor server or the subscriber server.

```
PROMPT > drmsqlrestore 0000000005 -resync -target master,model,msdb
PROMPT >
```

- Restore the subscription database

  On the subscriber server, execute the `drmsqlrestore` command.

```
PROMPT > drmsqlrestore 0000000002 -resync -target SubUserDB1
PROMPT >
```

7.18.2.5 Perform Database Recovery

Use the `drmsqlrecover` command or the `drmsqlrecovertool` command to perform a recovery (roll forward) by using the normal procedure.

7.18.2.6 Prepare to Restart Operations

The following must be prepared in order to restart SQL Server replication operations:

The server that performs these operations will depend on the subscription format, as follows:

Push subscription: Publisher/distributor server

Pull subscription: Subscriber server

1. Start the log reader agent.
2. Start the subscription agent.

   Synchronous subscription is started.
3. Perform subscription re-initialization or deletion and re-creation.

7.18.3 Setting and Checking the sync with backup Option

When the SQL Server replication function is used, the '`sync with backup`' option must be set for the publication database and the distribution database.

When backup is performed on the publication database transaction logs with the '`sync with backup`' option set, the backup is forwarded to the distribution database. Also, no publication database logs will be dropped until all transactions are forwarded to the distribution database.

7.18.3.1 Setting the sync with backup Option

When creating the replication environment, execute the following stored procedure on the production database and distribution database. For details about how to execute stored procedures, see *SQL Server Books Online*.

```
sp_replicationdboption 'database-name', 'sync with backup', 'true'
```

For the *database-name* argument, specify the name of the publication database or distribution database.

7.18.3.2 Checking the sync with backup Option

To ensure that the option has been set, execute the following SQL statement:

```
select databasepropertyex('database-name', 'IsSyncWithBackup')
```

For the *database-name* in the SQL statement, specify the name of the publication database or distribution database.

When this SQL statement is executed, one of the following values is returned:

If the '`sync with backup`' option has been set: 1

If the '`sync with backup`' option has not been set: 0

Chapter 8 Using Protection Manager With Exchange Databases

This chapter explains the procedures for running Protection Manager when backing up storage groups. For details on setting up commands, see the *HiCommand Protection Manager Command Reference* manual.

The term *storage group* refers to a database management unit created on Exchange Server. Note that this definition differs from the term storage group used in other HiCommand products.

- Operations for Backing up and Restoring a Storage Group (see section 8.1)
- Backing up and Restoring Data Between Volumes (see section 8.2)
- Backing up and Restoring Using a Tape Device (see section 8.3)
- Backing Up a Storage Group by Using a User Script (see section 8.4)
- Restoring by Using Transaction Logs (Roll-forward) (see section 8.5)
- Backing up and Restoring Storage Groups Between Remote Sites (Remote Copy) (see section 8.6)
- Using a Backup Management Product That Cannot Link with Protection Manager (see section 8.7)
- Using File Sharing to Link the Database Server and Backup Server (see section 8.8)
- Deleting backup Data (see section 8.9)

8.1 Operations for Backing up and Restoring a Storage Group

This chapter provides examples of backup and restore operations:

- Backing up and restoring storage groups between volumes
- Backing up and restoring storage groups to and from tape
- Restoring the data obtained by the VSS backup in units of Information Stores.
- Restoring by using transaction logs (Roll-forward).
- Backing up and restoring storage groups between remote sites
- Backing up and restoring groups with a backup management product that is unable to link with Protection Manager
- Backing up and restoring storage groups by using a file share

The rest of this section explains precautions on backing up and restoring storage groups. For details on common operating notes that apply to all kinds of objects subject to backup, see section 5.3.

8.1.1 Precautions When Backing up a Database

- When executing online backup with the `drmexgbackup` command, forcibly copy the data from the primary volume to the secondary volume while the Information Store is still mounted. It may be impossible to restore the backup data acquired by this function. To ensure restoration of the backup data, first use the `drmexgbackup` command to perform a cold backup. Make sure that the backup data is stored, and then perform an online backup.
- When executing online backup with the `drmexgbackup` command, confirm that no other volume is mounted in the directory on the volume subject to backup.
- When you use the `drmexgbackup` command to obtain repeated backups, you may experience periods in which the Exchange transaction log expands very suddenly. In this case, specify the `-transact_log_del` option when executing the `drmexgbackup` command. However, since transaction log files are deleted after backup when this option is specified, you will only be able to perform restoration with roll-forward recovery (executing the `drmexgrestore` command with the `-recovery` option specified) for the most recent backup. You cannot perform restoration with roll-forward recovery for backups obtained in the past.
- When specifying the `-transact_log_del` option to execute a cold backup on a clustered server system configuration, specify the `-c online` option at the same time.
- Do not access the operation-target volume by, for example, using Windows Explorer while a backup or restoration command is being executed in a VERITAS Cluster Server for Windows environment. Doing so might mistakenly place the cluster resources offline, resulting in an error in the backup or restoration command.
- When a VSS backup is executed, the following event information might be output to the Windows application event log on the backup server:

  Event type: error

Event source: PlugPlayManager

Event ID: 12

or

Event type: warning

Event source: ftdisk

Event ID: 57

or

Event type: warning

Event source: disk

Event ID: 51

This event information is logged during a VSS backup when the secondary volume becomes temporarily hidden. However, this will not influence the backup operation.

- If the Windows Disk Management window is being displayed on the backup server or on the database server to which no service packs have been applied in Windows Server 2003, close the window before performing the VSS backup. If the command is executed with the window remaining open, the VSS backup might fail.

8.1.2 Precautions When Restoring a Database

8.1.2.1 Precautions When Restoring from a Tape

When using the `drmtaperestore` command, `drmmediarestore` command, or `EX_DRM_TAPE_RESTORE` extended command to restore an Exchange Server storage group, you need to delete the Exchange Server transaction log files remaining on the secondary volume. On the backup server, delete the transaction log files as follows. Note that if you are using NetBackup for the tape backup management product, you do not need to perform the following when performing tape backup with the `-raw` option specified.

1. Execute the `drmtapecat` command with the `-o MSEXCHANGE` option specified. Then check the backup ID of the tape backup used for restoration, and check the absolute paths to the folders containing the transaction log files for the storage group.

   It is necessary to check the absolute paths to the folders containing the transaction log files, for all the storage groups that have been backed up with the backup ID used for restore operations.

2. Use the `drmmount` command to mount the secondary volume. Use the backup ID determined from step 1.

3. Check the folder in which the transaction log files specified in step 1 are stored, as well as the files stored in the folder.

Note that when you use the `drmmount` command to mount the secondary volume, the drive letter allocated may differ from that during backup. In this case, replace the drive letter of the transaction log file checked in step 1 with the driver letter allocated using the `drmmount` command.

4. Delete any files in the folder displayed in step 3 that have the `.log` extension.
5. Perform steps 3 and 4 for all storage groups backed up.
6. Use the `drmumount` command to unmount the secondary volume. The backup ID used here is the same as that used in step 1.

8.1.2.2 Performing a Recovery Using Roll Forward

If recovery is performed using roll forward (the `drmexgrestore` command is executed with the `-recovery` option specified) and there are insufficient transaction logs, the `KAVX1510-E` or `KAVX1703-E` message will be output. If the `drmexgbackup` command with the `-transact_log_del` option is used to delete the transaction logs, roll forward cannot be performed for any backup obtained before that point.

8.1.2.3 Procedure for Handing Message (KAVX1510-E)

In restoration with roll-forward recovery (executing the `drmexgrestore` command with the `-recovery` option specified), if the storage group to recover has more than 1,008 transactions log files, roll-forward will be completed when the `drmexgrestore` command is executed, but the Information Store will not be mounted. In this case, the `KAVX1510-E` message is output.

Open the OS event viewer, and make sure the following events are recorded in the application log:

- Check that an error log with event source `MSExchangeIS` and event ID `9518` is recorded. Then, open the properties of the error log, and check that the description field reads `Error 0xfffffd9a has occurred.`
- If the above error log has not been recorded, check that an information log with event source `ESE`, event ID `302`, and description `The database engine has successfully completed the restoration step` is recorded. Also check that this log is followed immediately by an information log with event source `ESE`, event ID `101`, and description `The database engine has stopped.`

If you confirm that the application logs are recorded, mount the Information Store as follows:

1. Start the system manager, view the properties of the storage group to be recovered, and check the location of the transaction logs as well as the initial characters of the log file. The initial characters are `E0`*`n`*, where *`n`* is from 0 to 3.
2. Of the transaction log files in the storage group to be recovered, move those other than the ones named `E0`*`n`*`.log` (where *`n`* is from 0 to 3) to a new folder.
3. From the system manager, mount the Information Store to be recovered.
4. Check that the Information Store is mounted, and then delete the transaction log files moved in step 2.

8.1.2.4 About Restoring Data Obtained with a VSS Backup

- If the Windows Disk Management window is being displayed on the backup server or on the database server to which no service packs have been applied in Windows Server 2003, close the window that displays the Windows Disk Management window before restoring the data obtained in the VSS backup. If the command is executed with the window remaining open, restoration may fail.
- While restoring data obtained in a VSS backup, you must not open the physical disk resource properties on the cluster administrator window. If the properties are open, restoration may fail.
- If transactions logs to which Exchange Server 2003 Service Pack 1 is not applied are mixed with transactions logs to which it is applied, restoration of data obtained using the VSS backup may fail. In such a case, an event log with an event ID of 454 will be output.

  If this occurs, recover using the following procedure:

  a Execute the following command for each of the `*.edb` and `*.stm` files to be restored, and record the leftmost numeric values displayed for `[Last Consistent]`.

  `eseutil /mh` *`name-of-file-to-be-restored`*

  The `eseutil` command is located in *`Exchange-Server-installation-folder`*`\bin`.

  b The minimum number among these recorded values is the `Last Consistent` log number. Move any transaction logs that have a lower number to a different folder.

  For example, if the `Last Consistent` log number was `1ab`, transaction log files `E0x00001.log` to `E0x001aa.log` would be moved.

  c Mount all information stores for the storage groups to be restored.

8.1.2.5 When the Information Store Cannot Be Mounted

If the data to be restored was obtained by using a VSS backup of version 04-40 or earlier, the Information Stores to be restored must be mounted. If the Information Store cannot be mounted, perform the following procedure:

1. Move the data files (`*.edb` and `*.stm`) for the Information Store that cannot be mounted to another folder.
2. Mount the Information Store.

   When the Information Store can be mounted, you can perform restoration for obtaining the database at the time of backup (executing the `drmexgrestore` command without the `-recovery` option specified) and restoration with roll-forward recovery (executing the `drmexgrestore` command with the `-recovery` option specified).
3. If an Information Store that cannot be mounted still exists after performing the above steps, the transaction log files (`*.log`) or checkpoint files (`*.chk`) of the storage group might be corrupted.

Move the transaction log files (`*.log`), checkpoint files (`*.chk`), and data files (`*.edb` and `*.stm`) for all the Information Stores in the storage group to another directory. After all the files are moved, mount all of the Information Stores. In such a case, you can only perform restoration for obtaining the database at the time of backup (executing the `drmexgrestore` command without the `-recovery` option specified).

8.1.3 Precautions on Recovery Storage Groups

Do not specify a recovery storage group when specifying a storage group, if you are using Exchange Server 2003 or Exchange Server 2007 as the database management system.

8.1.4 Precautions When Configuring Volumes

When reverting a database (executing the `drmexgrestore` command without the `-recovery` option specified) during online backup, set up the volume as follows:

Set the same RAID Manager group name for volumes that contain data files (`*.edb` and `*.stm`) transaction log files (`*.log`) and checkpoint files (`*.chk`).

8.1.5 Precautions on Database Verification

Use the `drmexgverify or EX_DRM_EXG_VERIFY` command to verify the integrity of the data that was backed up by using the `drmexgbackup` or `EX_DRM_EXG_BACKUP` command, online, or by using VSS. During online backup, the `drmexgverify` and `EX_DRM_EXG_VERIFY` commands verify the integrity of the Exchange database (`*.edb` files) only, and not that of the Exchange streaming database (`*.stm` files). This means that even if the `drmexgverify` command successfully terminates the verification, the Exchange streaming database may have been corrupted. In this case, the backup data may not be able to be restored.

8.2 Backing up and Restoring Data Between Volumes

In a configuration that uses only one server, you use commands to back up and restore data between the primary volume and secondary volume. Production Manager must be installed on the database server in order to execute commands.

Figure 8.1 shows a system configuration for backing up and restoring data between volumes.

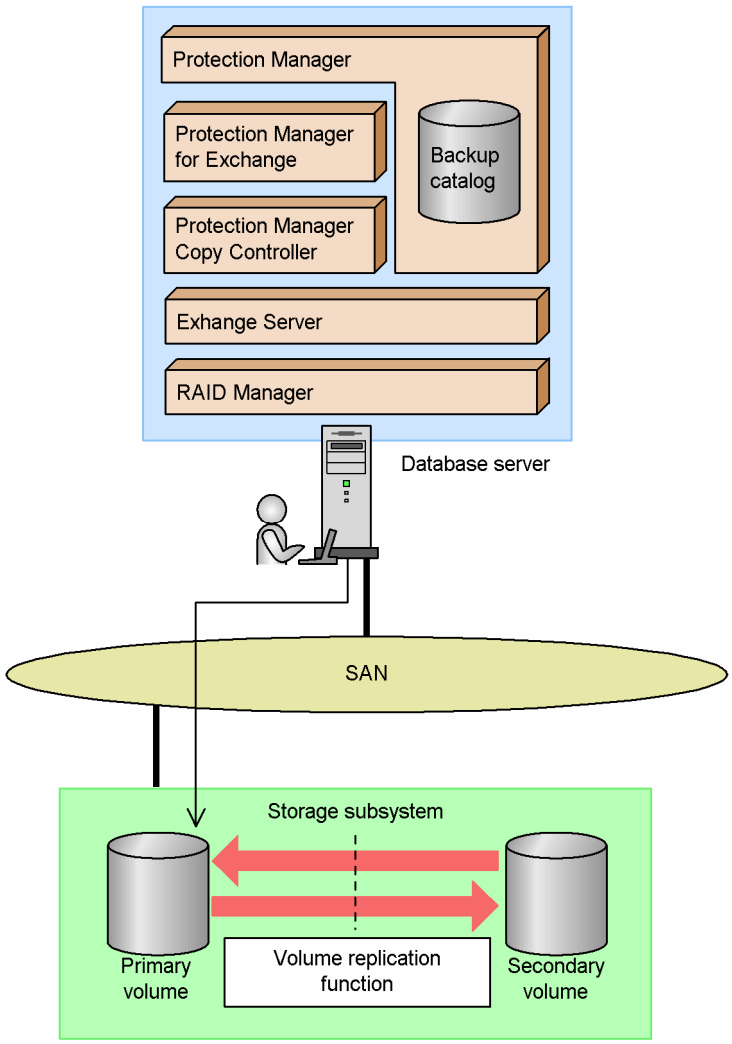


Figure 8.1 Backing up and Restoring Data Between Volumes (Exchange Database backup)

8.2.1 Processing Flow

The processing flow for backing up storage groups to a secondary volume and restoring them to the primary volume on a single-server system is shown in Figure 8.2 and Figure 8.3. Execute commands on the database server.

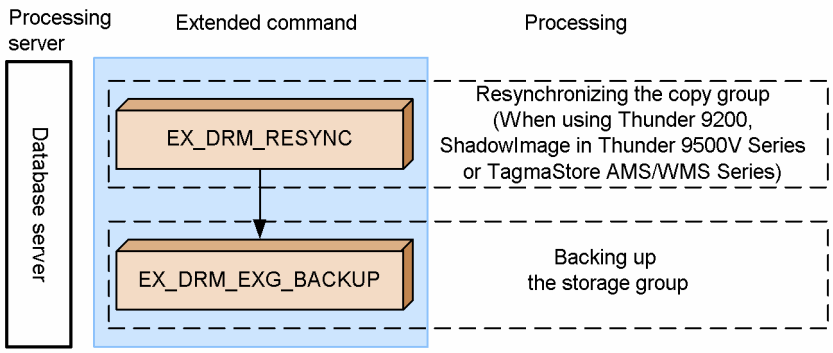


Figure 8.2 Processing Flow for Backing up Storage Groups to a Secondary Volume

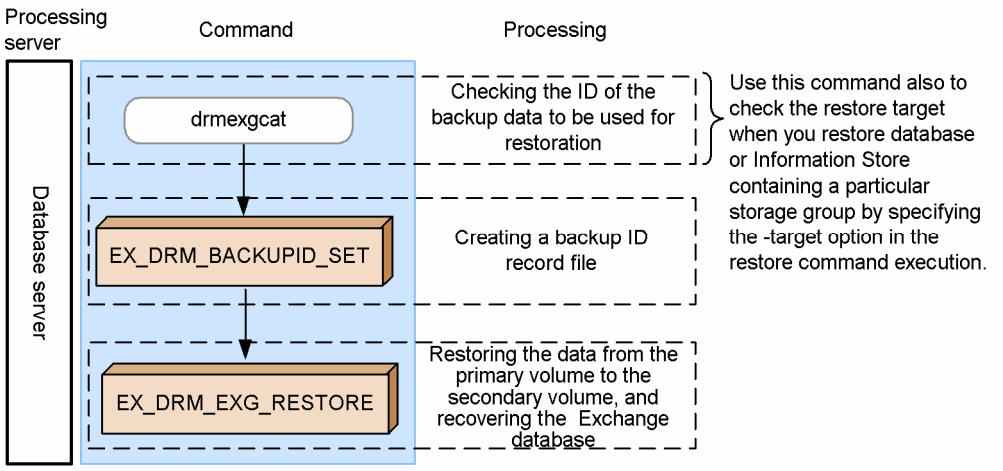


Figure 8.3 Processing Flow for Restoring Storage Groups on the Primary Volume

8.2.2 Backing up Storage Groups to a Secondary Volume

This section describes the processing flow for backing up storage groups to a secondary volume on a single-server system.

To back up a storage group:

1. Resynchronize the copy groups.

   If you are using a Thunder 9200 Series, and ShadowImage with a Thunder 9500V Series or a TagmaStore AMS/WMS Series, execute `EX_DRM_RESYNC` to resynchronize the copy group. Resynchronizing the copy groups before you perform backup enhances the speed of backup processing. In this example, the copy group name is `VG01,vol01`.

```
DBServer > EX_DRM_RESYNC Operation_A -cg VG01,vol01 -copy_size 7
```

2. Back up the storage group to a secondary volume.

   Back up the storage group. To perform backup, execute `EX_DRM_EXG_BACKUP`. Specify the operation ID `Operation_A` as the argument. In this example, delete the transaction log file to execute a cold backup.

```
DBServer > EX_DRM_EXG_BACKUP Operation_A -mode cold -transact_log_del
```

   Specify `-mode cold` to execute a cold backup. Even when `-mode cold` is not specified, a cold backup will be executed as if this option had been specified.

   Specify `-mode online` to execute an online backup. When an online backup is executed, you cannot specify the `-transact_log_del` option.

8.2.3 Restoring Storage Groups to a Primary Volume

This section explains how to restore a storage group when backup data has been saved on a secondary volume. In this example, a secondary volume is resynchronized with the primary volume to restore the storage group.

To restore a storage group:

1. Check the backup ID of the backup data.

   To check the backup ID of the backup data used to restore it from a secondary volume to the primary volume, execute the `drmexgcat` command on the database server.

```
DBServer> drmexgcat
STORAGEGROUP: STORAGE_1
BACKUP-ID:0000000001 BACKUP-MODE: COLD STORAGEGROUP: STORAGE_1 ORIGINAL-ID:0000000001
START-TIME:2003/06/01 17:45:36 END-TIME: 2003/06/01 17:52:36
:
:
```

   When restoring a database or Information Store relating to a specific storage group by executing the restoration command with the `-target` option specified, check the restoration target.

2. Create a backup ID record file.

Execute EX_DRM_BACKUPID_SET with the backup ID specified to create a backup ID record file. Here, use operation ID Operation_A.

```
DBServer > EX_DRM_BACKUPID_SET Operation_A -backup_id 0000000001
```

3. Restore data from the secondary volume to the primary volume.

   Execute EX_DRM_EXG_RESTORE on the database server to resynchronize the primary and secondary volume and restore the data.

```
DBServer > EX_DRM_EXG_RESTORE  Operation_A -resync
```

   Specify operation ID Operation_A for EX_DRM_EXG_RESTORE. Protection Products acquire information required for restoration from the operation definition file. Also, Protection Products acquire the backup ID from the backup ID record file registered in step 2, and restores the data.

8.3 Backing up and Restoring Using a Tape Device

The examples in this section assume the system configuration shown in Figure 8.4. A single-database-server system is used in this example, but a multi-database-server system may be used as appropriate.

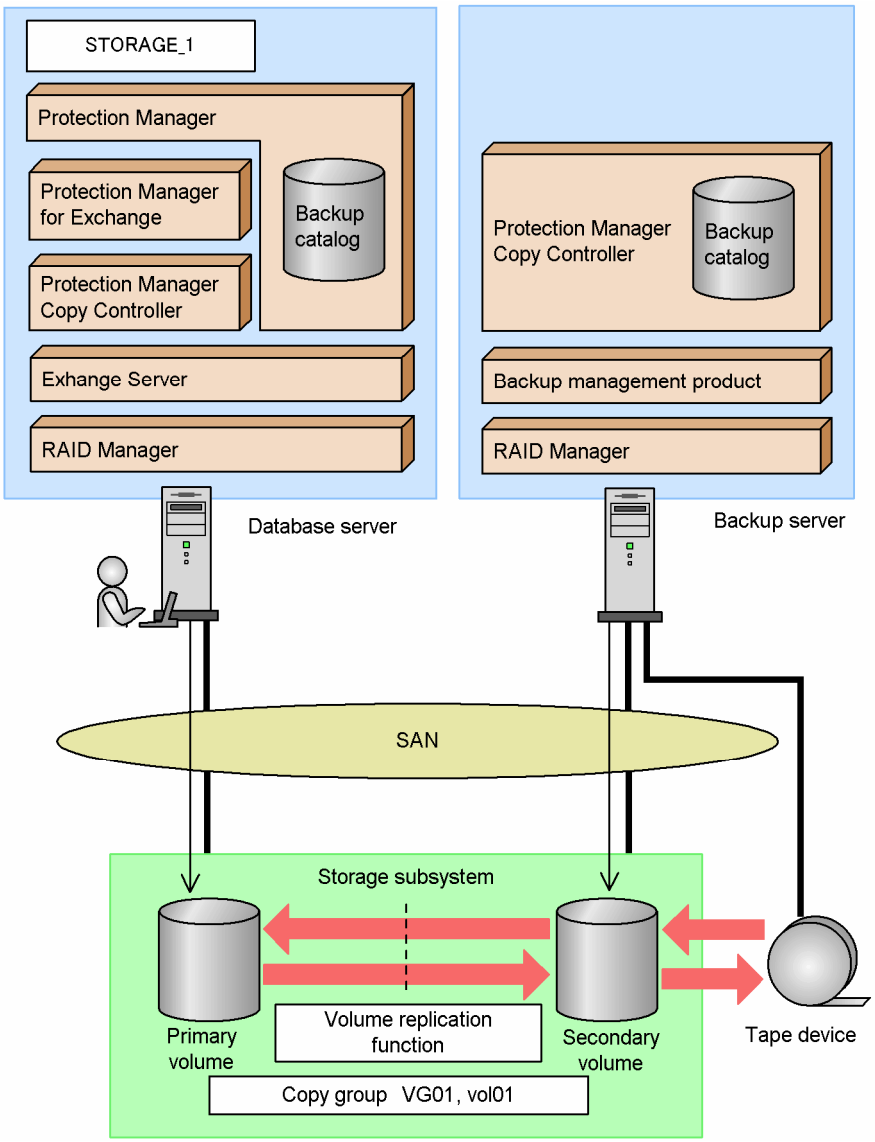


Figure 8.4 System Configuration for Backing up and Restoring Storage Groups to and from Tape

With Protection Manager, database servers can be clustered. With database servers configured in a failover cluster, if the active server fails, a standby server is available to take over operations from the active server.

The examples in this section assume the following conditions:

- The primary volume is NTFS-formatted.
- The primary volume and the secondary volume are pair defined in the database server and the backup server.
- The backup management product is installed on the backup server.
- The parameters of the backup management product have been registered by using the `drmtapeinit` command.
- The FTP service on the backup server is started, and the system is set up so a user who has logged onto the OS of the database server can log onto the FTP server and transfer files to the FTP server. The FTP user ID is `admin`, and the FTP user password is `password`.
- The Exchange Server used in the system is an Exchange Server 2003 Enterprise Edition.
- The storage group (`STORAGE_1)` exists on database server (`DBServer`), and services are running on database server.
- Temporary directory for Extended Commands is created by the database server and backup server.
- The secondary volume is not always mounted. It is mounted on the E drive (drive character `E:`) only while it is operating.
- The primary volume configured as a pair with the secondary volume to be backed up will not be backed up until the secondary volume is backed up onto tape.
- The `cscript //H:Cscript` command is executed from the command prompt and the host parameters are updated.

8.3.1 Backing up Storage Groups

This section describes the processing flow for backing up storage groups to tape and restoring them on a multi-server system (Figure 8.5 and Figure 8.6). Use extended commands to execute cold backups and backups with VSS.

- When usual operation is in the split state, resynchronize the copy group before backing up the data.
- When usual operation is in the pair state, there is no need to resynchronize the copy group before backing up data. Instead, when tape backup is complete, resynchronize the copy group to initialize it.

When backing up a storage group in an online backup operation, you must verify the integrity of the Exchange database before backing up the data backed up to tape.

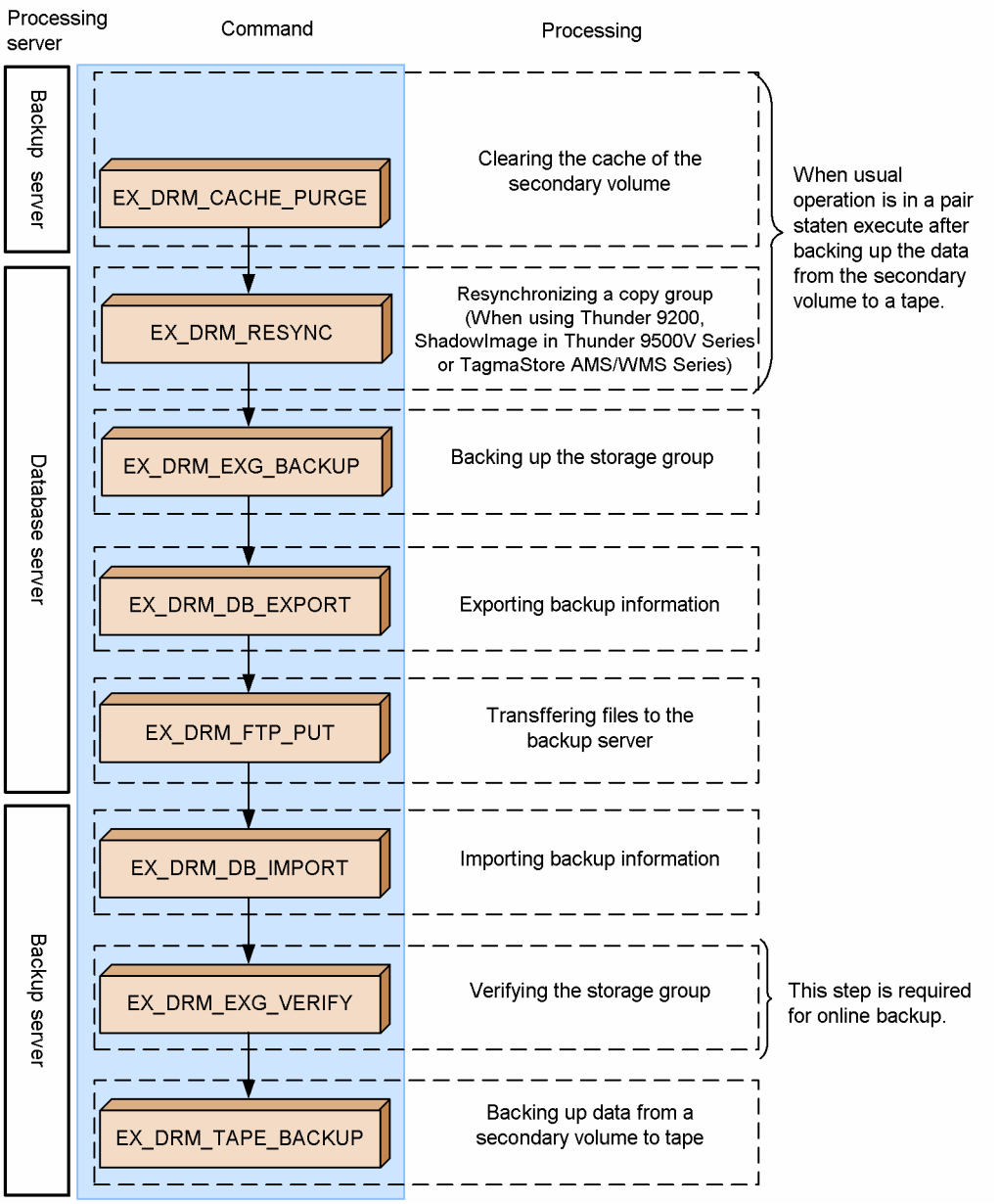


Figure 8.5 Processing Flow for Backing up Storage Groups to Tape

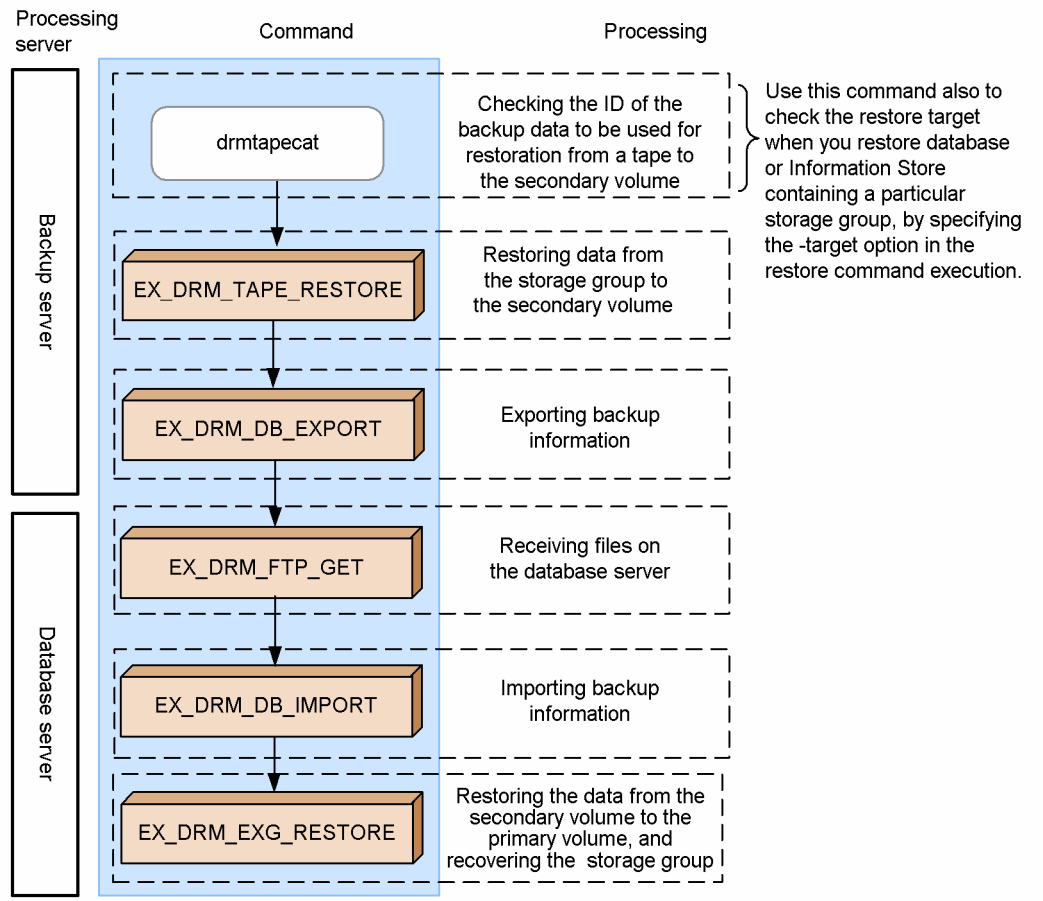


Figure 8.6 Processing Flow for Restoring Storage Groups from Tape

8.3.2 Backing up Storage Groups to Tape

8.3.2.1 Resynchronizing Copy Groups

When usual operation is in the split state, resynchronize the copy group before backing up the data.

When usual operation is in the pair state, there is no need to resynchronize the copy group before backing up data. Instead, when the tape backup is complete, resynchronize the copy group to initialize it.

To resynchronize a copy group:

1. Clear the cache of the secondary volume.

   Before performing backup, clear the system cache of the backup server.

   To clear the system cache, execute `EX_DRM_CACHE_PURGE` on the backup server and mount and then unmount the secondary volume. In this example, the copy group name is `VG01,vol01`.

```
BKServer > EX_DRM_CACHE_PURGE Operation_A -cg VG01,vol01
```

2. Resynchronize the copy group.

   If you are using a Thunder 9200 Series, and ShadowImage with a Thunder 9500V Series or a TagmaStore AMS/WMS Series, execute `EX_DRM_RESYNC` to resynchronize the copy group. Resynchronizing the copy group before backing up data will help reduce the backup time.

```
DBServer > EX_DRM_RESYNC Operation_A -cg VG01,vol01 -copy_size 7
```

8.3.2.2 Cold Backup

An example of backing up storage groups to tape in a cold backup is described below. In a cold backup, the Information Store to be backed up is unmounted and remains inaccessible until the backup completes. This is an example of performing a cold backup onto tape of an Exchange database. In this example, an cold backup of storage group `STORAGE_1` for the database server `DBServer` is performed, and the backup is placed on the secondary volume. The backup is then backed up onto tape from the secondary volume. The backup uses `Operation_A` as the operation ID.

To back up a storage group in a cold backup:

1. Back up the storage group to a secondary volume.

   Perform a cold back up of the storage group. To perform backup, execute `EX_DRM_EXG_BACKUP`. Specify the operation ID `Operation_A` as the argument. In the following example, the command deletes the transaction log file and backs up the database:

```
DBServer > EX_DRM_EXG_BACKUP Operation_A -mode cold -transact_log_del
```

2. Check that the storage group has been backed correctly.

   To check that backup information bearing the date on which the storage group had been backed up from the primary volume to the secondary volume is available, execute the `drmexgcat` command on the database server.

```
DBServer> drmexgcat -target STORAGE_1
STORAGEGROUP: STORAGE_1
BACKUP-ID:0000000001 BACKUP-MODE: COLD STORAGEGROUP: STORAGE_1 ORIGINAL-ID:0000000001
START-TIME:2003/06/01 17:45:36 END-TIME: 2003/06/01 17:52:36
:
:
```

   In this example, information about the backup operation carried out in step 1 has been loaded in the backup catalog with the new backup ID `0000000001`.

3. Export the backup information to temporary files.

   To back up data from the secondary volume to a tape, you must copy information about backup that was performed from the primary volume to the secondary volume. Copy this information to the backup server. To export information about backup to temporary files, execute `EX_DRM_DB_EXPORT`. Temporary files are stored in the temporary folder for extended commands, on the database server.

```
DBServer > EX_DRM_DB_EXPORT Operation_A
```

4. Transfer the temporary files to the backup server.

   Batch-transfer the temporary files from the database server to the backup server. To transfer the files, execute `EX_DRM_FTP_PUT` on the database server. In this example, the user ID used for logon to the FTP server is `admin` and the password is `password`. Temporary files are stored in the temporary folder for extended commands, on the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

5. Import the temporary files, which were transmitted from the database server, into the backup catalog on the backup server.

   To import the temporary files, which were transmitted from the database server, into the backup catalog on the backup server, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

6. Back up the data of the secondary volume to a tape.

   To back up the data, execute `EX_DRM_TAPE_BACKUP` on the backup server. In this example, the drive containing the secondary volume is `E:`.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   When you execute backup, information about this backup is registered in the backup catalog with the new backup ID `0000000002`.

8.3.2.3 Online Backup

An example of backing up a storage group to tape in an online backup operation is described below. Online backup operations are executed with an Information Store and a disk drive mounted in position. In this example, an online backup of storage group `STORAGE_1` for the database server `DBServer` is performed on the secondary volume, and then the consistency of the Exchange database is verified. Next, the storage group is backed up from the secondary volume onto tape. `Operation_A` is used as the operation ID.

To back up a storage group in an online backup operation:

1. Back up a storage group to a secondary volume.

   To back up a storage group in an online backup operation, execute `EX_DRM_EXG_BACKUP`. Specify the operation ID `Operation_A` as an argument.

```
DBServer > EX_DRM_EXG_BACKUP Operation_A -mode online
```

2. Check that the storage group has been backed correctly.

   To check that backup information bearing the date on which the storage group had been backed up from the primary volume to the secondary volume is available, execute the `drmexgcat` command on the database server.

```
DBServer> drmexgcat -target STORAGE_1
STORAGEGROUP: STORAGE_1
BACKUP-ID:0000000001 BACKUP-MODE: ONLINE STORAGEGROUP: STORAGE_1 ORIGINAL-ID:0000000001
START-TIME:2003/06/01 17:45:36 END-TIME: 2003/06/01 17:52:36
:
:
```

   In this example, information about the backup operation carried out in step 1 has been loaded in the backup catalog with the new backup ID `0000000001`.

3. Export the backup information to temporary files.

   To back up from the secondary volume onto tape, you must copy onto the backup server all information relating to the backup operation from the primary volume to the secondary volume. To export into temporary files the information relating to the backup operation, execute `EX_DRM_DB_EXPORT`. The temporary files are stored in the temporary folder for extended commands, on the database server.

```
DBServer > EX_DRM_DB_EXPORT Operation_A
```

4. Transfer the temporary files to the backup server.

   Transfer the temporary files from the database server to the backup server all at once. To transfer these files, execute the `EX_DRM_FTP_PUT` command on the database server. In the following example, the command uses the user ID `admin` and the password `password` to log on to the FTP server. The temporary files are stored in the temporary folder for extended commands, on the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

5. Import the temporary files, which were transmitted from the database server, into the backup catalog for the backup server.

   To import the temporary files, which were transmitted from the database server, into the backup catalog for the backup server, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

6. Verify the storage group thus backed up.

   To verify the consistency of the Exchange database, which was backed up online, onto the secondary volume, execute the `EX_DRM_EXG_VERIFY` command on the backup server.

```
BKServer > EX_DRM_EXG_VERIFY Operation_A
```

7. Back up the data on the secondary volume onto tape.

   To back up the data, execute the `EX_DRM_TAPE_BACKUP` command on the backup server. In the following example, the command specifies `E:` as the drive characters of the secondary volume:

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   When you execute the backup, information relating to this backup operation is registered in the backup catalog with the new backup ID `0000000002`.

8.3.2.4 Backup Using VSS

An example of using VSS to back up a storage group is described below. Backup operations using VSS are executed with an Information Store and a disk drive mounted in position. In this example, the system first backs up the storage group, `STORAGE_1` in the database server `DBServer` into the secondary volume, and then backs them up from the secondary volume to tape. The Operation ID used in this example is `Operation_A`.

To perform a VSS backup, the Protection Manager service must be running on the backup server.

To use VSS to back up a storage group:

1. Back up the storage group to the secondary volume.

   Using VSS, back up the storage group in an online backup operation. To perform backup, execute `EX_DRM_EXG_BACKUP`. Specify the operation ID `Operation_A` as the argument.

```
DBServer > EX_DRM_EXG_BACKUP Operation_A -mode -vss
```

2. Check that the storage group has been backed up correctly.

   To check that backup information bearing the date on which the storage group had been backed up from the primary volume to the secondary volume is available, execute the `drmexgcat` command on the database server.

```
DBServer > drmexgcat -target STORAGE_1
STORAGEGROUP: STORAGE_1
BACKUP-ID:0000000001 BACKUP-MODE: VSS STORAGEGROUP: STORAGE_1 ORIGINAL-ID:0000000001
START-TIME:2003/06/01 17:45:36 END-TIME: 2003/06/01 17:52:36
:
:
```

   In this example, information about the backup operation carried out in step 1 has been loaded in the backup catalog with the new backup ID `0000000001`.

3. Export the backup information to temporary files.

   To back up data from the secondary volume to a tape, you must copy information about backup that was performed from the primary volume to the secondary volume. Copy this information to the backup server. To export information about backup to temporary files, execute `EX_DRM_DB_EXPORT`. Temporary files are stored in the temporary folder for extended commands, on the database server.

```
DBServer > EX_DRM_DB_EXPORT Operation_A
```

4. Transfer the temporary files to the backup server.

   Batch-transfer the temporary files from the database server to the backup server. To transfer the files, execute `EX_DRM_FTP_PUT` on the database server. In this example, the user ID used for logon to the FTP server is `admin` and the password is `password`. Temporary files are stored in the temporary folder for extended commands, on the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

5. Import the temporary files, which were transmitted from the database server, into the backup catalog on the backup server.

   To import the temporary files, which were transmitted from the database server, into the backup catalog on the backup server, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

6. Back up the data of the secondary volume to a tape.

   To back up the data, execute `EX_DRM_TAPE_BACKUP` on the backup server. In this example, the drive containing the secondary volume is `E:`.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt E:
```

   When you execute backup, information about this backup is registered in the backup catalog with the new backup ID `0000000002`.

8.3.3 Restoring Storage Groups from Tape

This section provides an example of restoring data that has been backed up to tape, and recovering a storage group. In this example, data is restored onto a secondary volume from tape, and the secondary volume is resynchronized with the primary volume to restore (recover) the data. This example uses `Operation_A` as the operation ID.

To perform restoration of data from a VSS backup, the Protection Manager service must be running on the backup server.

To restore a storage group:

1. Check the backup ID of the backup data.

   To check the backup ID of the backup data used to restore the data from tape to the secondary volume, execute the `drmtapecat` command.

```
BKServer >drmtapecat -hostname DBServer -l
BACKUP-ID  ORIGINAL-ID HOSTNAME   BACKUP-OBJECT INSTANCE SNAPSHOT TIME     EXPIRATION TIME
BACKUP-MEDIA
0000000002 0000000001  DBServer MSEXCHANGE   -     2002/06/01 17:45:36 PERMANENT    MEDIA1
:
:
```

   When restoring a database or Information Store relating to a specific storage group by executing the restoration command with the `-target` option specified, you must confirm the restoration target. In this case, execute the `drmtapecat` command with the following options specified:

   - `-o MSEXCHANGE`
   - `-backup_id` ***backup-ID***

2. Restore the backed up data from the tape to the secondary volume.

   To restore the data, execute `EX_DRM_TAPE_RESTORE` on the backup server.

```
BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002
```

3. Export backup information to temporary files.

   To restore the data from the secondary volume to the primary volume, you must copy backup information about the restore performed from the tape to the secondary volume. Copy this information to the database server. To export the backup information to temporary files in the temporary folder for extended commands, execute `EX_DRM_DB_EXPORT`.

```
BKServer > EX_DRM_DB_EXPORT Operation_A
```

4. Receive the temporary files on the database server.

   To batch-receive the temporary files held on the backup server onto the database server, execute `EX_DRM_FTP_GET` on the database server. In this example, the user ID used for logon to the FTP server is `admin` and the password is `password`. Temporary files are stored in the temporary folder for extended commands, on the database server.

```
DBServer > EX_DRM_FTP_GET Operation_A -server BKServer -user admin -password password
```

5. Import the temporary files, which were transmitted from the database server, into the backup catalog on the database server.

   To import the temporary files, which were transmitted from the database server, into the backup catalog on the database server, execute `EX_DRM_DB_IMPORT` on the database server.

```
DBServer >EX_DRM_DB_IMPORT Operation_A
```

6. Restore the data of the secondary volume to the primary volume.

   Resynchronize the primary volume and secondary volume and then restore the data. To restore the data, execute `EX_DRM_EXG_RESTORE` on the database server.

```
DBServer > EX_DRM_EXG_RESTORE Operation_A -resync -recovery
```

   Note: If data was backed up using VSS, the data can be restored in units of Information Stores. To restore data in units of Information Stores, specify the Information Store name in the `-target` option.

   An example of executing the `drmexgcat` command immediately before restoring data from the secondary volume to the primary volume is shown below:

```
DBServer > drmexgcat -target STORAGE_1
STORAGEGROUP: STORAGE_1
BACKUP-ID:0000000002 BACKUP-MODE: VSS STORAGEGROUP: STORAGE_1 ORIGINAL-ID:0000000001
START-TIME:2002/06/01 10:30:00 END-TIME: 2002/06/01 10:31:00 HOSTNAME:DBServer
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE      COPY-GROUP
M MAILBOXSTORE MAILBOX1(STORAGE_1)  E:\STR1\priv1.edb      E:          -       Harddisk3
VG01,vol01
- -          -                -                 -         -      -          VG01,vol01
M MAILBOXSTORE MAILBOX1(STORAGE_1)  E:\STR1\priv1.stm      E:          -       Harddisk3
VG01,vol01
- -          -                -                 -         -      -          VG01,vol01
:
:
```

   The following is an example of executing restoration when the backup data shown above exists, and you want to restore only Information Store `MAILBOX1` that belongs to storage group `STORAGE_1`:

```
DBServer > EX_DRM_EXG_RESTORE Operation_A -resync -recovery -target STORAGE_1\MAILBOX1
```

   To perform restoration in units of Information Stores, you need to execute a roll-forward with the `-recovery` option specified. For details on roll-forward, see section 8.5.

8.4 Backing Up a Storage Group by Using a User Script

By using a backup command that specifies a user script, you can use a single operation to back up a storage group to tape from the primary volume via the secondary volume.

8.4.1 System Configuration

The example is based on the system configuration shown in the figure below.

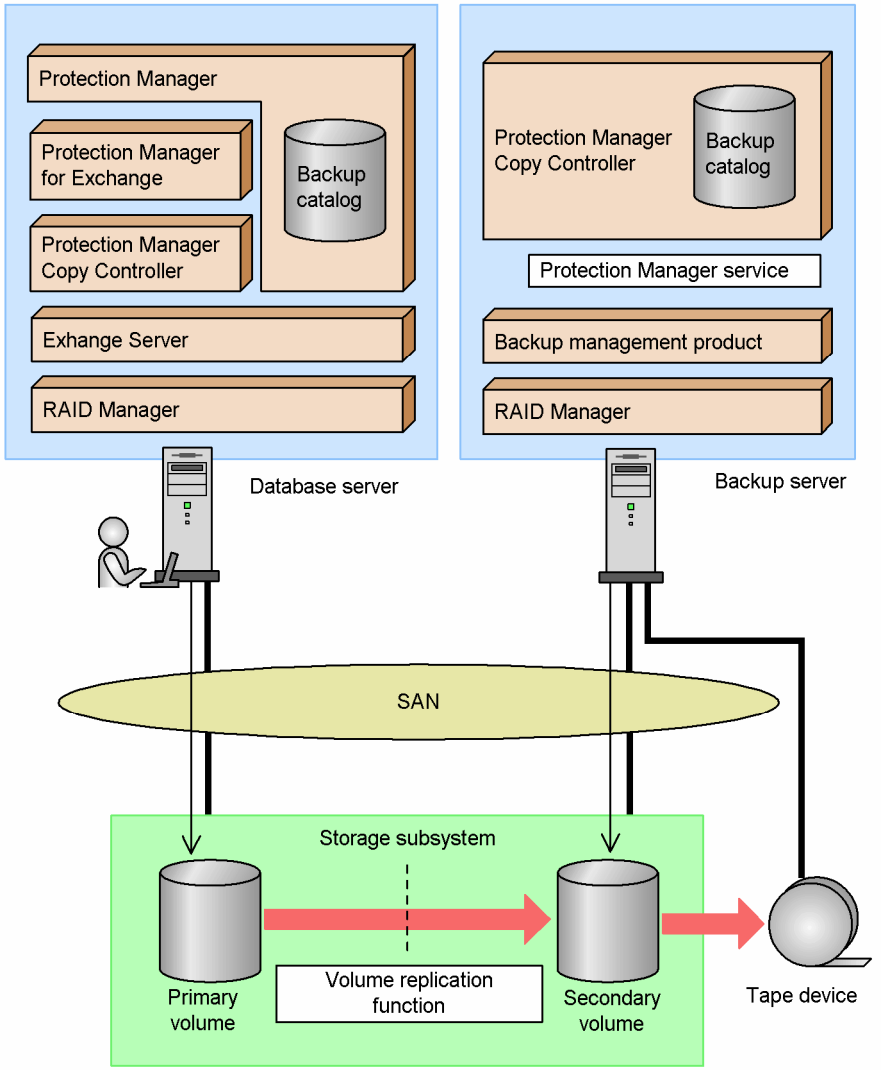


Figure 8.7 System Configuration for Backup of a Storage Group to Tape

8.4.2 Processing Flow

- In this example, the processing flow of the `drmexgbackup` command that specifies a user script is as follows:
- A VSS backup of the storage group is performed.

- Once the backup from the primary volume to the secondary volume has been completed, the backup catalog is transferred to the backup server.
- On the backup server, the secondary volume is mounted on the `G` and `H` drives.
- The secondary volume is backed up to tape by using `NTBACKUP`, and then is unmounted (this is a step in the user post-processing section).

The `drmexgbackup` command terminates without waiting for the tape backup operation to terminate.

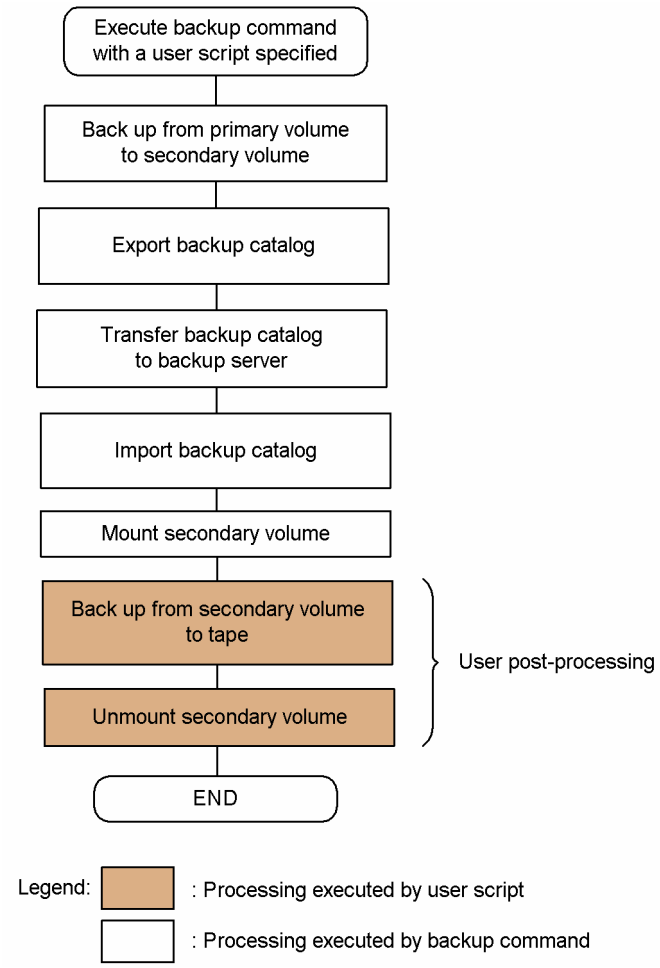


Figure 8.8 Processing Flow

8.4.3 User Script Example

Table 8.1 shows the user script that is created for the example.

Table 8.1 User Script Creation Example

Script text	Explanation
LOCAL_BACKUP=YES *...(1)* #Pre-processing section #None #Post-processing section [POST_PROC] *...(2)* #Tape backup of the secondary volume [CMD] CMDLINE=C:\tmp\tapebackup.bat TIMEOUT=NOWAIT *...(3)* END_CODE=TERMINATE_NZ *...(4)* LOCATION=REMOTE *...(5)* PARENT_STAT=NORMAL *...(6)*	(1) YES is required. (2) Start of the user post-processing section (3) Execute the next command without waiting for the command to terminate. (4) Treat a command return value that is a value other than 0 as an error. (5) Execute on the remote server. The -s option is required for the backup command. (6) Execute only when the backup command is normal.

The following is an example of backing up the secondary volume to tape using tapebackup.bat:

```
rem Use NTBACKUP to execute the job Job1, and perform a copy backup of G:\ and H:\ to tape Tape1
rem For the backup source specification, use the backup selection file (C:\tmp\exg.bks)
rem Set the backup comment passed to the DRMENV_COMMENT environment variable as the backup job explanation
"C:\Windows\system32\ntbackup.exe" backup "@C:\tmp\exg.bks" /j "Job 1" /a /t "Tape 1" /D
"%DRMENV_COMMENT%" /m copy
IF NOT "%errorlevel%"=="0" GOTO ERROR
rem After the tape backup, specify the backup ID imported to the backup server, and unmount the secondary
volume
"C:\Program Files\Hitachi\DRM\bin\drmumount.exe" %DRMENV_R_BACKUPID%
IF NOT "%errorlevel%"=="0" GOTO ERROR
exit 0
:ERROR
exit 1
```

Note: A line that begins with the characters rem is a comment line.

8.4.4 Backup Execution Example

An example of using a user script is shown below:

```
PROMPT>drmexgbackup -mode vss -script C:\tmp\script.txt -s BKHOST -auto_import
-auto_mount G: -comment TEST1
```

8.5 Restoring by Using Transaction Logs

When backup data on Exchange Server is restored, the database returns to the state it was in at the time of the backup. If the transaction logs from when the backup was executed up until when the restoration command was executed are stored on the primary volume, the transaction logs can be rolled forward by specifying the `-recovery` option. In this case, by applying the transaction logs after backup, the database can be recovered to the state it was in immediately before the error occurred.

The procedure for using the transaction logs of Exchange Server to restore backup data by using roll-forward is described below.

Figure 8.9 shows a flow of operation using transaction logs.

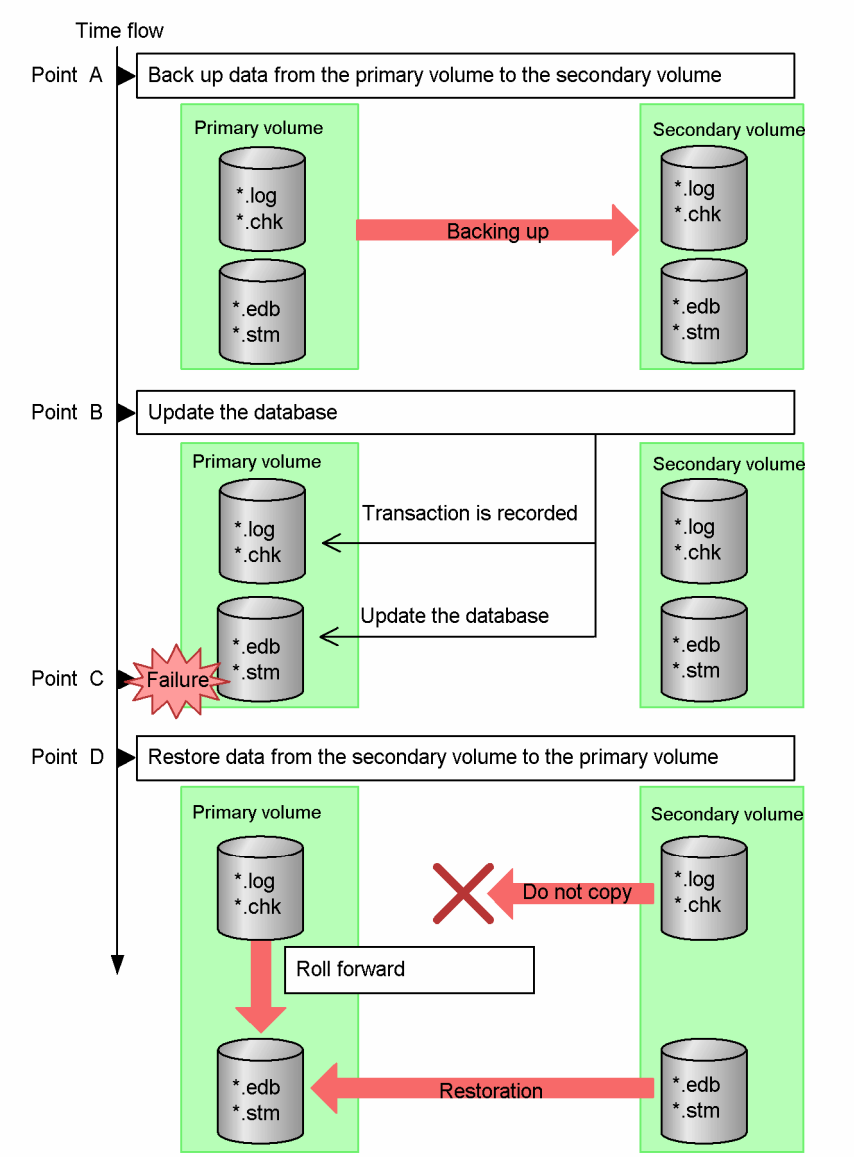


Figure 8.9 Flow of Operation Using Transaction Logs

In Figure 8.9, after the database is backed up from the primary volume to the secondary volume at point A, the database is updated at point B. If an error occurs in a data file on the primary volume at point C and then the database is restored from the secondary volume to the primary volume, the database returns to the state it was in at point A. In this case, if the transaction logs from when the backup was executed up until when the restoration command was executed are stored on the primary volume, the database can be recovered to the state it was in immediately before the error occurred (state immediately before point C) by executing roll-forward.

8.5.1 System Configuration

This example describes the procedure for restoring only the data in the mailbox store (Information Store) out of the backup data obtained by VSS from the secondary volume to the primary volume, and rolling forward transaction logs. (If restoration is executed in units of Information Stores, you must roll forward transaction logs at the time of restoration.) The system configuration described in this operation example is shown Figure 8.10.

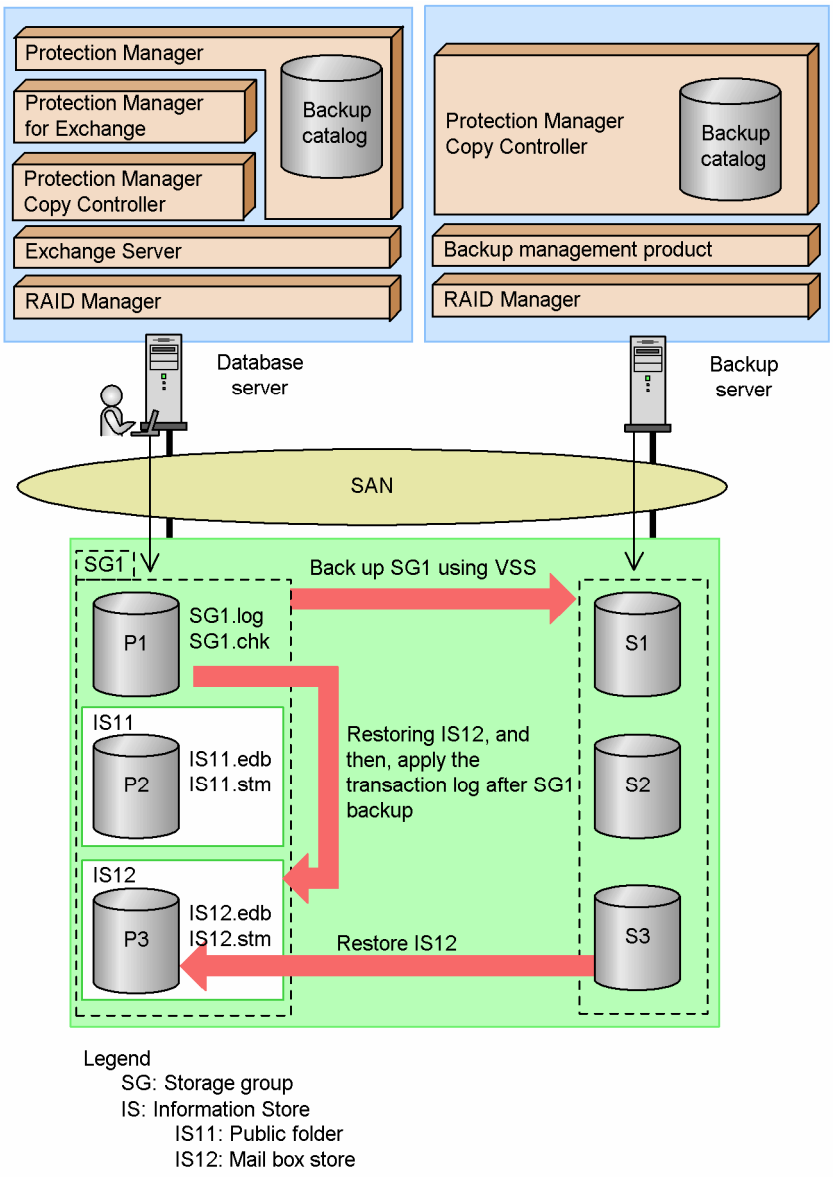


Figure 8.10 System Configuration for Restoring Data in Units of Information Stores by Applying Transaction Logs

The prerequisites for this example are the same as for backup and restoration using VSS plus the prerequisites that follow:

- Only the data files (`*.edb` and `*.stm`) in the Information Store are stored on the same disk.
- The transaction log file on the primary volume is not destroyed (roll-forward is executable).
- Storage group `SG1` has been backed up from the primary volume to the secondary volume using VSS.

8.5.2 Rolling Forward the Transaction Logs At the Time of Restoration

This section describes the procedure for restoring only a specific Information Store in the storage group that was backed up from the primary volume to the secondary volume using VSS, and rolling forward transaction logs.

To perform restoration of data from a VSS backup, the Protection Manager service must be running on the backup server.

To roll forward transaction logs at the time of restoration:

1. Check the backup ID and Information Store name of the backup data.

   To check the backup ID and Information Store name, execute the `drmexgcat` command. The backup ID is displayed in `BACKUP-ID` and the Information Store name is displayed in `INFORMATIONSTORE`. Since this example restores the mailbox store, check the Information Store name of the item for which `OBJECT` is `MAILBOXSTORE`.

```
DBServer > drmexgcat -target SG1
STORAGEGROUP: SG1
BACKUP-ID:0000000001 BACKUP-MODE: VSS STORAGEGROUP: ST1 ORIGINAL-ID:0000000001
START-TIME:2005/06/01 10:30:00 END-TIME: 2005/06/01 10:31:00 HOSTNAME:DBServer
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE     COPY-GROUP
M MAILBOXSTORE IS12(SG1)            E:\SG1\IS12.edb        G:          -       Harddisk1
VG01,vol03
- -          -                  -                  -         -      -          VG01,vol03
M MAILBOXSTORE IS12(SG1)            E:\SG1\IS12.stm        G:          -       Harddisk1
VG01,vol03
- -          -                  -                  -         -      -          VG01,vol03
P PUBLICSTORE  IS11(SG1)            E:\SG1\IS11.edb        F:          -       Harddisk1
VG01,vol02
- -          -                  -                  -         -      -          VG01,vol02
P PUBLICSTORE  IS11(SG1)            E:\SG1\IS11.stm        F:          -       Harddisk1
VG01,vol02
- -          -                  -                  -         -      -          VG01,vol02
T TRANSACT     -                    E:\SG1\E00.log         E:          -       Harddisk1
VG01,vol01
- -          -                  -                  -         -      -          VG01,vol01
C CHECKPOINT   -                    E:\SG1\E00.chk         E:          -       Harddisk1
VG01,vol01
- -          -                  -                  -         -      -          VG01,vol01
```

2. Create a backup ID record file.

   Execute `EX_DRM_BACKUPID_SET` with the backup ID specified to create a backup ID record file. Here, use operation ID `Operation_A`.

```
DBServer > EX_DRM_BACKUPID_SET Operation_A -backup_id 0000000001
```

3. Specify the Information Store name to restore the Information Store from the secondary volume to the primary volume and roll forward the transaction logs.

   Resynchronize the primary volume and secondary volume and then restore the data. To restore the data, execute `EX_DRM_EXG_RESTORE` on the database server.

   To execute roll-forward, specify the `-recovery` option. To specify the Information Store name, specify the `-target` option.

```
DBServer > EX_DRM_EXG_RESTORE Operation_A -resync -recovery -target SG1\IS12
```

8.6 Backing up and Restoring Storage Groups Between Remote Sites (Remote Copy)

This section provides an example of backing up storage groups on Exchange Server from a primary volume in the local site to a secondary volume in the remote site, and an example of restoring storage groups from a secondary volume in the remote site to a primary volume in the local site, using the volume replication functionality such as TrueCopy or UR, which allows you to copy data between subsystems.

The following examples are provided:

- Backing up storage groups to a remote site
- Restoring the storage groups backed up in the remote site to the local site (if the local site is operating properly)

The examples in this section assume the system configuration shown in the Figure 8.11.

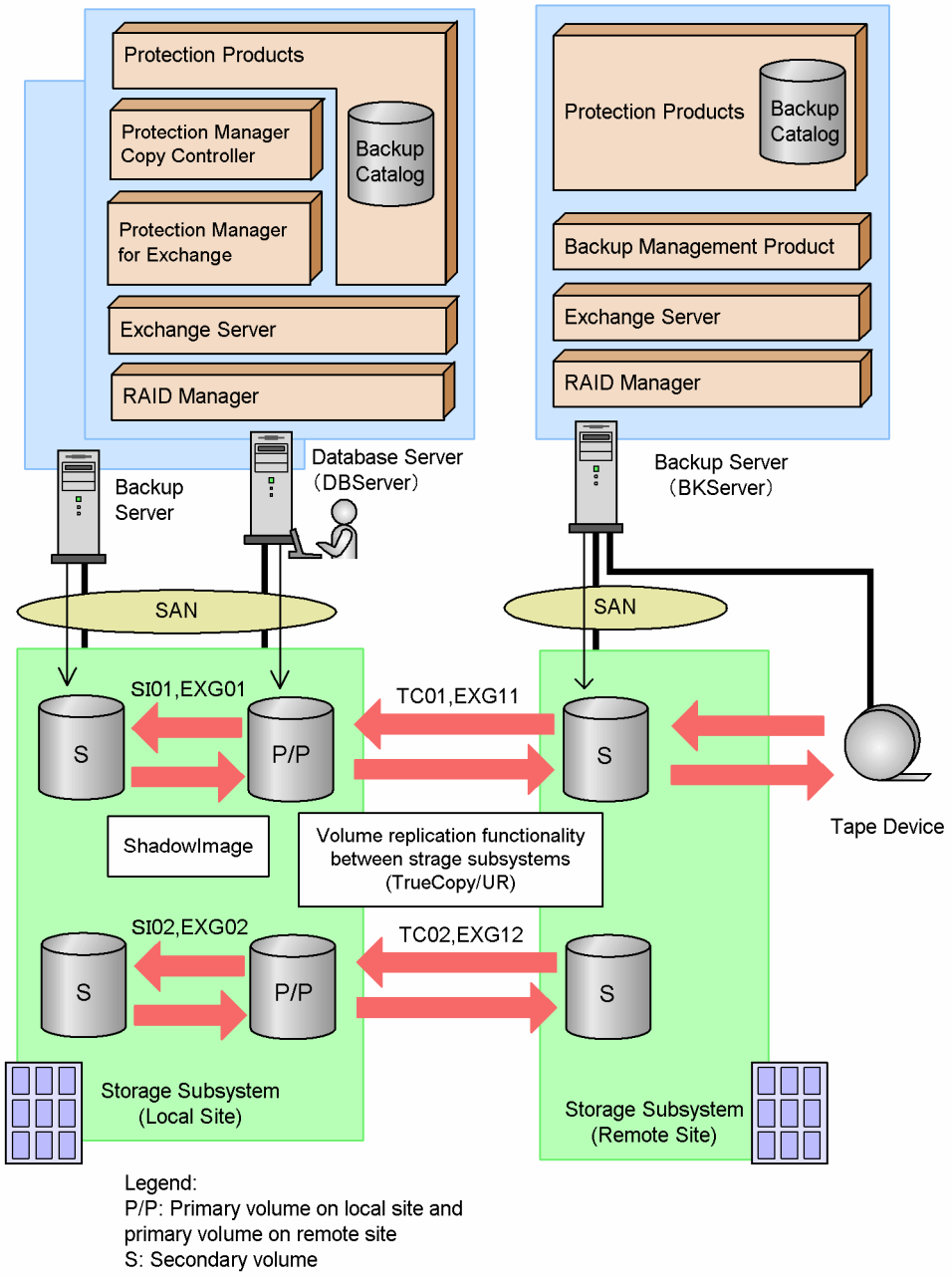


Figure 8.11 System Configuration for Backing up and Restoring Storage Groups Between Remote Sites

The examples in this section assume the following conditions:

- The system has a database server (`DBServer`) in the local site and a backup server (`BKServer`) in the remote site connected to a tape device.
- The FTP service on the backup server in the remote site is started, and the system is set up so that a user who has logged onto the OS of each file server can log into the FTP server and transfer files to the FTP server. The FTP user ID is `admin`, and the FTP user password is `password`.
- As Exchange Server has been installed on the backup server (`BKServer`) in the remote site, if the storage subsystem in the local site fails, Exchange Server can be used as a database server.
- The primary volume in the local site is mounted on the `D:` drive and `E:` drive of `DBServer`.
- TrueCopy volume is configured as a pair (usual operation is in the pair state) with the copy group names `TC01,EXG11` and `TC02,EXG12`.
- A storage group `STR1` exists on the database server (`DBServer`) in the local site, and services are running.
- The volume (usual operation is in the pair state) `TC01,EXG11` for TrueCopy or UR contains the data files for the storage group `STR1`.
- The volume (usual operation is in the pair state) `TC02,EXG12` for TrueCopy or UR contains the transaction log files and checkpoint files for the storage group `STR1`.
- ShadowImage volume is configured as a pair with the copy group name `SI01,EXG01` and `SI02,EXG02`.
- The system is set up so that the generation identifier for the backup destination in the remote site is `remote_0`.
- The volume in the remote site is not normally mounted. If required, it will be mounted to the `D:` drive and the `E:` drive of the backup server.
- Use `Operation_A` as the operation ID.

8.6.1 Backing up Storage Groups to a Remote Site

This section explains how to back up storage groups to a remote site. The procedure for restoring the data to a local site and the procedure for restoring the data to a remote site is explained separately.

8.6.1.1 Operations at the Local Site

1. Check the volume on which the database will be backed up and the generation identifier.

   Execute the `drmexgdisplay` command on the database server.

```
DBServer > drmexgdisplay -cf
STORAGEGROUP: STR1
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE      COPY-GROUP
COPY-FUNC    GEN-NAME
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.edb      D:          -       Harddisk1
SI01,EXG01      ShadowImage local_0
- -          -                 -                  -         -      -          SI01,EXG01
ShadowImage local_0
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.edb      D:          -       Harddisk1
TC01,EXG11      TrueCopy    remote_0
- -          -                 -                  -         -      -          TC01,EXG11
TrueCopy    remote_0
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.stm      D:          -       Harddisk1
SI01,EXG01      ShadowImage local_0
- -          -                 -                  -         -      -          SI01,EXG01
ShadowImage local_0
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.stm      D:          -       Harddisk1
TC01,EXG11      TrueCopy    remote_0
- -          -                 -                  -         -      -          TC01,EXG11
TrueCopy    remote_0
T TRANSACT     -                    E:\STR1\E00.log        E:          -       Harddisk2
SI02,EXG02      ShadowImage local_0
- -          -                 -                  -         -      -          SI02,EXG02
ShadowImage local_0
T TRANSACT     -                    E:\STR1\E00.log        E:          -       Harddisk2
TC02,EXG12      TrueCopy    remote_0
- -          -                 -                  -         -      -          TC02,EXG12
TrueCopy    remote_0
T TRANSACT     -                    E:\STR1\E000001.log    E:          -       Harddisk2
SI02,EXG02      ShadowImage local_0
- -          -                 -                  -         -      -          SI02,EXG02
ShadowImage local_0
T TRANSACT     -                    E:\STR1\E000001.log    E:          -       Harddisk2
TC02,EXG12      TrueCopy    remote_0
- -          -                 -                  -         -      -          TC02,EXG12
TrueCopy    remote_0
C CHECKPOINT   -                    E:\STR1\E00.chk        E:          -       Harddisk2
SI02,EXG02      ShadowImage local_0
- -          -                 -                  -         -      -          SI02,EXG02
ShadowImage local_0
C CHECKPOINT   -                    E:\STR1\E00.chk        E:          -       Harddisk2
TC02,EXG12      TrueCopy    remote_0
- -          -                 -                  -         -      -          TC02,EXG12
TrueCopy    remote_0
DBServer >
```

2. Back up the database to the secondary volume in the remote site.

   To back up the database to the secondary volume in the remote site, specify the generation identifier for the back up destination `remote_0` on the database server, and then execute `EX_DRM_EXG_BACKUP`.

```
DBServer > EX_DRM_EXG_BACKUP Operation_A -rc remote_0
```

3. Check that the storage group has been backed correctly.

   To check that backup information on which the storage group had been backed up is available, execute the `drmexgcat` command on the database server.

```
DBServer > drmexgcat
STORAGEGROUP: STR1
BACKUP-ID:0000000001 BACKUP-MODE: COLD STORAGEGROUP: STR1 ORIGINAL-ID:0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:01:00 HOSTNAME:DBServer
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE     COPY-GROUP
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.edb      D:          -       Harddisk1
TC01,EXG11
- -          -                 -                   -         -      -          TC01,EXG11
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.stm      D:          -       Harddisk1
TC01,EXG11
- -          -                 -                   -         -      -          TC01,EXG11
T TRANSACT     -                   E:\STR1\E00.log        E:          -       Harddisk2
TC02,EXG12
- -          -                 -                   -         -      -          TC02,EXG12
T TRANSACT     -                   E:\STR1\E000001.log    E:          -       Harddisk2
TC02,EXG12
- -          -                 -                   -         -      -          TC02,EXG12
C CHECKPOINT   -                   E:\STR1\E00.chk        E:          -       Harddisk2
TC02,EXG12
- -          -                 -                   -         -      -          TC02,EXG12
DBServer >
```

4. Export the backup information to temporary files.

   To back up data from the secondary volume to a tape, you must copy information about backups that were performed from the primary volume to the secondary volume. Copy this information to the backup server. To export information about backup to temporary files, execute `EX_DRM_DB_EXPORT`. The temporary files are stored in the temporary directory for extended commands, on the database server.

```
DBServer > EX_DRM_DB_EXPORT Operation_A
```

5. Transfer the temporary files to the backup server.

   Batch-transfer the temporary files from the database server to the backup server. The temporary files are stored in the temporary directory for extended commands, on the backup server.

```
DBServer > EX_DRM_FTP_PUT Operation_A -server BKServer -user admin -password password
```

   Note: The data can be transferred using the pair function of TrueCopy or UR, without using `EX_DRM_FTP_PUT`.

8.6.1.2 Operations at the Remote Site

1. Import the temporary files, which were transmitted from the database server in the local site, into the backup catalog on the backup server.

   To import the temporary files, execute `EX_DRM_DB_IMPORT` on the backup server.

```
BKServer > EX_DRM_DB_IMPORT Operation_A
```

2. Check that the temporary file has been imported correctly.

   To check that backup information bearing the date on which the storage groups had been backed up from the primary volume to the secondary volume has been imported into the backup server, execute the `drmexgcat` command on the backup server.

```
BKServer > drmexgcat
STORAGEGROUP: STR1
BACKUP-ID:0000000001 BACKUP-MODE: COLD STORAGEGROUP: STR1 ORIGINAL-ID:0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:01:00 HOSTNAME:DBServer
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE      COPY-GROUP
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.edb      D:          -       Harddisk1
TC01,EXG11
- -          -                   -                 -         -      -          TC01,EXG11
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.stm      D:          -       Harddisk1
TC01,EXG11
- -          -                   -                 -         -      -          TC01,EXG11
T TRANSACT     -                    E:\STR1\E00.log        E:          -       Harddisk2
TC02,EXG12
- -          -                   -                 -         -      -          TC02,EXG12
T TRANSACT     -                    E:\STR1\E000001.log    E:          -       Harddisk2
TC02,EXG12
- -          -                   -                 -         -      -          TC02,EXG12
C CHECKPOINT   -                    E:\STR1\E00.chk        E:          -       Harddisk2
TC02,EXG12
- -          -                   -                 -         -      -          TC02,EXG12
BKServer >
```

3. Back up the data on the secondary volume onto tape.

   To back up the data, execute `EX_DRM_TAPE_BACKUP` on the backup server. The command specifies the `D:` drive as the mount point for tape backup.

```
BKServer > EX_DRM_TAPE_BACKUP Operation_A -mount_pt D:
```

   In this example, when you execute the backup, information relating to this backup operation is registered in the backup catalog with the new backup ID `0000000002`.

4. Check that the storage group has been backed to tape correctly.

   To check that backup information bearing the date on which the storage group had been backed up from the primary volume to the secondary volume is available, execute the `drmtapecat` command on the database server.

```
BKServer > drmtapecat 0000000002
BACKUP-ID   BACKUP-OBJECT    SNAPSHOT TIME        EXPIRATION TIME      BACKUP-MEDIA
0000000002  MSEXCHANGE       2004/08/05 10:00:00  2004/11/05 10:00:00  MEDIA1
BKServer >
```

8.6.1.3 Operations at the Local Site

Resynchronize the copy groups by executing `EX_DRM_RESYNC` on the database server:

```
DBServer > EX_DRM_RESYNC Operation_A
```

8.6.2 Restoring a Storage Group to Local Site

This section explains how to restore a storage group to a local site when the local site is operating normally without any failures. The procedures for restoring the data to a local site and to a remote site are explained separately.

8.6.2.1 Operations at the Remote Site

To restore a storage group to a local site:

1. Split the TrueCopy pair (usual operation is in the pair state).

   ```
   BKServer > pairsplit -g TC01 -d EXG11 -rw
   BKServer > pairsplit -g TC02 -d EXG12 -rw
   BKServer >
   ```

2. Check the backup ID of the backup data.

   To check the backup ID, execute the `drmtapecat` command on the backup server.

   ```
   BKServer >drmtapecat -l
   BACKUP-ID ORIGINAL-ID BACKUP-OBJECT INSTANCE SNAPSHOT TIME   EXPIRATION TIME  BACKUP-MEDIA
   0000000002 0000000001 MSEXCHANGE    default  2004/08/05 10:00:00 2004/11/05 10:00:00  -
   :
   :
   BKServer >
   ```

3. Restore the backed up data from the tape to the secondary volume.

   To restore the data, execute `EX_DRM_TAPE_RESTORE` on the backup server. The command specifies the `D:` drive as the mount point.

   ```
   BKServer > EX_DRM_TAPE_RESTORE Operation_A -backup_id 0000000002 -mount_pt D:
   ```

   When you execute restore, information about this restore is registered in the backup catalog of the backup server with the new backup ID `0000000003`.

4. Check that the storage group has been stored from the tape to the second volume correctly.

   To check that backup information bearing the date on which the storage group had been backed up from the primary volume to the second volume has been restored, execute the `drmexgcat` command on the backup server.

```
BKServer > drmexgcat
STORAGEGROUP: STR1
BACKUP-ID:0000000003 BACKUP-MODE: COLD STORAGEGROUP: STR1 ORIGINAL-ID:0000000001
START-TIME:2004/08/05 10:00:00 END-TIME: 2004/08/05 10:01:00 HOSTNAME:DBServer
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE     COPY-GROUP
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.edb      D:          -       Harddisk1
TC01,EXG11
- -          -                  -                  -         -      -          TC01,EXG11
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.stm      D:          -       Harddisk1
TC01,EXG11
- -          -                  -                  -         -      -          TC01,EXG11
T TRANSACT     -                    E:\STR1\E00.log        E:          -       Harddisk2
TC02,EXG12
- -          -                  -                  -         -      -          TC02,EXG12
T TRANSACT     -                    E:\STR1\E000001.log    E:          -       Harddisk2
TC02,EXG12
- -          -                  -                  -         -      -          TC02,EXG12
C CHECKPOINT   -                    E:\STR1\E00.chk        E:          -       Harddisk2
TC02,EXG12
- -          -                  -                  -         -      -          TC02,EXG12
BKServer >
```

5. Export the backup information to temporary files.

   To restore from the secondary volume to the primary volume, you must first identify the backup information related to the restore operation from the tape to the secondary volume and copy that information onto the file server. To export the backup information into temporary files in the temporary directory for extended commands, execute `EX_DRM_DB_EXPORT`.

```
BKServer > EX_DRM_DB_EXPORT Operation_A
```

8.6.2.2 Operations at the Local Site

To restore a storage group to a local site:

1. Receive the temporary files on the database server in the local site.

   To batch-receive the temporary files held on the backup server onto the database server, execute `EX_DRM_FTP_GET` on the database server. The temporary files are stored in the temporary directory for extended commands, on the database server.

```
DBServer > EX_DRM_FTP_GET Operation_A -server BKServer -user admin -password password
```

   Note that in this procedure, the data can be transferred using the pair function of TrueCopy or UR, without using `EX_DRM_FTP_GET`.

2. Import the temporary files, which were transmitted from the database server, into the backup catalog on the database server.

   To import the temporary files, which were transmitted from the database server, into the backup catalog on the database server, execute `EX_DRM_DB_IMPORT`.

```
DBServer >EX_DRM_DB_IMPORT Operation_A
```

3. Check that the temporary file has been imported correctly.

   To check that backup information bearing the date on which the storage groups had been backed up from the primary volume to the secondary volume has been imported into the database server, execute the `drmexgcat` command on the database server.

```
DBServer > drmexgcat
STRAGEGROUP: STR1
BACKUP-ID:0000000003 BACKUP-MODE: COLD STRAGEGROUP: STR1 ORIGINAL-ID:0000000001
START-TIME:2002/06/01 10:00:00 END-TIME: 2004/08/05 10:01:00 HOSTNAME:DBServer
T OBJECT     INFORMATIONSTORE    FILE              FS        DG     DEVICE     COPY-GROUP
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.edb      D:          -       Harddisk1
TC01,EXG11
- -          -                 -                 -         -      -          TC01,EXG11
M MAILBOXSTORE MAILBOX1(STR1)       D:\STR1\priv1.stm      D:          -       Harddisk1
TC01,EXG11
- -          -                 -                 -         -      -          TC01,EXG11
T TRANSACT     -                    E:\STR1\E00.log        E:          -       Harddisk2
TC02,EXG12
- -          -                 -                 -         -      -          TC02,EXG12
T TRANSACT     -                    E:\STR1\E000001.log    E:          -       Harddisk2
TC02,EXG12
- -          -                 -                 -         -      -          TC02,EXG12
C CHECKPOINT   -                    E:\STR1\E00.chk        E:          -       Harddisk2
TC02,EXG12
- -          -                 -                 -         -      -          TC02,EXG12
DBServer >
```

4. Restore data from the secondary volume to the primary volume.

   Resynchronize the primary volume and secondary volume and then restore the data. To restore the data, execute `EX_DRM_EXG_RESTORE` on the database server.

```
DBServer > EX_DRM_EXG_RESTORE Operation_A -resync
```

5. Resynchronize the copy groups.

   To resynchronize the copy groups, execute `EX_DRM_RESYNC` on the database server.

```
DBServer > EX_DRM_RESYNC Operation_A
```

8.7 Using a Backup Management Product That Cannot Link with Protection Manager

If a backup management product that is unable to link with Protection Manager is used, follow the steps below to back up storage groups to tape and restore them from tape. For more details on steps 1 to 6 in the backup procedure, see section 8.3.2. For more details on the restoration procedure, see section 8.3.3.

To back up storage groups to tape (where a backup management product that is unable to link with Protection Manager is used):

1. When usual operation is in the split state, execute `EX_DRM_CACHE_PURGE` to clear the secondary volume cache.
2. When usual operation is in the split state, execute `EX_DRM_RESYNC` to resynchronize the copy group.
3. Execute `EX_DRM_EXG_BACKUP` to back up the file system to the secondary volume.
4. Execute `EX_DRM_DB_EXPORT` to export the backup information to temporary files.
5. Execute `EX_DRM_FTP_PUT` to transfer the temporary files to the backup server.
6. Execute `EX_DRM_DB_IMPORT` to import the temporary files transferred from the database server to the backup catalog on the backup server.
7. Check the backup ID.

   To check the backup ID, open the backup ID record file (*`operation-ID`*`.bid`) in the following directory on the backup server:

   `FTP_HOME_DIR-value``\`*`FTP_SUB_DIR-value`*`\`*`operation-ID`*`\BK\`
   `operation-ID``.bid`
8. Check the storage location of the data files.

   To make this check, execute the `drmmount` command with the backup ID determined in step 7 specified as an argument.

   The command results show the mounted drive name. The mounted drive contains the data files (`*.edb` or `*.stm`), transaction log files (`*.log`), and checkpoint files (`*.chk`) of the storage group. If necessary, execute the `drmexgcat` command to check the contents of the file to be backed up.
9. Use a backup management product to back up all the files in the mounted drive to tape.
10. Execute the `drmumount` command to unmount the mount point mounted in step 8.

To restore a storage group from a tape (where a backup management product that is unable to link with Protection Manager is used):

1. Execute the drmexgcat command to identify the backup ID for the backup catalog to be restored.
2. Execute the `drmmount` command, and then restore the backup from tape to the storage location checked in backup steps 7 and 8.
3. Execute `EX_DRM_BACKUPID_SET` by specifying the backup ID to create a backup ID record file.
4. Carry out the procedures following step 4 described in section 8.3.3 to restore the backup from the secondary volume to the primary volume.

8.8 Using File Sharing to Link the Database Server and Backup Server

When backup information is transferred between a database server and a backup server, extended commands (`EX_DRM_FTP_PUT` or `EX_DRM_FTP_GET`) are normally used to transfer the information in an FTP operation. Instead of FTP, however, file sharing can also be used to back up and restore storage groups. The following sections describe examples of backing up or restoring `storage groups` by using a file share.

8.8.1 Preparing to Back up or Restore Storage Groups by Using a File Share

To prepare to back up and restore storage groups by using a file share:

1. Check the temporary directory for extended commands that was created automatically by the check tool for operation definition files.

   For more information about where the temporary directory for extended commands is created automatically, see section 4.15.9.

2. Running Explorer or any other tool, share the temporary directory for extended commands that is on the backup server.

3. Running Explorer or any other tool on the database server, assign the shared temporary directory for extended commands that is on the backup server as a network drive. For now, assign it to `X:` on the database server.

4. Set access permissions for the automatically generated temporary directory for extended commands that is on the backup server, to make it accessible to users who are executing extended commands from the database server to read or write to and from it.

8.8.2 Example of Using a File Share

This section explains how to back up storage groups by using a file share. For more operational details on steps 1 to 4 and step 6 and subsequent steps, see section 8.3.2.

To back up storage groups by using a file share:

1. When usual operation is in the split state, execute `EX_DRM_CACHE_PURGE` to clear the secondary volume cache.

2. When usual operation is in the split state, execute `EX_DRM_RESYNC` to resynchronize the copy group.

3. Execute `EX_DRM_EXG_BACKUP` to back up the storage group to the secondary volume.

4. Execute `EX_DRM_DB_EXPORT` to export the backup information to temporary files.

5. Copy the temporary files from the database server to the backup server.

   To copy the temporary files from the temporary directory for extended commands that is on the database server to the temporary directory for extended commands that is on the backup server, execute a copy command on the database server.

```
DBServer > copy /y
temporary-directory-for-extended-commands-on-the-database-server\Operation_A\DB\Operati
on_A.drm X:\
```

6. Execute EX_DRM_DB_IMPORT to import the temporary files transferred from the database server to the backup catalog on the backup server.
7. Execute EX_DRM_TAPE_BACKUP to back up the data on the secondary volume to tape.
8. When usual operation is in the pair state, execute EX_DRM_CACHE_PURGE to clear the secondary volume cache.
9. When usual operation is in the pair state, execute EX_DRM_RESYNC to resynchronize the copy group.

8.8.3 Example of Restoring Storage Groups by Using a File Share

This section explains how to restore storage groups on the backup server by using a file share. For more operational details on steps 1 to 3 and step 5 and subsequent steps, see section 8.3.3.

To restore storage groups on the backup server by using a file share:

1. Execute the drmtapecat command to check the backup ID of the backup data.
2. Execute EX_DRM_TAPE_RESTORE to restore the backup data from the tape to the secondary volume.
3. Execute EX_DRM_DB_EXPORT to import the temporary files transferred from the backup server to the backup catalog on the database server.
4. Copy the temporary files from the backup server to the database server.

   To copy the temporary files from the temporary directory for extended commands that is on the database server to the temporary directory for extended commands that is on the backup server, execute the copy command on the database server.

```
DBServer > copy /y X:\Operation_A.drm
temporary-directory-for-extended-commands-on-the-database-server\Operation_A\DB
```

5. Execute EX_DRM_DB_IMPORT to import the temporary files transferred from the backup server to the backup catalog on the database server.
6. Execute EX_DRM_EXG_RESTORE to restore the secondary volume data to the primary volume.

8.9 Deleting Backup Data

If the backup data in the secondary volume is no longer necessary, you may delete it. Delete backup data after the data has been backed up from a secondary volume to tape or after the data has been restored from a secondary volume to the primary volume.

To delete the backup data, execute the `drmresync` command, specifying the backup ID of the data to be deleted:

```
PROMPT> drmresync -backup_id backup-ID
KAVX0001-I The drmresync command will now start.
KAVX0002-I The drmresync command will now end.
PROMPT>
```

Note: When you execute this command, the primary volume and the secondary volume are resynchronized and mirrored. Now the backup information of the specified backup ID is deleted from the backup catalog.

Chapter 9 Troubleshooting

This chapter explains how to deal with problems that may occur during the operation of Protection Manager.

- Addressing Problems (see section 9.1)
- Troubleshooting Extended Commands (see section 9.2)
- Collecting Data When a Problem Occurs (see section 9.3)
- Collecting Required Data (see section 9.4)
- Investigating Collected Data (see section 9.5)
- Troubleshooting Dictionary Map File Errors (see section 9.6)
- Calling the Hitachi Data Systems Support Center (see section 9.7)

9.1 Addressing Problems

To deal with problems that may occur during the operation of Protection Manager follow the steps below. If a problem occurs during operation using an extended command, see section 9.2.

1. Check the conditions in which the problem occurred. Check the operating conditions and output messages corresponding to when the problem occurred.

   For details on how to respond to each message, see the *HiCommand Protection Manager Messages* manual.

2. Collect log information and other data that can be used to identify the primary cause. Collect various items of log information output from Protection Manager and other data such as environment setup information.

   For details on the data to collect and how to collect it, see sections 9.3 and 9.4.

3. Based on the data collected, investigate the cause and analyze the problem. Use the collected data to investigate the cause of the problem and determine where the problem occurred. If you still cannot understand the cause of the problem after this step, contact the dealer from which you purchased the product.

9.2 Troubleshooting Extended Commands

If a problem occurs during operation using an extended command, a message that indicates an error has occurred in the extended command is output to `drm_script.log` (the *extended command trace log*), which is the log file that is used for extended commands. In such a case, correct the error by using the procedure described below.

To handle an extended command error:

1. Check the problem status.

   Refer to the extended command trace log file `drm_script.log` (or `drm_script.log.old`) to check the operating status that existed at the time when the problem with the extended command occurred, and to check the message contents.

   For details on the output destination of the extended command trace log, see section 9.3. For details on the message contents and message troubleshooting, see the *HiCommand Protection Manager Messages* manual.

2. Check the problem in the Protection Manager log file. When an error occurs in a Protection Manager command executed within an extended command, an error message is output to the Protection Manager log file `drm_output.log` (or `drm_output.log.old`). See the content of the Protection Manager log file regarding the time when the error message for the extended command was output, and check the message and the operation status from the time the problem with the Protection Manager command occurred.

3. Check the OS log file. If no error message was output to `drm_output.log` (or `drm_output.log.old`), see the OS log file, and then check the operating status that existed at the time when the problem occurred.

9.3 Collecting Data When a Problem Occurs

This section describes the data that needs to be collected to determine the cause of a problem.

9.3.1 Data To Be Collected (in Windows)

9.3.1.1 Collecting Windows OS Log Information

Collect the Windows OS log information shown in Table 9.1.

Table 9.1 OS Log Information for Windows Systems

Information type	Summary	Default file name
System log	Windows event log[1]	There is no corresponding file name.
System files	`hosts` file	*system-directory*`\system32\drivers\etc\hosts`
	`services` file	*system-directory*`\system32\drivers\etc\` `services`
	Windows system information	There is no corresponding file name.
Information about the OS and installed product	▪ Names of the OS and installed product ▪ Version information ▪ Patch information	There is no corresponding file name.
Environment setting information	Environment variable	There is no corresponding file name.
Disk configuration information	Configuration information for logical volumes	There is no corresponding file name.
Dump information	`Dr. Watson` log file[2]	▪ Windows 2000 Server *system-drive*`\Documents and Settings\All Users\Documents\DrWatson\drwtsn32.log` ▪ Windows Server 2003 *system-drive*`\Documents and Settings\`*user-name*[#3]`\Local Settings\Application Data\Microsoft\Dr Watson\drwtsn32.log`

Information type	Summary	Default file name
	Memory dump[4]	▪ Windows 2000 Server *system-drive*\Documents and Settings\All Users\Documents\DrWatson\user.dmp ▪ Windows Server 2003 *system-drive*\Documents and Settings*user-name*[3]\Local Settings\Application Data\Microsoft\Dr Watson\user.dmp
Cluster control software information	Definition file and log file	There is no corresponding file name.
	CHKDSK execution logs	*system-directory*\cluster\ChkDsk_*.log

[1]Use Windows Event Viewer to collect files saved in text or CSV format.

[2] If these files are set to be output to another folder, collect them from the appropriate folder.

[3] The name of the user that executed the Protection Manager command is applied.

[4]If you have set up the system so that dump data is output to other files, collect the applicable data.

9.3.1.2 Protection Manager Information

Collect information pertaining to Protection Manager, as shown in Table 9.2. Also, in cases where the problem occurred over a network connection, you will need to collect files from the connected machine.

Messages relating to execution of Protection Products commands (start, stop, warning, and error messages) are also output to the Windows event log.

Table 9.2 Protection Manager Information

Information type	Summary	Default files
Version information	Protection Manager version information file	All of the following corresponding files: *Protection-Manager-installation-folder*\.version*
Environment settings	Files for each type of Protection Manager environment setting	All of the files in the following directory: *Protection-Manager-installation-folder*\conf\
Standard output logs	Protection Manager log files[1]	*Protection-Manager-installation-folder*\log\drm_output.log *Protection-Manager-installation-folder*\log\drm_output.log.old

Information type	Summary	Default files
Extended command trace logs	Protection Manager trace log files for extended commands[1]	*Protection-Manager-installation-folder*\script\log\drm_script.log *Protection-Manager-installation-folder*\script\log\drm_script.log.old
Detailed trace logs	Protection Manager trace log files[2]	*Protection-Manager-installation-folder*\log\drm_pp_trace[1-16].log
Installation trace logs	Protection Manager installation trace log files	*Protection-Manager-installation-folder*\log\drm_inst.log
Trace logs of NetBackup	Protection Manager log files[3]	*Protection-Manager-installation-folder*\log\drm_nbu_backup.log *Protection-Manager-installation-folder*\log\drm_nbu_backup.log.old *Protection-Manager-installation-folder*\log\drm_nbu_restore.log *Protection-Manager-installation-folder*\log\drm_nbu_restore.log.old
Internal operation information	Temporary files	All of the files in the following directory: *Protection-Manager-installation-folder*\tmp
Dictionary map	Dictionary map files[4]	All of the files in the following directory: *Protection-Manager-installation-folder*\db

[1] When the size of these log files reaches 10 MB, a file with `.old` attached to the end of the file name is created. (For `drm_output.log`, the file to be created is `drm_output.log.old`.) If a file whose name ends with `.old` already exists, it is overwritten.

[2] When the size of these log files reaches 10 MB, files given a number from 1 to 16 for its file name are created.

[3] When the `drmtapebackup` command or `drmmediabackup` command is executed and the backup command of NetBackup is executed, the result of command operation is output to `drm_nbu_backup.log`. When the size of the `drm_nbu_backup.log` file reaches 10 MB, the contents of the file are moved to `drm_nbu_backup.log.old`. If `drm_nbu_backup.log.old` already exists, it is overwritten.

When the `drmtaperestore` command or `drmmediarestore` command is executed and the restoration command of NetBackup is executed, the result of command operation is output to `drm_nbu_restore.log`. When the size of the `drm_nbu_restore.log` file reaches 10 MB, the contents of the file are moved to `drm_nbu_restore.log.old`. If `drm_nbu_restore.log.old` already exists, it is overwritten.

[4]If you have set other directories in `DRM_DB_PATH` of the Protection Manager configuration definition file (`init.conf`), collect data from the applicable directories as well.

9.3.1.3 RAID Manager Information

Collect information pertaining to RAID Manager, as shown in Table 9.3. Also, in cases where the problem occurred over a network connection, you will need to collect files from the connected machine.

Table 9.3 RAID Manager Information

Information type	Summary	Default files
Environment settings	Files for each type of RAID Manager environment setting[1]	All of the following corresponding files: *system-directory*\horcm*
Log information	Each type of RAID Manager log files[2]	All of the following corresponding files: *system-drive*\HORCM\log*
RAID configuration information	Execution result of the RAID Manager `inqraid`, `raidscan` or `pairdisplay` command	There is no corresponding file name.

[1]If you have specified other files, collect the applicable data.

[2]If you have set up the system so that log files are output to other directories, collect data from the applicable directories.

9.3.1.4 Collecting Database Information

Collect the following information about the database (Table 9.4). In cases where the problem occurred over a network connection, you will need to collect files from the connected machine.

Table 9.4 Database Information (When the Backup is of a SQL Server Database or Exchange Database)

Information type	Summary	Default file
Definition information	Definition file	There is no corresponding file name.
Log information	Log file	There is no corresponding file name.

9.3.1.5 Information Pertaining to Protection Manager Console

Collect the log information pertaining to Protection Manager Console listed in the following table.

Table 9.5 Information Pertaining to Protection Manager Console

Information type	Summary	Default file
Install file list	List of all files including hidden and read only files in Protection Manager Console	--
Version information	Version information files in Protection Manager Console	The following corresponding file: *Protection-Manager-Console-installation-directory*\.version_con
	Java version information	--
Registry information	Registry information in Protection Manager Console	--
	Java registry information	--
Environment settings	Various types of environment setup files in Protection Manager Console	All files in the following directory: *Protection-Manager-Console-installation-directory*\conf\
Trace log information	Trace log files in Protection Manager Console	All files in the following directory: *Protection-Manager-Console-installation-directory*\log\
Internal operation information	Temporary files	All files in the following directories: ▪ *Protection-Manager-Console-installation-directory*\status\ ▪ *Protection-Manager-Console-installation-directory*\tmp\ ▪ *Protection-Manager-Console-installation-directory*\work\
	Information files for backup job management in Protection Manager Console	All files in the following directory: *Protection-Manager-Console-installation-directory*\schedule\

Legend: -- indicates that the information type is not applicable.

9.3.1.6 Operation Details

The following information, pertaining to the operation that was being performed when the problem occurred, also needs to be collected.

- The details of the operation (parameters specified by commands, etc.)
- The date and time when the problem occurred
- The machine configuration (the version of each OS and the host name)
- Whether or not the error is reproducible

9.3.2 Log Files Output by Extended Commands

When an extended command is executed, extended command execution history information and the error status (if an error occurred) are recorded in the log file for extended commands.

The log information is recorded when:

- An extended command started
- An extended command terminated
- A command other than Protection Manager commands (`drm`***xxx***) is executed inside the extended command
- A command other than Protection Manager commands (`drm`***xxx***) terminated successfully inside the extended command
- A command (including Protection Manager commands (`drm`***xxx***)) inside the extended command is executed and then failed

The log file output directory can store up to two generations of log files. The paths of the log files are as follows.

Current log file:

Protection-Manager-installation-directory\script\log\drm_script.log

Previous log file:

Protection-Manager-installation-directory\script\log\
drm_script.log.old

For details on how to respond to a message, indicating an error was encountered during extended command execution, that was output to the extended command execution log file drm_script.log, see section 9.2.

If the number of lines in the current log file exceeds the maximum specified with the MAX_LOG_LINES entry of the host environment settings file, the current log file is backed up. Then, a new log file is generated to continue logging. If the system fails to record information in the log file when an extended command is executed, log information for the extended command is output to the standard output log file drm_output.log.

Note that the log information for Protection Manager commands executed inside the extended command is output to the standard output log file drm_trace.log and the trace log file drm_pp_trace[1-16].log.

Table 9.6 shows the items that are output to the log file for extended commands.

Table 9.6 Items Output to the Log File for Extended Commands

Item	Explanation
Time-stamp	The log output date and time are output (to the millisecond). Format: *yyyy/mm/dd hh:mm:ss.sss*
Log type	The following character string to indicate the log type is output: ▪ START: The extended command started ▪ END: The extended command terminated ▪ INFO: Information ▪ WARNING: Warning ▪ ERROR: Error ▪ DEBUG: Debugging
Operation ID	The operation ID for the executed operation is output.
File server name or database server name	When the backup is of a file system: The name of the file server that was backed up. When the backup is of a SQL Server database: The name of the server containing the SQL server instance that was backed up. When the backup is of a storage group: The character string defined in the specification item DB_SERVER_NAME in the operation definition file is output.
Instance name	When the backup is of a file system: The name of the mount point directory or mount point directory list file that was backed up. When the backup is of a SQL Server database: The name of the SQL server instance that was backed up is output. When the backup is of a storage group: The character string defined in the specification item INSTANCE_NAME in the operation definition file is output.
Extended command's name	The character string that identifies the extended command is output (the extended command's file name except the file extension is output).
Extended command's option	The options specified by the extended command are output.
Log message ID	The ID of the log message is output.
Log message text	The log message is output by the extended command.

The following shows an example of the log file.

When the backup is of a file system:

```
2003/07/31 20:15:11.560 |START     |operation1 |FSServer_A |M: |EX_DRM_FS_BACKUP |KAVX0450-I
The EX_DRM_FS_BACKUP command will now start.
2003/07/31 20:17:22.060 |END       |operation1 |FSServer_A |M: |EX_DRM_FS_BACKUP |The
EX_DRM_FS_BACKUP command will now end.
```

When the backup is of a SQL Server database:

```
2003/07/31 20:15:11.560 |START     |operation1 |DBServer_A |- |EX_DRM_SQL_BACKUP |KAVX0450-I
The EX_DRM_SQL_BACKUP command will now start.
2003/07/31 20:17:22.060 |END       |operation1 |DBServer_A |- |EX_DRM_SQL_BACKUP |KAVX0451-I
The EX_DRM_SQL_BACKUP command will now end.
```

When the backup is of a storage group:

```
2003/07/31 20:15:11.560 |START     |operation1 |DBServer_A |- |EX_DRM_EXG_BACKUP |KAVX0450-I
The EX_DRM_EXG_BACKUP command will now start.
2003/07/31 20:17:22.060 |END       |operation1 |DBServer_A |- |EX_DRM_EXG_BACKUP |KAVX0451-I
The EX_DRM_EXG_BACKUP command will now end.
```

9.4 Collecting Required Data

This section describes how to collect the required data when a problem occurs.

9.4.1 Collecting Data

9.4.1.1 Executing the Data Collection Tool

Execute the data collection tool to collect the data to analyze problems. You can use the Protection Manager data collection tool to collect all Protection Manager information described in section 9.3.1 in a single operation.

The following shows the format of executing the data collection tool:

Format:

```
drm_log.bat log-collection-directory -EVTLOG_CSV
```

The data location can be changed when you execute the tool specifying the desired directory of the log collection. The log collection directory must be specified in a full path. If you want to specify a path including a blank, put it in parentheses. By default, the data is stored in `%TEMP%\drmlog`.

When the tool is executed with the `-EVTLOG_CSV` option specified, the Windows event log outputs in CSV format, and as evt format files. Messages relating to execution of Protection Products commands (start, stop, warning, and error messages) are also output to the Windows event log.

Execute the data collection tool as follows:

```
PROMPT> Protection-Products-installation-destination\bin\util\drm_log.bat
```

After you execute the data collection tool, confirm that the necessary data has been collected.

9.4.1.2 Collecting Windows System Information

In Windows 2000 Server:

From the **Start** menu, click **Settings**, **Control Panel**, **Administrative Tools**, and then **Computer Management**. Collect the system information (**System Information**) in **Computer Management**.

In Windows Server 2003:

From the **Start** menu, choose **Run**, and enter `msinfo32` in **Name** of the Run dialog box. Then, from the **File** menu in the System Information window, choose **Export** to collect files saved in text format.

9.4.1.3 Collecting Information about the OS and the Installed Product

In Windows 2000 Server:

From the **Start** menu, click **Settings**, **Control Panel**, and then **Add/Remove Programs**. Obtain a hard copy of the Add/Remove Programs window while holding the **Change or Remove Programs** button down.

In Windows Server 2003:

From the **Start** menu, click **Control Panel**, and then **Add or Remove Programs**. Obtain a hard copy of the Add or Remove Programs window while holding the **Change or Remove Programs** button down.

9.4.1.4 Collecting Disk Configuration Information

In Windows 2000 Server:

From the **Start** menu, click **Settings**, **Control Panel**, **Administrative Tools**, and then **Computer Management**. Click **Disk Management** in **Computer Management**, and then obtain a hard copy of the window displaying volume information.

In Windows Server 2003:

From the **Start** menu, click **Control Panel**, **Administrative Tools**, and then **Computer Management**. Click **Disk Management** in **Computer Management**, and then obtain a hard copy of the window displaying volume information.

9.4.1.5 Collecting Cluster Control Software Information

Collect definition information and log information for the cluster control software.

9.4.1.6 Collecting Database Information

When the back up is of a SQL server or Exchange database, collect definition information and log information for the database.

9.4.1.7 Collecting Backup Management Product Information

Collect definition information and log information for the backup management product.

9.4.1.8 Recording Status

Find out details about the operation that was being performed when the problem occurred, and record the details for later use. The following information needs to be confirmed.

- The details of the operation (parameters specified by commands, etc.)
- Error information, such as messages output to the console

- The date and time when the problem occurred
- The machine configuration (the version of each OS and the host name)
- Whether or not the error is reproducible

9.4.2 Adjusting the Output Level for Detailed Trace Log Information

The level of information output to trace log files of Protection Manager can be changed whenever necessary. To adjust the amount of log information to be collected, or to select a certain type of log information to be collected, edit the Protection Products configuration definition file (`init.conf`), which is in the following location:

Protection-Products-installation-destination\conf\init.conf

Method for editing the file:

Specify a numerical value in `LOGLEVEL=`*n*. The numerical value that can be specified and the correspondence with the target log information to be output are shown below. By default, `3` is specified.

`0: Error`

`1: Warning, Error`

`2: Notification, Warning, Error`

`3: Information, Notification, Warning, Error` (default)

`9: All (Information, Notification, Warning, Error, Debug)`

9.4.3 Adjusting the Number of Log Files for Detailed Trace Log Information

The number of trace log files output to trace log files of Protection Manager can be changed whenever necessary. To change the number of log files, edit the Protection Products configuration definition file (`init.conf`), which is in the following location:

Protection-Products-installation-destination\conf\init.conf.

Method for editing the file:

Specify a numerical value in `PP_LOGFILE_NUM=`*n*. The following shows the values you can specify, and the default.

Range of specifiable values: `1` to `16` (default: `2`)

9.5 Investigating Collected Data

This section explains the procedure for investigating the data collected for solving problems when a Protection Manager command is executed and a problem occurs.

Investigate the data in the following order:

1. Standard output logs
2. Trace logs for Protection Manager-cooperative software
3. Detailed trace logs

9.5.1 Investigating Standard Output Logs

When execution of a Protection Manager results in some problem, investigate the information in the standard output logs. When a problem occurs, the messages that were displayed in the standard output, and the time when the problem occurred, are recorded in the standard output logs.

Refer to the contents of the standard output log, and check what messages were output when the problem occurred. For details on how to respond to each message, see the *HiCommand Protection Manager Messages* manual.

9.5.2 Investigating Trace Logs

When the cause of a problem cannot be solved even after investigating the message in the standard output log, or when a problem is not eliminated even if you have carried out the recommended action, then investigate the information in the trace log of the software that cooperates with Protection Manager (referred to as *Protection Manager-cooperative software*).

Protection Manager functions by cooperating with various products: such as a logic volume manager, a backup manager product, and RAID Manager. The execution results of such Protection Manager-cooperative software are recorded in trace logs. The trace log files that record the execution results of Protection Manager-cooperative software are as follows:

- Trace logs related to disk operations

  The execution results when controlling RAID Manager and a logic volume manager are output to trace logs related to disk operations. When you investigate trace logs related to disk operations and a problem with the RAID Manager is found, examine the detailed trace log provided by the RAID Manager and handle the problem according to the contents of the log.

- Trace logs of NetBackup

  The execution results when controlling NetBackup are output to a NetBackup trace log. When you investigate a NetBackup trace log and a problem with NetBackup is found, handle the problem by using the VERITAS NetBackup GUI or by referring to the detailed trace log provided by VERITAS NetBackup.

9.5.3 Investigating the Detailed Trace Log

When a problem is not solved, even after investigating the standard output log and the trace log files of the software linking with Protection Manager, investigate the Protection Manager detailed trace log.

The following information is output to the detailed trace log:

- Internal functions that were used from the execution of a command to the end, as shown in the internal processing sequence information
- The time and the command execution process ID for the processing that was performed at the system call level
- Keywords showing normal and abnormal processing
- Messages showing error causes

The keyword `ER` showing an abnormality is output to the log record that the problem generated. In the case of OS system-call abnormalities, the system also outputs the returned value in addition to the keyword. The error causes are output in a message.

Table 9.7 shows the items output to the detailed trace log.

Table 9.7 Items Output to the Detailed Trace Log

Output item	Description
Number	Log sequence number
Date	Date when the log was output, using the *yyyy/mm/dd* format
Time	Time when the log was output, using the *hh*:*mm*:*ss*.*sss* format
Application name	Protection Manager command name (`drm`*xxx*) that triggered the output of the log.
Process ID	Process ID
Thread ID	Thread ID
Message ID	ID of the message
Type	Type of the message. The types of message are as follows: ▪ `ER` (Error) ▪ `WR` (Warning) ▪ `NT` (Notification) ▪ `IF` (Information) ▪ `DB` (Debug)

Output item	Description
Text	Maintenance information on the product and the message text. The output format is *maintenance-information* : *message-text*.

Table 9.8 shows the message IDs that are output in the detailed trace log. In the detailed trace log indicating the error, use the message ID to identify the product that caused the error. Then, take actions according to the manual for the product.

Table 9.8 Sources of Messages (Controlled Objects) and Corresponding Message IDs

Source of messages (controlled objects)	Message output is triggered by:	Message ID
Volume Manager	Volume Manager operations (general)	KAVX9000-I, KAVX9001-W, KAVX9002-E
	Volume Manager operations (cluster)	KAVX9003-I, KAVX9004-W, KAVX9005-E
System	Operations that change the system	KAVX9100-I, KAVX9101-W, KAVX9102-E
	System operations using the Windows API	KAVX9103-I, KAVX9104-W, KAVX9105-E
	Service operations using the Windows API	KAVX9106-I, KAVX9107-W, KAVX9108-E
	Device operations using the Windows API	KAVX9109-I, KAVX9110-W, KAVX9111-E
Cluster	Cluster operations using the Windows API	KAVX9203-I, KAVX9204-W, KAVX9205-E
Database	Operations and instruction using commands	KAVX9300-I, KAVX9301-W, KAVX9302-E
	Operations and instruction using OCI	KAVX9303-I, KAVX9304-W, KAVX9305-E
	Operations and instruction using ODBC	KAVX9306-I, KAVX9307-W, KAVX9308-E
COM component	Use of COM (MSSQL) interface	KAVX9400-I, KAVX9401-W, KAVX9402-E
	Use of COM (ADO) interface	KAVX9403-I, KAVX9404-W, KAVX9405-E
	Use of COM (CDOEXM) interface	KAVX9406-I, KAVX9407-W, KAVX9408-E
	Use of COM (ADSI) interface	KAVX9409-I, KAVX9410-W, KAVX9411-E
Other products	For detailed trace	KAVX9800-I, KAVX9801-W, KAVX9802-E
	General operations	KAVX9803-I, KAVX9804-W, KAVX9805-E
	General file operations	KAVX9806-I, KAVX9807-W, KAVX9808-E
External product	Cluster products	KAVX9900-I
	RAID Manager	KAVX9901-I
	Backup management product	KAVX9902-I

To investigate the detailed trace log:

1. Use a text editor to search for the keyword `ER` near the time when the abnormality occurred. When the backup is of a SQL Server database, note that the detailed trace log information generated by executing the `drmsqlbackup` command might not be recorded in time-series order.
2. Check the processing in which the abnormality occurred. Check what sort of processing (system call) was being performed when the abnormality occurred.
3. Respond to the error according to the message showing the error causes. If you cannot identify or eliminate the cause of the problem even after you have investigated the detailed trace log, collect the necessary information and contact the dealer from which you purchased the product.

9.6 Troubleshooting Dictionary Map File Errors

During execution of a Protection Manager command, when the system shuts down unexpectedly or the command is forced to terminate, dictionary map files may be lost or become incorrect.

When you execute a Protection Manager command, the following message might appear:

```
KAVX0006-E An error that stopped processing occurred.
Cause = DRM-10106: An internal error occurred.
```

In such a case, if the following message is output to the detailed trace log, a dictionary map file is invalid:

```
ISAM API error code = xxx (yyyyy)
```

The dictionary map files consist of key definition files (`.DEF`), primary key files (`.K01`), secondary key files (`.K`*`nn`*, where *`nn`* is `02` - `99`), and data files (`.DRF`). Therefore, a dictionary map file error can be one of the following error situations:

- Missing or incorrect key definition files
- Missing or incorrect key files
- Missing data files
- Incorrect data files

The utility commands for checking and recovering from dictionary map file errors are stored in the *`Protection-Manager-installation-destination`*`\bin\util` directory. Use these utility commands in the following procedures to check and recover from errors.

Note: The following procedures assume that the dictionary map file location paths are the default paths. Therefore, you may need to change the dictionary map file location paths depending on your operation environment. You may also need to back up all dictionary map files before performing error recovery operations.

9.6.1 Verifying Dictionary Map Files

To verify dictionary map files:

1. Use the `drmischk` command to verify all dictionary map files.

   ```
   drmischk -l 3 "C:\Program Files\HITACHI\DRM\db\DRM*"
   ```

2. Take the following actions in accordance with the output results.

   When the message KAIU020-I is output:

   There are no errors in the dictionary map files.

   Execute the `drmfsdisplay` or `drmfscat` command. If the `drmfsdisplay` or `drmfscat` command can be executed successfully, the dictionary map files are not damaged.

   When the message KAIU004-E is output:

   Some key definition files (`.KDF`) are missing or incorrect.

   Perform the procedures outlined in section 9.6.2.

   When the message KAIU014-E is output:

   Some key files (`.K01` or `.K`*`nn`*) are missing or incorrect.

   Perform the procedures outlined in section 9.6.3.

   When the message KAIU015-E is output:

   Some data files (`.DRF`) are missing.

   Perform the procedures outlined in section 9.6.4.

   When the message KAIU015-E is output:

   Some data files (`.DRF`) are incorrect.

   Perform the procedures outlined in section 9.6.5.

3. If no error message is output, take the following action.

   Although each individual dictionary map is correct, a logical conflict may have occurred between the dictionary map files. In such a case, follow the procedures provided in section 9.6.6.

9.6.2 Restoring Key Definition Files

To restore key definition files:

1. Install the Protection Manager programs on the desired server.
2. Copy and restore the key definition files (`.KDF`) of the applicable dictionary map files from that server.
3. Execute the `drmischk` command to verify the restored dictionary map files.

   ```
   drmischk -l 3 "C:\Program Files\HITACHI\DRM\db\DRM*"
   ```

4. Make sure that the message `KAIU020-I` is output, and then execute the `drmfsdisplay` or `drmfscat` command. If the `drmfsdisplay` or `drmfscat` command can be executed successfully, the dictionary map files are not damaged.

9.6.3 Restoring Key Files

To restore key files (.K01 or .Knn):

1. Create the parameter file for error recovery in the desired directory. Contents of the parameter file:

   ```
   fi-"C:\Program Files\HITACHI\DRM\db\table-to-be-recovered"
   rn-
   ```

2. Execute the `drmiskeymnt` command to restore missing key files (`.K01` or `.Knn`).

   ```
   drmiskeymnt parameter-file-name
   ```

3. Execute the `drmischk` command to verify the restored dictionary map files.

   ```
   drmischk -l 3 "C:\Program Files\HITACHI\DRM\db\DRM*"
   ```

4. Make sure that the message `KAIU020-I` is output, and then execute the `drmfsdisplay` or `drmfscat` command. If the `drmfsdisplay` or `drmfscat` command can be executed successfully, the dictionary map files are not damaged.

9.6.4 Restoring Missing Data Files

9.6.4.1 Restoring Files When Data File Backups Exist

To restore missing data files when backups exist:

1. Restore the data files (`.DRF`) of all dictionary map files.
2. Create the parameter file for error recovery in the desired directory. Contents of the parameter file:

   ```
   fi-"C:\Program Files\HITACHI\DRM\db\table-to-be-recovered"
   rn-
   ```

3. Execute the `drmiskeymnt` command to restore the key files (`.K01` or `.Knn`) of all dictionary map files.
4. Execute the `drmischk` command to verify the restored dictionary map files.

   ```
   drmischk -l 3 "C:\Program Files\HITACHI\DRM\db\DRM*"
   ```

5. Make sure that the message `KAIU020-I` is output, and then execute the `drmfsdisplay` or `drmfscat` command. If the `drmfsdisplay` or `drmfscat` command can be executed successfully, the dictionary map files are not damaged.

9.6.4.2 Restoring Files When Data File Backups Do Not Exist

You cannot restore data files (`.DRF`), which means that you will also be unable to restore the backup catalog information. Execute the `drmdbsetup` command to recreate the dictionary map file.

9.6.5 Restoring Incorrect Data Files

To restore incorrect data files (`.DRF`):

1. Execute the `drmisext` command to extract as many records as possible to a temporary file.

   `drmischk "C:\Program Files\HITACHI\DRM\db\`*`table-to-be-recovered`*`" `*`temporary-file-name`*

2. Create dictionary map files from the temporary file containing the data extracted by the `drmisconv` command.

   `drmisconv -t SI `*`temporary-file-name`*` "C:\Program Files\HITACHI\DRM\db\`*`table-to-be-recovered`*`"`

3. Execute the `drmischk` command to verify the restored dictionary map files.

   `drmischk -l 3 "C:\Program Files\HITACHI\DRM\db\DRM*"`

4. Make sure that the message `KAIU020-I` is output, and then execute the `drmfsdisplay` or `drmfscat` command. If the `drmfsdisplay` or `drmfscat` command can be executed successfully, the dictionary map files are not damaged.

9.6.6 Recreating Dictionary Map Files

1. Copy and save all files and directories in the directory storing the dictionary map files.
2. Delete all files and directories in the directory storing the dictionary map files.
3. When using Protection Manager for SQL, delete all files and directories in the directory storing the VDI metafiles.

   To identify the directory storing the VDI metafiles, execute the `drmsqlinit` command with the `-v` option.

4. Recreate the dictionary map files.

   For details about how to create a dictionary map file, see section 4.5

5. Update the dictionary map files. For details about how to update a dictionary map file, see section 4.13.

Dictionary map files recreated in accordance with the above procedure do not contain a backup catalog showing a backup history. To recover a backup catalog, you must have protected Protection Products backup information beforehand. For details on protecting the operating environment, see section 4.18.

9.7 Calling the Hitachi Data Systems Support Center

The Hitachi Data Systems customer support staff is available 24 hours a day, seven days a week. If you need technical support, please call:

- United States: (800) 446-0744
- Outside the United States: (858) 547-4526

Appendix A Sample HPtM Environment Configuration

This appendix describes an example configuration of Protection Manager environment. This example is based on operation in a specific configuration. For other configurations, you must adjust the settings to suit the operation.

A.1 System Configuration

Figure A.1 shows the system configuration on which the environment configuration example is based.

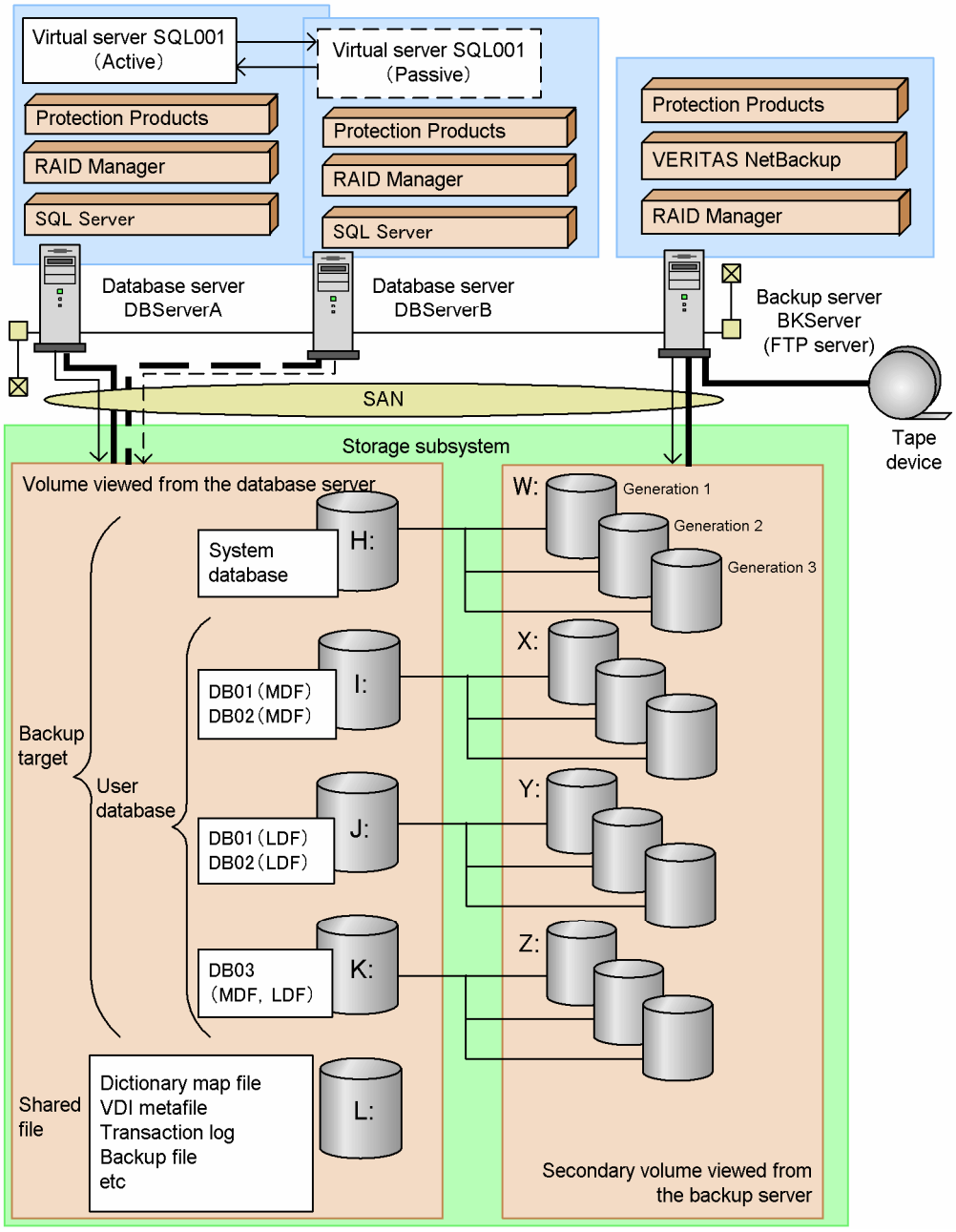


Figure A.1 System Configuration for Protection Manager Environment Configuration

A.1.1 Server Configuration

Using two database servers, configure a virtual server using an active-passive cluster configuration. The backup server is linked to a tape device. Table A.1 shows the server configuration.

Table A.1 Server Configuration for Protection Manager Environment Configuration)

Item	Information (Database server)	Information (Backup server)
OS	Windows Server 2003	Windows Server 2003
Physical server name	DBServerA DBServerB	BKServer
Storage subsystem assistant software	RAID Manager (instance number: 1) Protection Manager Copy Controller Protection Manager for SQL	RAID Manager (instance number: 1) Protection Manager Copy Controller
Protection Manager installation directory	C:\Program Files\Hitachi\DRM (*default-installation-directory*)	C:\Program Files\Hitachi\DRM (*default-installation-directory*)
Cluster software	Microsoft Cluster Service (MSCS)	Not applicable
Virtual server name	SQL001	Not applicable
DBMS	SQL Server	Not applicable
SQL Server instance name	The specified SQL Server instance name	Not applicable
Backup management product	Not applicable	VERITAS NetBackup
FTP server	Not applicable	IIS FTP service
FTP root	Not applicable	C:\FTP_ROOT

A.1.2 Storage Subsystem Configuration

The storage subsystem contains the SQL Server database to be backed up, and other volumes for various user files. The primary volume can be mounted and viewed from the database server. The secondary volume can be mounted and viewed from the backup server. Table A.2 shows the storage subsystem configuration.

Table A.2 Storage Subsystem Configuration for Protection Manager Environment Configuration

Item	Information
Storage Subsystem	Lightning 9900V Series
Usage for the volume replication function	Usual operation is in a split state

Table A.3 shows the storage subsystem volume configuration.

Table A.3 Storage Subsystem Volume Configuration for Protection Manager Environment Configuration

Mount drive from the database server	Mount drive from the backup server	Copy group name	Generation	Volume contents
H:	W:	VG01,VOL11	1st generation	System database
		VG02,VOL21	2nd generation	
		VG03,VOL31	3rd generation	
I:	X:	VG01,VOL12	1st generation	DB01 (MDF) DB02 (MDF)
		VG02,VOL22	2nd generation	
		VG03,VOL32	3rd generation	
J:	Y:	VG01,VOL13	1st generation	DB01 (LDF)
		VG02,VOL23	2nd generation	DB02 (LDF)
		VG03,VOL33	3rd generation	
K:	Z:	VG01,VOL14	1st generation	DB03 (MDF,LDF)
		VG02,VOL24	2nd generation	
		VG03,VOL34	3rd generation	
L:	Not applicable	Not applicable	Not applicable	Dictionary map file, VDI metafile, transaction log backup file, temporary directory for extended commands, etc.

In an active-passive cluster configuration, backup is not performed for the shared directory, and files required by Protection Manager are stored. Table A.4 shows examples of shared directory usage. In this example, the shared drive is `L:`.

Table A.4 Examples of Shared Directory Usage for Protection Manager Environment Configuration)

Item	Information
Dictionary map file storage directory	`L:\PTM`
VDI metafile storage directory	`L:\mssql\meta`
UNDO log file storage directory	`L:\mssql\undo`
Transaction log backup file storage directory	`L:\mssql\log`
Temporary directory for extended commands (automatically generated)	`L:\script_work\` *`operation-ID`* `\DB`

A.2 Example Settings for the RAID Manager Configuration Definition File

This section provides example settings for the RAID Manager configuration definition file (`horcm`*n*`.conf`).

Since a RAID Manager instance (instance number: `1`) is running on the database server, and a RAID Manager instance (instance number: `2`) is running on the backup server, `horcm1.conf` and `horcm2.conf` files are placed on the database server and backup server, respectively. Also, since this example uses a cluster configuration, configuration definition files with the same contents need to be placed on both the active server and the standby server.

RAID Manager configuration definition file, for the database server:
(`C:\Windows\horcm1.conf`)

```
HORCM_MON
#ip_address     service          poll(10ms)     timeout(10ms)
localhost       horcm1           6000           3000

HORCM_CMD
#dev_name                dev_name                dev_name
\\.\PHYSICALDRIVE1

HORCM_DEV
#dev_group     dev_name        port#          TargetID        LU#      MU#
VG01           VOL11           CL2-A           16             19       0
VG01           VOL12           CL2-A           16             20       0
VG01           VOL13           CL2-A           16             21       0
VG01           VOL14           CL2-A           16             22       0
VG02           VOL21           CL2-A           16             19       1
VG02           VOL22           CL2-A           16             20       1
VG02           VOL23           CL2-A           16             21       1
VG02           VOL24           CL2-A           16             22       1
VG03           VOL31           CL2-A           16             19       2
VG03           VOL32           CL2-A           16             20       2
VG03           VOL33           CL2-A           16             21       2
VG03           VOL34           CL2-A           16             22       2

HORCM_INST
#dev_group        ip_address      service
VG01              localhost       horcm2
VG02              localhost       horcm2
VG03              localhost       horcm2
```

RAID Manager configuration definition file, for the database server:
(C:\Windows\horcm2.conf)

```
HORCM_MON
#ip_address     service         poll(10ms)     timeout(10ms)
localhost       horcm2          6000             3000

HORCM_CMD
#dev_name                dev_name                dev_name
\\.\PHYSICALDRIVE1

HORCM_DEV
#dev_group     dev_name        port#          TargetID        LU#      MU#
VG01           VOL11           CL2-B           15             18
VG01           VOL12           CL2-B           15             19
VG01           VOL13           CL2-B           15             20
VG01           VOL14           CL2-B           15             21
VG02           VOL21           CL2-B           15             22
VG02           VOL22           CL2-B           15             23
VG02           VOL23           CL2-B           15             24
VG02           VOL24           CL2-B           15             25
VG03           VOL31           CL2-B           15             44
VG03           VOL32           CL2-B           15             45
VG03           VOL33           CL2-B           15             46
VG03           VOL34           CL2-B           15             47

HORCM_INST
#dev_group        ip_address      service
VG01              localhost    horcm1
VG02              localhost    horcm1
VG03              localhost    horcm1
```

RAID Manager configuration definition file, for the backup server:
(C:\Windows\horcm1.conf)

```
HORCM_MON
#ip_address     service         poll(10ms)     timeout(10ms)
localhost       horcm1          6000           3000

HORCM_CMD
#dev_name                dev_name                dev_name
\\.\PHYSICALDRIVE0

HORCM_DEV
#dev_group     dev_name        port#          TargetID        LU#      MU#
VG01           VOL11           CL2-A           16             19       0
VG01           VOL12           CL2-A           16             20       0
VG01           VOL13           CL2-A           16             21       0
VG01           VOL14           CL2-A           16             22       0
VG02           VOL21           CL2-A           16             19       1
VG02           VOL22           CL2-A           16             20       1
VG02           VOL23           CL2-A           16             21       1
VG02           VOL24           CL2-A           16             22       1
VG03           VOL31           CL2-A           16             19       2
VG03           VOL32           CL2-A           16             20       2
VG03           VOL33           CL2-A           16             21       2
VG03           VOL34           CL2-A           16             22       2

HORCM_INST
#dev_group        ip_address      service
VG01              localhost       horcm2
VG02              localhost       horcm2
VG03              localhost       horcm2
```

RAID Manager configuration definition file, for the backup server:
(`C:\Windows\horcm2.conf`)

```
HORCM_MON
#ip_address     service         poll(10ms)     timeout(10ms)
localhost       horcm2          6000           3000

HORCM_CMD
#dev_name                dev_name                dev_name
\\.\PHYSICALDRIVE0

HORCM_DEV
#dev_group     dev_name        port#          TargetID        LU#      MU#
VG01           VOL11           CL2-B           15             18
VG01           VOL12           CL2-B           15             19
VG01           VOL13           CL2-B           15             20
VG01           VOL14           CL2-B           15             21
VG02           VOL21           CL2-B           15             22
VG02           VOL22           CL2-B           15             23
VG02           VOL23           CL2-B           15             24
VG02           VOL24           CL2-B           15             25
VG03           VOL31           CL2-B           15             44
VG03           VOL32           CL2-B           15             45
VG03           VOL33           CL2-B           15             46
VG03           VOL34           CL2-B           15             47

HORCM_INST
#dev_group        ip_address      service
VG01              localhost    horcm1
VG02              localhost    horcm1
VG03              localhost    horcm1
```

For details about RAID Manager configuration definition files (`horcm`*n*`.conf`), see section 4.2 or the RAID Manager manual.

A.3 Example Settings for Protection Manager Configuration Definition File (init.conf)

This section provides example settings for the Protection Manager configuration definition File (`init.conf`).

A Protection Manager configuration definition file (`init.conf`) is placed on both the database server and backup server. For the database server, the same file is placed on both the active server and the standby server.

Protection Manager configuration definition file, for the database server: (`C:\Program Files\Hitachi\DRM\conf\init.conf`)

```
LOGLEVEL=3
PP_LOGFILE_NUM=2
COM_RETRY_TIME=0
COM_RETRY_WAIT=10
CLU_RETRY_TIME=6
CLU_RETRY_WAIT=10
SVC_RETRY_TIME=6
SVC_RETRY_WAIT=10
DRM_DB_PATH=L:\PTM;SQL001
```

In this environment configuration example, the dictionary map file storage directory is `L:\PTM`. Since the name of the virtual server is `SQL001`, specify `L:\PTM;SQL001` for `DRM_DB_PATH`.

Protection Manager configuration definition file, for the backup server: (`C:\Program Files\Hitachi\DRM\conf\init.conf`)

```
LOGLEVEL=9
PP_LOGFILE_NUM=2
COM_RETRY_TIME=0
COM_RETRY_WAIT=10
CLU_RETRY_TIME=6
CLU_RETRY_WAIT=10
SVC_RETRY_TIME=6
SVC_RETRY_WAIT=10
```

For details about the Protection Manager configuration definition file (`init.conf`), see section 4.3 and section 4.5.

A.4 Example Settings for the RAID Manager-Linkage Definition File (DEFAULT.dat)

This section provides example settings for the RAID Manager-linkage definition file (`DEFAULT.dat`).

This example assumes that the RAID Manager that manages the primary volume and secondary volume is placed on the database server.

The RAID Manager-linkage definition file (`DEFAULT.dat`) is placed on the database server and the backup server. For the database server, the same file is placed on the active server and the standby server.

- RAID Manager-linkage definition file, for the database server: (`C:\Program Files\Hitachi\DRM\conf\raid\DEFAULT.dat`)

```
HORCMINST=1
HORCMINST_AUX=2
RETRY_TIME=120
RETRY_WAIT=5
VENDER=HITACHI
PRODUCT=OPEN-3
INSTALLPATH=C:\HORCM
```

- RAID Manager-linkage definition file, for the backup server: (`C:\Program Files\Hitachi\DRM\conf\raid\DEFAULT.dat`)

```
HORCMINST=2
HORCMINST_AUX=1
RETRY_TIME=120
RETRY_WAIT=5
VENDER=HITACHI
PRODUCT=OPEN-3
INSTALLPATH=C:\HORCM
```

For details about the RAID Manager-linkage definition file (`DEFAULT.dat`), see section 4.4.

A.5 Example of Creating Dictionary Map Files

Since this environment configuration example uses an active-passive cluster configuration, the dictionary map file is created on the shared directory. In this example, the dictionary map file storage directory is `L:\PTM`.

To create a dictionary map file, open an MS-DOS prompt on the database server (active server), and execute the following command (utility):

```
PROMPT> set DRM_HOSTNAME=SQL001
PROMPT> C:\Program Files\Hitachi\DRM\bin\util\drmdbsetup -i
```

Check that the dictionary map file has been created in the shared directory `L:\PTM`.

For details about creating a dictionary map file, see section 4.5.

A.6 Example of Registering SQL Server Information

This section provides an example of registering SQL Server information. Since this environment configuration uses an active-passive cluster configuration, SQL Server information is created in the shared directory. In this example, such information is registered in `L:\mssql`.

To register SQL Server information, execute the `drmsqlinit` command on both the active server and the standby server. In this example, since the SQL Server instance uses a set instance, specify the `DEFAULT` parameter.

```
PROMPT> set DRM_HOSTNAME=SQL001
PROMPT> drmsqlinit DEFAULT
PROMPT> KAVX1100-I Enter the name of the storage directory for VDI metafiles:
        L:\mssql\meta
```

```
PROMPT> KAVX1101-I Enter the timeout interval for VDI generation (in
seconds, between 0 and 3600):
```

```
        1000
```

```
PROMPT> KAVX1102-I Enter the name of the storage directory for undo files:
```

```
        L:\mssql\undo
```

```
PROMPT> KAVX1107-I Enter the name of the storage directory for backup-log
files:
```

```
        L:\mssql\log
```

Check that the SQL Server database configuration definition file was created.

Database configuration definition file:

(`C:\Program Files\Hitachi\DRM\conf\MSSQL\DEFAULT.dat`)

```
VDI_METAFILE_DIR=l:\mssql\meta
VDI_TIMEOUT_SEC=1000
UNDOFILE_DIR=l:\mssql\undo
LOGBACKUPFILE_DIR=l:\mssql\log
```

For details on how to register SQL Server information, see section 4.8.

A.7 Example Settings for Linkage to Backup Management Product

This example uses VERITAS NetBackup 4.5 as the backup management product, and assumes the following:

- A CLASS is created for each parallel backup (in this example, 5 are created).
- Each CLASS is created under its own name: Class1, Class2, Class3, Class4, and Class5.
- A SCHEDULE is created in each CLASS, with the following names:

  Class1:Schedule1

  Class2:Schedule2

  Class3:Schedule3

  Class4:Schedule4

  Class5:Schedule5
- The save period for each media is set to 14 days.

CLASS can be replaced with POLICY for this example. However, both POLICY and CLASS cannot be specified for the same MOUNT_POINT.

A.7.1 Example Registering Information for Linkage to the Backup Management Product

This section provides an example registering information for linkage to the backup management product.

To register information for linkage to the backup management product, execute the drmtapeinit command on the backup server, as follows:

```
PROMPT> drmtapeinit
PROMPT> KAVX0411-I Enter the backup management product name.
        NBU
```

```
PROMPT>  KAVX0417-I Enter the number of days to store the backup catalog.
```

```
        10
```

Check that the configuration definition file for linkage with a backup management product has been created.

- Configuration definition file for linkage with a backup management product

Database configuration definition file: (C:\Program Files\Hitachi\DRM\conf\tape\DEFAULT.dat)

```
NBU
10
```

For details about registering information for linkage to the backup management product, see section 4.11.1.

A.7.2 Example of the Configuration Definition File for Tape Backup

This section provides an example of the configuration definition file for tape backup.

The configuration definition file for tape backup is placed on the backup server. In this example, the mount point for mounting the secondary volume from the backup server (`W:`, `X:`, `Y:`, and `Z:`) is specified. Also, `default` is specified for the metafile directory, but this can also be an absolute path. In this example, `C:\FTP_ROOT\script\OperationA\AP` is specified.

Configuration definition file, for tape backup:

(`C:\Program Files\Hitachi\DRM\conf\tape\NBU.dat`)

```
INST_PATH=C:\Program Files\VERITAS\NetBackup

MOUNT_POINT=W:
CLASS=Class1
SCHEDULE=Schedule1

MOUNT_POINT=X:
CLASS=Class2
SCHEDULE=Schedule2

MOUNT_POINT=Y:
CLASS=Class3
SCHEDULE=Schedule3

MOUNT_POINT=Z:
CLASS=Class4
SCHEDULE=Schedule4

MOUNT_POINT=default
CLASS=Class5
SCHEDULE=Schedule5

PARALLEL_COUNT=5
```

For details on the configuration definition file for tape backup, see section 4.11.2.

A.8 Example Settings for the Host Environment Settings File

This section provides example settings for the host environment settings file, as used with the extended command.

The host environment settings file is placed on each the database server and backup server. For the database server, the same file is placed on both the active server and the standby server.

Host environment settings file, for the database server:

(C:\Program Files\Hitachi\DRM\script\conf\host.dat)

```
HOST_ROLE=DB
MAX_LOG_LINES=1000
```

Host environment settings file, for the backup server:

(C:\Program Files\Hitachi\DRM\script\conf\host.dat)

```
HOST_ROLE=BK
MAX_LOG_LINES=1000
```

For details about the host environment settings file, see section 4.15.4.

A.9 Example Settings for an Operation Definition File

This section provides example settings for an operation definition file.

Operation definition files (*operation-ID*.dat) with the same contents are placed on each backup server (both the active server and the standby server). In this example, the operation ID is OperationA.

Example operation definition file:

(C:\Program Files\Hitachi\DRM\script\conf_OperationA)

```
BACKUP_OBJECT=MSSQL
DB_SERVER_NAME=SQL001
INSTANCE_NAME=DEFAULT
TARGET_NAME=
FTP_HOME_DIR=C:\FTP_ROOT
FTP_SUB_DIR=script
SET_DRM_HOSTNAME=1
```

For details about the operation definition file, see section 4.15.6 for file system backup, section 4.15.7 for SQL database backup, or section 4.15.8 for Exchange database backup.

A.10 Example Settings for a Copy-Group List File

This section provides example settings for a copy-group list file. It is not necessary to create this file, but is useful when you want to define multiple copy groups at the same time.

In this example, a copy-group list file is created for each generation.

Copy-group list file is created for the first generation: (`L:\CG01.txt`)

```
VG01,VOL11
VG01,VOL12
VG01,VOL13
VG01,VOL14
```

Copy-group list file is created for the second generation: (`L:\CG02.txt`)

```
VG02,VOL21
VG02,VOL22
VG02,VOL23
VG02,VOL24
```

Copy-group list file is created for the third generation: (`L:\CG03.txt`)

```
VG03,VOL31
VG03,VOL32
VG03,VOL33
VG03,VOL34
```

For details about copy-group list files, see section 4.12.1.

Appendix B Files Used in Protection Manager

Table B.1 lists the files used in Protection Manager.

Table B.1 List of Files Used in Protection Manager

File Name	Description	Storage location
Dictionary map files	Files that contain mapping information: ranging from information about the objects to be backed up, to information about the storage subsystems. These files are necessary in Protection Manager to automate backup processing. Dictionary map files consist of the following: ▪ Application map file ▪ Core map file ▪ Copy group map file ▪ Backup catalog	*Protection-Manager-installation-directory*\db Note that the storage location needs to be changed on a shared disk for a cluster configuration. To change the storage location, specify DRM_DB_PATH in the Protection Manager configuration definition file (init.conf).
Application map file	Files that contain relational information between database objects to be backed up, such as journal logs, and the corresponding files.	Files shared, regardless of the operating environment.
Core map file	Files that contain related information: ranging from the mount point directories of file systems, to the disk numbers within the storage subsystems.	Files shared, regardless of the operating environment.
Copy group map file	Files that contain mapping information between primary volumes and their corresponding secondary volumes.	Files shared, regardless of the operating environment.
Backup catalog file	Files that contain exported backup catalog information. These files are used to exchange backup catalog information between file servers (or database servers) and backup servers.	Files shared, regardless of the operating environment.
Database configuration definition files (*instance-name*.dat)	Files that contain information (parameters) about target databases registered by using the drmsqlinit command. These files are used when backup is of a SQL Server database.	▪ For SQL Server: *Protection-Manager-installation-directory*\conf\MSSQL
Configuration definition files for tape backup	Definition files for linkage to backup management products. ▪ For NetBackup: NBU.dat ▪ For Backup Exec: BEWS.dat These files are used for linkage to backup management products	*Protection-Manager-installation-directory*\conf\tape
Configuration definition file for linkage with a backup management product (DEFAULT.dat	Definition files for linkage to backup management products that were registered by using the drmtapeinit command. These files are used for linkage to backup management products.	*Protection-Products-installation-directory*\conf\tape
RAID Manager configuration definition files (horcm*n*.conf)	Files that define the system configuration for operating RAID Manager.	▪ In Windows 2000 Server: *system-drive*\WINNT ▪ In Windows Server 2003: *system-drive*\Windows

File Name	Description	Storage location
RAID Manager linkage definition file (DEFAULT.dat)[1]	Files that describe the definitions for linking Protection Manager with RAID Manager.	*Protection-Manager-installation-directory*\conf\raid
Protection Manager configuration definition file (init.conf)[1]	File that defines the operating settings for Protection Manager.	*Protection-Manager-installation-directory*\conf
Batch definition files	When the backup is of a file system: ▪ Definition files containing a list of files or directory absolute paths subject to backup. When the backup is of a SQL Server database: ▪ Definition files containing a list of databases subject to backup. When the backup is of an Exchange database: ▪ Definition files containing a list of names of storage groups subject to backup.	User creates the files in the desired location.
Mount point directory batch definition files	Files that batch specify multiple file system mount point directories subject to backup.	--
Copy group batch definition files	Files that contain a list of copy groups to be executed. These files are used for extended commands or basic commands.	Using an extended command or basic command.
Transaction log batch definition files	Files that contain a list of SQL Server database transaction log files. These files are used when the backup is of a SQL Server database.	--
Operation definition files	Files that define information for items subject to backup, such as server names and backup objects. These files are used for extended commands.	*Protection-Manager-installation-directory*\script\conf
Host environment settings file (host.dat)[1]	Files that define information set for each host, such as server type (file server or backup server) and maximum number of lines in a log. These files are used for extended commands.	*Protection-Manager-installation-directory*\script\conf
Backup ID files	Temporary files created when extended commands are executed. These files are used to carry over backup IDs between extended commands.	File server or database server: *directory-at-the-same-level-as-the-dictionary-map-file-directory*\script_work*operation-ID*\DB[2] Backup server: *FTP HOME_DIR-value**FTP SUB_DIR-value**operation-ID*\BK and *FTP_HOME_DIR-value**FTP_SUB_DIR-value**operation-ID*\AP
Temporary files	Saved copies of files specific to the system subject to backup, created during disk backup. These files are used during database restoration. These files are used for extended commands. However, these files are not used when the backup is of a file system	
Transaction log file	▪ Files used during recovery operations for SQL Server databases. ▪ Files that temporarily store Exchange Server transactions. These files are used when the backup is of a SQL Server database or Exchange database.	

File Name	Description	Storage location
User script file[3]	File used for executing commands during file system or database backup. When the `drmsqlbackup` command is executed with the `-script` option, the script file that the user specifies is read and the command corresponding to the definition in the user script file is executed. This file is used for backup in a cascade configuration.	User creates the files in the desired location.
Cluster configuration definition file (*cluster-name*`.dat`)	File that stores information about the cluster resources registered by the `drmclusinit` command. This file is used when VERITAS Cluster Server is used.	*Protection-Manager-installation-directory*\conf\cluster
Copy parameters definition file (*any-file-name*)	Create this file to change the retry time for confirming the pair statuses used when commands are executed in operation. Specify this file during command execution for backup, restoration, and resynchronization.	*Protection-Manager-installation-directory*\conf\raid A user-specified file name.
Copy group mount file (`CG_MP.conf`)	This file is necessary when copy group names are used for parameters, in a dynamic disk configuration.	*Protection-Manager-installation-directory*\conf\vm\
VSS definition file (`vsscom.conf`)	File that backs up a file system or Exchange database, that is to be set when using VSS to perform online backup.	*Protection-Manager-installation-directory*\conf
Mail -sending definition file (`mail.conf`)	File that sets the destination addresses and contents of e-mail that is to be sent in the event of an error during execution of a backup command.	*Protection-Manager-installation-directory*\conf

[1]Installed in the same directory is another file with the same file name except that `.model` is suffixed to the end of the file name. The default values are set in this other file. Thus, if you remove the suffix `.model` from the file name, you can use this other file as a definition file.

[2]For example, if the dictionary map file directory is `L:\PTM`, this is `L:\script_work\`*`operation-ID`*`\DB`.

[3]A sample script file is installed in the following location:

`Protection-Manager-installation-folder``\script\sample`

Appendix C Replacing Primary and Secondary Volume Disks While Protection Manager Programs Are Running

If the disk used as a primary or secondary volume is damaged, you must restore data and change the settings according to one of the procedures below, as well as replace the damaged disk. The procedure differs depending on whether a primary disk or secondary disk belongs to the replaced disk.

- Procedure for replacing a disk to which a primary volume belongs:
- Procedure for replacing a disk to which a secondary volume belongs:

  Copy the latest data from the primary volume to a new secondary volume.

  When restoring tape backup data to a new secondary volume, use the `drmmount` command (with the `-force` option specified) and the `drmmediarestore` command.

C.1 Procedure for Replacing a Primary Volume Disk (in Windows)

Figure E-1 shows how to replace a primary volume disk:

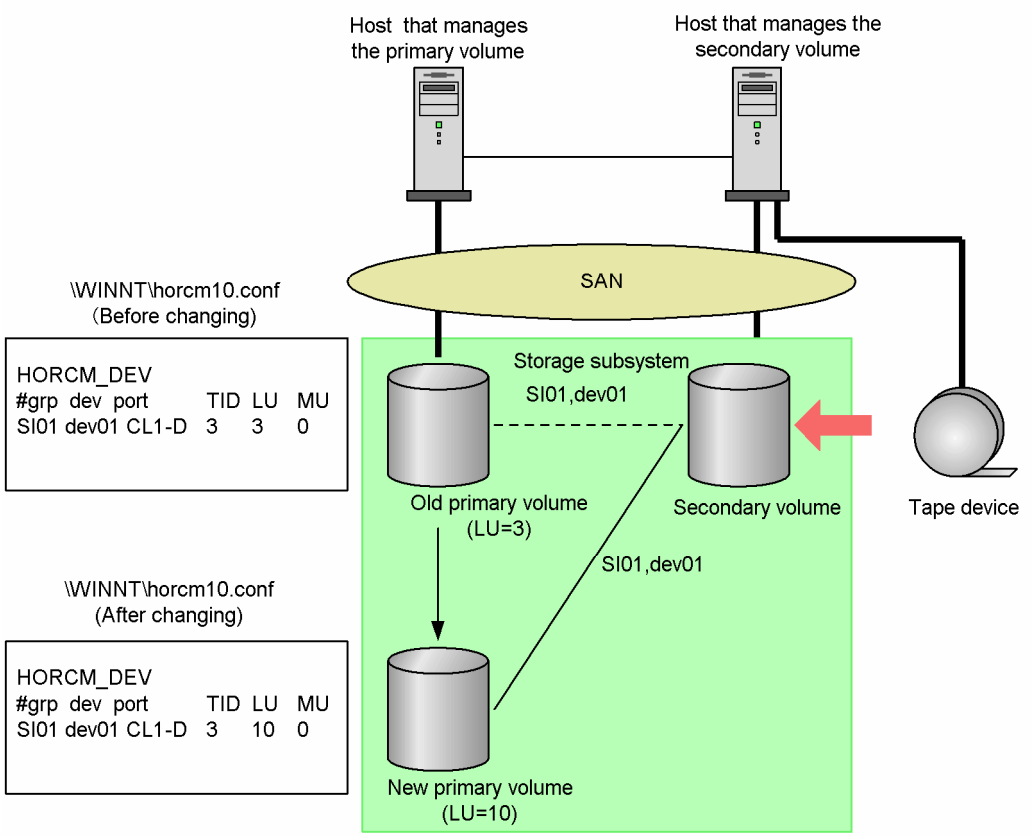


Figure E.1 Example of Replacing a Primary Volume Disk

To replace the primary volume:

1. Execute the following RAID Manager command to delete the pair definition of the primary volume to be replaced:

```
PROMPT> pairsplit -g SI01 -S
```

2. On the host that manages the primary volume, delete primary volume partitions.

   In Windows 2000 Server:

   From the **Start** menu, choose **Settings**, **Control Panel**, **Management Tool**, and **Computer Management**. Select **Disk Management** from the Computer Management window, right-click the partition that you want to delete, and click **Delete Partition**.

   In Windows 2003 Server:

   From the **Start** menu, choose **Control Panel**, **Management Tool**, and **Computer Management**. Select **Disk Management** from the Computer Management window, right-click the partition that you want to delete, and click **Delete Partition**.

3. On the host that manages the primary volume, create partitions for the new primary volume.

   In Windows 2000 Server:

   From the **Start** menu, choose **Settings**, **Control Panel**, **Management Tool**, and **Computer Management**. Select **Disk Management** from the Computer Management window and create partitions in disk management.

   In Windows Server 2003:

   From the **Start** menu, choose **Control Panel**, **Management Tool**, and **Computer Management**. Select **Disk Management** from the Computer Management window and create partitions in disk management.

4. On the host that manages the primary volume, change the contents of the definition file for the target RAID Manager instance to the Port, TargetID, and LU of the new primary volume and then restart the RAID Manager instance.

```
PROMPT> horcmshutdown.sh 10
PROMPT> horcmstart.sh 10
```

5. Execute the following command to generate a pair of the new primary volume and the secondary volume on the host that manages the primary volume:

```
PROMPT> paircreate -g SI01 -vl -c 15
```

6. Execute the `pairevtwait` command and wait till the pair state becomes PAIR.

```
PROMPT> pairevtwait -g SI01 -s PAIR -t 10 180
```

7. Execute the `pairsplit` command to split the pair.

```
PROMPT> pairsplit -g SI01
```

8. Execute the `drmmount` command to mount the secondary volume.

```
PROMPT> drmmount 0000000053
```

9. Execute the `drmmediarestore` command to restore tape data on the host that manages the secondary volume.

```
PROMPT> drmmediarestore 0000000053
```

10. Execute the `drmumount` command to unmount the secondary volume.

```
PROMPT> drmumount 0000000053
```

11. On the host that manages the secondary volume, export the backup catalog used when data was restored from the tape and transfer it to the host that manages the primary volume.

```
PROMPT> drmdbexport 0000000054 -f /tmp/expfile
PROMPT> ftp dbhost
```

12. On the host that manages the primary volume, import the catalog.

```
PROMPT> drmdbimport -f /tmp/expfile
```

13. Execute the `drmfsrestore` command with the `-force` option to restore data from the secondary volume to the primary volume.

```
PROMPT> drmfsrestore 0000000155 -resync -force
```

C.2 Procedure for Replacing a Secondary Volume Disk

Figure E-2 shows how to replace a disk on which a secondary volume belongs:

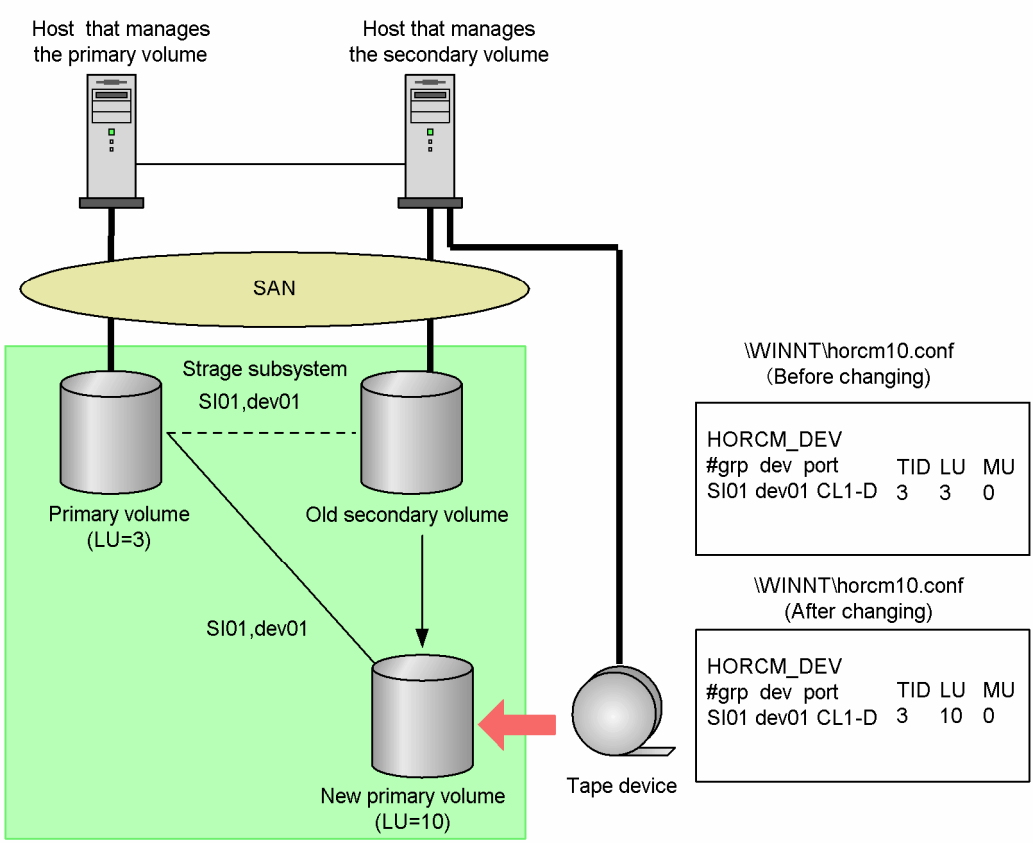


Figure E.2 Example of Replacing a Secondary Volume Disk

To replace the secondary volume:

1. Delete the pair definition of the primary volume.

```
PROMPT> pairsplit -g SI01 -S
```

2. On the host that manages the secondary volume, create partitions for a new secondary volume.

   In Windows 2000 Server:

   From the **Start** menu, choose **Settings**, **Control Panel**, **Management Tool**, and **Computer Management**. Select **Disk Management** from the Computer Management window and create partitions by using the disk management functionality.

   In Windows Server 2003:

   From the **Start** menu, choose **Control Panel**, **Management Tool**, and **Computer Management**. Select **Disk Management** from the Computer Management window and create partitions by using the disk management functionality.

3. On the host that manages the secondary volume, change the definition file for the RAID Manager instance (secondary) and restart the RAID Manager instance.

```
PROMPT> horcmshutdown 10
PROMPT> horcmstart 10
```

4. On the host that manages the primary volume, generate a primary volume and secondary volume pair.

```
PROMPT> paircreate –g SI01 –vl –c 15
```

5. Wait for the completion of pair generation.

```
PROMPT> pairevtwait –g SI01 –s PAIR –t 10 180
```

6. Split the pair.

```
PROMPT> pairsplit –g SI01
```

7. To restore data from the tape, execute the `drmmount` command with the `–force` option to mount the secondary volume.

```
PROMPT> drmmount 0000000065 –force
```

8. Execute the `drmmediarestore` command to restore data from the tape to the secondary volume.

```
PROMPT> drmmediarestore 0000000065
```

9. Execute the `drmumount` command to unmount the secondary volume.

```
PROMPT> drmumount 0000000065
```

Appendix D Restoring from Tape to a Primary Volume in QuickShadow Configuration

This appendix provides an example of restoring from tape directly to a primary volume, in a QuickShadow configuration.

Note: The procedure for restoring from tape directly to a primary volume is not supported when the configuration uses Exchange Server.

D.1 QuickShadow Configuration

This example assumes the following as the example QuickShadow configuration:

- There are two server machines, one each for the database server and backup server. Each is directly connected to a tape device.
- The tape device connected to the backup server is used for backing up the contents of the secondary volume to tape, using the usual backup procedure.
- The tape device connected to the database server is used for restoring the tape contents directly to the primary volume.

D.2 Backing up to Tape on the Backup Server

When backing up to tape on the backup server, export the backup information to a file, and then manage that file appropriately. Give the output file a name that allows you to understand the backup contents, so that you can identify the file at restoration.

To export the backed-up data to a file, on the backup server execute the following command:

```
PROMPT> drmdbexport backup-ID -f file-name
```

The backup ID used in this command is the backup ID that was used when backing up to tape. (Create the file in the DRM_DB_PATH directory.) Make sure that you back up the output file because, if this file becomes unavailable, you will be unable to restore from tape directly to a primary volume.

When using an extended command to back up to tape, check the backup-ID record file (*operation-ID*.bid) in the directory shown below to confirm the backup ID. The backup ID used for export is the backup ID used when backing up to tape. (Create the file in the DRM_DB_PATH directory.)

*FTP_HOME_DIR-value**FTP_SUB_DIR-value**operation-ID*\BK*operation-ID*.bid

D.3 Restoring From Tape Directly to the Primary Volume

Perform the following procedure on the database server.

To restore data that was backed up to tape directly to the primary volume:

1. Execute the following command to import the file of backup information corresponding to the data to be restored. This file is from among the backup information files that were output by the `drmdbexport` command (the files exported as described in section F.2) when backing up to tape, and which were sent from the backup server to the database server.

```
PROMPT> drmdbimport -f file-name
```

   (Create the file in the `DRM_DB_PATH` directory.)

2. Stop the database.

   When the backup is of a SQL Server database:

   If restoration does not include a system database, detach the database to be restored.

   If restoration does include a system database, stop the SQL Server service.

   If the system is in a cluster configuration, place the following resources offline: `SQL Server`, `SQL Server Agent`, `SQL Server Fulltext`.

3. Delete the pair definition of the primary volume.

   Execute the `drmfscat` or `drmsqlcat` command to confirm the volume subject to restoration from the imported backup information. Then, using a RAID Manager command, delete the pair definition of this volume.

4. Restore from the tape to the primary volume.

   Using backup management software, restore the data from the restoration-target tape to the backup-source primary volume.

5. Recover the database control information.

   When the backup is of a SQL Server database:

   Using the backup catalog stored on the backup server, restore the metafiles to the database server.

6. Recover the database.

   When the backup is of a SQL Server database, execute the `drmsqlrestore` command with the `-no_resync` option specified.

```
PROMPT> drmsqlrestore backup-ID -no_resync
```

   Execute the `drmsqlrecovertool` command to perform recovery.

7. Recover the pair definition of the primary volume.

   Using a RAID Manager command, redefine the pair definition of the volume that is the target of restoration.

Appendix E Actions Taken for Message KAVX5105-W

This appendix describes the cause when the message KAVX5105-W is output, and the actions to be taken.

To restore data backed up by TrueCopy or UR, Protection Manager programs restore the data in the secondary volume by using pair resynchronization, which swaps the primary and secondary volumes.

Protection Manager programs restore the data backed up by TrueCopy or UR as follows:

1. Swap the primary volume and secondary volume to resynchronize the pair.
2. Restore the data to the volume connected to the file server or the database server.
3. Swap the primary and secondary volumes again to return them to the original state. If, however, an error occurs during restoration, the operations on the pair may be interrupted.

If the KAVX5105-W message is output, then the swapping of the secondary and primary volumes in step 3 has failed, so the volume connected to the file server or the database server remains a secondary volume when the restoration finishes. Because the restoration of the volume connected to the file server or the database server has finished, operation on the volume from the file server or the database server is allowed while the actual primary volume is recognized as the secondary volume. Under these conditions, you cannot execute Protection Manager commands, such as a backup command, on this copy group.

First check the RAID Manager log information to identify the cause of the failure of the swapping of the primary and secondary volumes, remove the error, and then carry out the following procedures to restore the volume status.

To recover from failure of the swapping of the primary and secondary volumes:

1. Make sure that the RAID Manager instance managing both the primary and secondary volumes is running.
2. Set up the instance that manages the volumes connected to the file server or the database server, and the environment variable used to manipulate the pair volumes for TrueCopy or UR.

   In the following example, the instance that manages RAID Manager is indicated by HORCMINST=0, and the environment variable that manipulate the pair volumes for TrueCopy or UR is HORCC_MRCF=.

```
PROMPT> set HORCMINST=0
PROMPT> set HORCC_MRCF=
```

3. Check the copy group that was restored.

   In the following example, the copy group whose backup ID is `0000000001` was restored:

```
PROMPT > drmsqlcat default -backup_id 0000000001
INSTANCE: default
BACKUP-ID: 0000000001 BACKUP-MODE: ON-LINE INSTANCE: default ORIGINAL-ID: 0000000001
START-TIME: 2002/06/01 10:00:00 END-TIME: 2002/06/01 10:03:00 HOSTNAME: SQL1
T  DB   OBJECT    FILE             FS  DG  DEVICE     COPY-GROUP
M  DB1  METAFILE  C:\METADIR\Meta1 C:  -   -          -
D  DB1  DATAFILE  D:\SQL\data1     D:  -   Harddisk1  TC01,dev01
-  -    -         -                -   -   -          TC01,dev01
PROMPT>
```

4. Check the pair status of the copy group. Make sure that the status of the primary volume is `PSUS`, and the status of the secondary volume is `SSWS`. In the example below, the copy group is indicated by `TC01` and `dev01`.

```
PROMPT > pairdisplay -g TC01 -d dev01 -fc
```

5. Perform pair resynchronization on the copy group to swap the primary and secondary volumes, and then wait until the status changes to `PAIR`.

```
PROMPT> pairresync -g TC01 -d dev01 -swaps
PROMPT> pairevtwait -g TC01 -d dev01 -s pair -t 300
```

6. Check the pair status of the copy group. Make sure that the status of both the primary and secondary volumes are `PAIR`.

```
PROMPT> pairdisplay -g TC01 -d dev01 -fc
```

7. Split the copy group pair, and wait until the status changes to `PSUS`.

```
PROMPT> pairsplit -g TC01 -d dev01 -rw
PROMPT> pairevtwait -g TC01 -d dev01 -s psus -t 300
PROMPT> pairdisplay -g TC01 -d dev01 -fc
```

8. Check the pair status of the copy group. Make sure that the status of the primary volume is `PSUS`, and the status of the secondary volume is `SSUS`.

```
PROMPT>pairdisplay -g TC01 -d dev01 -fc
```

Appendix F Actions Taken for Message KAVX5137-E

This section describes the cause, recovery procedures and actions to be taken when the message KAVX5137-E is output:

F.1 The cause of the message (KAVX5137-E)

When the backup server identifies multiple secondary volumes recovered from the same primary volume, Windows changes the disk signatures that are overlapped.

When Microsoft Cluster Service (MSCS) is used as cluster software, the MSCS resource fails to change the status to online when trying to restore the secondary volume after the disk signature has been modified. For example, Windows fails to restore them properly under the following conditions:

- When Microsoft Cluster Service is being used as cluster software.
- Secondary volumes are in various generations.
- The secondary volumes are set to use in the backup server.
- The OS (Windows) of the backup server detected the overlapped disk signatures and changed them.
- The secondary volume should be restored after its disk signature is changed.

For Protection Products programs, the signature of the physical disk is browsed when restoring the Microsoft Cluster Service environment and the offline resources changed to online. If the signature of the physical disk differs from the one recorded in the cluster, the message KAVX5137-E appears and terminates the Restore command.[1]

[1] EX_DRM_FS_RESTORE(drmfsrestore command), EX_DRM_SQL_RESTORE(drmsqlrestore command), and DRM_EXG_RESTORE(drmexgrestore command)

F.2 Recovery procedure for message KAVX5137-E

This section describes, for each server, how to recover when the KAVX5137-E message is output.

F.2.1 Operations in the backup server

Check the disk signature of the secondary volume and update the disk signature.

To check and update the disk signature:

1. Check the disk signature of the secondary volume

   Execute the drmdevctl command with the -sigview option specified.

   For the restoration source of a secondary volume, check the disk signature (CUR_DISKID) of the current point. Also check the disk signature (BKU_DISKID) to determine whether the value is correct.

   The following shows an example of executing the drmdevctl command:

```
BKServer > drmdevctl 0000000001 -sigview
COPY_GROUP   ICE          TYPE   CUR_DISKID   BKU_DISKID
VG01,dev01   Harddisk10   MBR    ABCDEF0004   ABCDEF0001
VG01,dev02   Harddisk11   MBR    ABCDEF0005   ABCDEF0002
VG01,dev03   Harddisk12   MBR    ABCDEF0006   ABCDEF0003
```

   The above example shows that the disk signature (CUR_DISKID) of the current point, and the disk signature (BKU_DISKID) recorded at the time of the backup, are different because the disk signature has been changed after the backup.

2. Update the disk signatures of the secondary volume

   Execute the drmdevctl command with the -sigset option specified.

   The following shows an example of executing the drmdevctl command:

```
BKServer > drmdevctl 0000000001 -sigset
```

   The above example shows how the disk signature is updated using the signatures recorded at backup.

3. Check to see whether the disk signatures of the secondary volume are being updated correctly:

   Execute the drmdevctl command with the -sigview option specified.

   Make sure to check the disk signatures carefully, as Windows might update the updated disk signatures again if the same disk signatures are found on a different disk.

   The following shows an example of executing the drmdevctl command:

```
BKServer > drmdevctl 0000000001 -sigview
COPY_GROUP   ICE          TYPE   CUR_DISKID   BKU_DISKID
VG01,dev01   Harddisk10   MBR    ABCDEF0001   ABCDEF0001
VG01,dev02   Harddisk11   MBR    ABCDEF0002   ABCDEF0002
VG01,dev03   Harddisk12   MBR    ABCDEF0003   ABCDEF0003
```

F.2.2 Operations on the file server or database server

The secondary volume is restored using the RAID Manager commands.

To restore the secondary volume:

1. Change the status of the cluster resource.

   When restoration was executed while the cluster resource is offline:

   Check if the physical disk resource is `offline` or `failed`.

   When restoration was executed while the cluster resource is online:

   Place the physical disk resource `online (Ext Maintenance, Internal State 'Offline')`.

   To change the physical disk resource from `Online` to `Online(Maintenance)`, execute the following command:

```
FSServer > CLUSTER RESOURCE "resource-name" /MAINTENANCEMODE:ON
```

   To change the physical disk resource from `Online(Maintenance)` to `online (Ext Maintenance, Internal State 'Offline')`, execute the following command:

```
FSServer > CLUSTER RESOURCE "resource-name" /EXTMAINTENANCEMODE:ON
```

2. Using the RAID Manager commands, restore the secondary volume.

   Execute the `pairresync` command and `pairsplit` command. Make sure that the pair status of the `pairresync` command is `PSUS-SSUS` and for `pairsplit` command, the pair status is `PAIR-PAIR`.

   The following shows an example of executing the `pairresync` command and `pairsplit` command:

```
FSServer > pairresync -g VG01 -restore
FSServer > pairsplit -g VG01
```

3. Change the cluster group to online:

   When restoration was executed while the cluster resource is offline:

   Launch the cluster administrator and change the cluster group to online.

   When restoration was executed while the cluster resource is online:

   Place the physical disk resource `Online`.

   To change the physical disk resource from `Online(Ext Maintenance, Internal State 'Offline')` to `Online(Maintenance)`, execute the following command:

```
FSServer > CLUSTER RESOURCE "resource-name" /EXTMAINTENANCEMODE:OFF
```

   To change the physical disk resource from `Online(Maintenance)` to `Online`, execute the following command:

```
FSServer > CLUSTER RESOURCE "resource-name" /MAINTENANCEMODE:OFF
```

   After the above operations, start the Windows disk management function to mount the disk in the drive or folder where it was.

4. To make the restoration command executable, perform the following operation:

   If the backup is of an Exchange database:

   Start the system manager, and re-mount the information store.

   If the backup is of a SQL Server database:

   See section 7.1.10to determine whether the database can properly be restored.

5. Re-execute the restore command

   Re-execute the restore command of the Protection Products programs.

Note: Even if the `CLUSTER` command returns system error code 997, after waiting several seconds the status will return to normal.

F.2.3 How to avoid the message (KAVX5137-E)

Do one of the following procedures to avoid the message (`KAVX5137-E`):

- Conceal all the secondary volumes from the backup server.

  If you conceal all the secondary volumes, the disk signatures will not be changed. Note that sometimes the disk signatures have already been changed, before a secondary volume is concealed from the backup server. After concealing the secondary volumes, obtain a backup of all the volumes designated as secondary volumes. If you need to backup data of the secondary volumes, make the backup copy on tape, because all secondary volumes are overwritten.

  See section 4.14.1 for information on how to conceal all the secondary volumes from the backup server.

- Check and update the disk signatures before restoring

  If you do not want to conceal all the secondary volumes from the backup server, the disk signatures will be overwritten. Before restoring, check to see if the disk signatures have been changed. If they have, restore the disk signature after the update.

Acronyms and Abbreviations

API	application programming interface
application map file	One of the map files making up a dictionary map file. Application map files are used to keep track of the mapping information between the application data to be backed up, and the files on the file system.
backup catalog	A collection of information necessary to manage the history and generations of backups performed using Protection Products programs. When a backup is executed, a record containing information related to the backup is created within the backup catalog. When backed-up data is restored, Protection Products programs reference the backup catalog information, and execute accordingly.
backup ID	A part of the information contained in the backup catalog. Backup IDs are used to differentiate backup data, and are assigned automatically when backups are performed using Protection Manager.
backup information	Various information that is recorded in a backup catalog when Protection Manager programs are used to perform a backup.
basic disk	A standard physical disk in Windows. You cannot create volumes that span multiple disks.
basic volume	A logical volume allocated to a basic disk. Protection Manager only supports configurations with one basic volume for each basic disk.
CLI	command line interface
cluster software	Software used to enhance overall system performance by using multiple instances of the system. The following software can be used in cooperation with Protection Manager: Microsoft Cluster Service, VERITAS Cluster Server.
copy group	A copy group consists of a primary volume and secondary volume that are synchronized and split using volume replication functionality and RAID Manager functionality. A copy group can also be called a pair volume.
copy group map file	One of the map files making up a dictionary map file. Copy group map files are used to keep track of the mapping information between the primary volume and its corresponding secondary volumes.
core map file	One of the map files making up a dictionary map file. Core map files are used to keep track of the mapping information from the mount point directory on the file system to the disk number within the RAID device.
data retention utility	A generic name for the following when they do not need to be distinguished: -Data Retention Utility -Hitachi OPEN LDEV Guard
DBMS	An abbreviation for Database Management System. A DBMS consists of programs that manage databases.

DKC	Disk Controller
dictionary map file	A file required for automating backup processing using Protection Manager. A dictionary map file keeps track of mapping information, from the objects to be backed up to the RAID devices. A dictionary map file consists of three types of map files, and a backup catalog. –Application map file –Core map file –Copy group map file –Backup catalog
disk group	With VxVM, one or more logical devices can be grouped as individual units. Disk groups can consist of one or more volumes, regardless of the content of the logical volumes that comprise the disk group. In a VERITAS Volume Manager for Windows environment, a configured disk group is recognized as one or more disk sets, and is handled as a minimum unit when backing up or restoring data.
disk set	For Protection Products programs, in a dynamic disk configuration, a group consisting of one or more dynamic disks and one or more logical volumes allocated to the dynamic disks is called a *disk set*. In LDM environments, you can perform backups or restorations on units as small as disk groups.
dynamic disk	Dynamic disk functionality can be used to create a disk set by using the logical disk manager (LDM) that comes with Windows, or to create a disk group by using VxVM. The term *dynamic disk* indicates physical disks composing a disk group or disk set. You can combine or split one or more dynamic disks to create a logical volume (dynamic volume).
dynamic volume	A logical volume created by joining or splitting one or more dynamic disks and then assigning them.
failover	The action of automatically switching to a backup system when an error occurs in a system that has been duplicated (that is, multiple instances of the system were made) using cluster software.
freezing a database	DBMS temporarily stops I/O operations for the disk on which the database is stored. Transactions from applications are controlled by DBMS until the database is thawed. If a database is frozen when a paired volume is synchronized, the primary and secondary volumes will be identical. Backing up the database in this state will ensure that its integrity is maintained during backup processing.
GUI	graphical user interface
GUID	global unique identifier
HiCommand Protection Manager	A general name for the following products: HiCommand Protection Manager Copy Controller HiCommand Protection Manager for SQL HiCommand Protection Manager for Exchange HiCommand Protection Manager Console
HiCommand Protection Manager Console	This program allows the backup and restoration functionality provided by Protection Products programs to be used from a screen.

HiCommand Protection Manager Copy Controller	A program product that performs backups and restorations for file systems. There are also prerequisite programs for backing up databases such as SQL Server.
HiCommand Protection Manager for Exchange Server	A program that backs up and restores Exchange Server databases. HiCommand Protection Manager for Exchange is an optional program product.
HiCommand Protection Manager for SQL Server	A program that backs up and restores databases on SQL Server. HiCommand Protection Manager for SQL is an optional program product.
IIS	Microsoft® Internet Information Services
IPF	Itanium Processor Family
LAN	local area network
LDM	An abbreviation for *Microsoft Logical Disk Manager*. LDM is the standard Windows logical volume manager.
LU	logical unit
LVM	logical volume manager
metadata	Data consisting of information about other data: such as the configuration and attributes of a file system.
MSCS	Microsoft Custer Service
NTFS	NT file system
OS	operating system
pair volume	A pair of physical volumes that are mirror-controlled by RAID Manager and the volume replication functions.
RAID Manager	Software used to control Hitachi Lightning™/Hitachi Thunder™ Series storage subsystems. RAID Manager is installed on a host connected to the storage subsystem.
Remote copy functionality	In this manual, the volume copy functionality that replicates a volume to a remote site, such as TrueCopy or UR, is generically called *volume copy between storage subsystems* (or *remote copy functionality*). By backing up data to a remote site, you can restore the data from the remote site if a problem occurs in a local site storage subsystem. This improves data stability. For example, even if the data at a local site is damaged due to a disaster, you can use the data in a remote site to recover it in a short period of time.
SAN	storage area network
Setup GUI	Functionality that allows you to set up an operating environment for Protection Manager via a GUI. Using the GUI means it is not necessary to execute commands or edit the configuration files. Setup GUI becomes available when Protection Manager Console is installed.

storage group	A method provided by Exchange Server for managing a grouping of multiple databases. Multiple databases can be managed together since databases within the same group use a common transaction log.
system log	Log information that is produced by the operating system for purposes of status and trouble reporting. The system log is output to the Windows Event Log file.
target definition file	A user-defined configuration file used to specify multiple operation targets. The file enables users to apply Protection Manager commands to multiple pre-specified targets.
target ID	A number used to differentiate the devices connected via a SCSI bus. It is also called a SCSI ID.
VDI	An abbreviation of virtual device interface, an application interface for volume replication functionality provided by SQL Server.
Volume dynamic recognition	The volume dynamic recognition functionality enables or disables concealment of the physical volumes in the storage subsystem connected to a server from the server, by executing the Protection Manager commands. This enables control of the access to the physical volume from a server, and prevents operational mistakes.
volume replication functionality between storage subsystems	A generic name used in this manual for TrueCopy and UR. The volume replication functionality between a storage subsystem copies data between two storage subsystems, and transfers high-speed data without going through a LAN. This allows you to transfer data, quickly and independent of a host, to a remote site during data center emergencies, migrations, and mergers.
volume replication functions	A generic name used for high-speed volume replication functions in the subsystem (e.g., ShadowImage and TrueCopy). The mirror control function provided by the subsystem can be used to quickly create a replica of the volume without passing through the LAN.
VSS	An abbreviation of Volume Shadow Copy Service, one of the storage support functions that can be used on Windows Server 2003. Normally, when database data stored on a storage subsystem volume is copied to another volume, access from hosts (database applications) must be suspended and the database must be frozen. By using VSS, since the operating system functionality controls transactions for a database application and I/O operations for a volume, database can be frozen. This provides higher reliability for backup operations. To use VSS to freeze a database application, however, VSS must support the database application. Protection Manager programs handle Exchange Server 2003 as a database application supported by VSS, and support backup and restoration of the file systems used by VSS.
VxVM	An abbreviation for the logical volume manager VERITAS Volume Manager.

Index

A

actions taken for message (KAVX5137-E), 517
active-active, 25, 127
Active-Passive, 24, 125
  cluster configuration, 124
adjusting
  number of log files for detailed trace log, 472
application map file, 499
applying transaction logs to recover databases from backup at a particular time, 321
automatic creation of transaction log list file, 152
automatic execution of extended commands, 168

B

backing up, 5
  Protection Products operating environment, 200
  SQL Server database, 385
  transaction logs in a cascade configuration, 388
backing up and restoring a file system by using a backup management product that is unable to link with protection products, 265
backing up and restoring data between volumes processing flow (for file system), 240
backing up and restoring multiple generations by locking a copy group, 330
backing up and restoring storage group using backup management product that is unable to link with protection manager, 453
backing up and restoring transaction log backup file, 323
backing up data
  using VSS, 21
backing up file system to secondary volume, 241
backing up file system to tape, 246
backing up secondary volume
  on remote destination, 34
backing up storage group by using user script system configuration, 436
backing up storage group to secondary volume, 423
backing up storage group to tape, 429
backing up transaction logs, 317
backup and restoration precautions, 213
backup catalog, 6
backup catalog file, 499
Backup Exec backup job script file, 146
backup ID, 6
backup ID file, 500
backup management product
  setup for linkage with, 143
backup server
  backup to tape, 510
Backup with VSS, 56
BACKUP_OBJECT, 174, 179, 184
BACKUP_SERVER, 140
basic commands, 208
batch definition files, 500
before using
  extended commands, 168
  log-shipping function, 358
BUSY_RETRY_TIME, 115
BUSY_RETRY_WAIT, 113, 115

C

cascade configuration, 346, 365
cascade configuration (to back up SQL Server database), 34
CG_MP.conf, 501
changing FTP user, 169
checking copy-group list file, 188
checking FTP service (if you are going to back up to tape), 188
checking temporary directory for extended commands, 187
checkpoint files, 42, 70
CLU_MSCS_RESTORE, 104
CLU_RETRY_TIME, 100
CLU_RETRY_WAIT, 100
cluster configuration definition file, 501
cluster resources
  offline, 8
  online, 8
coding rules for user script files, 373
collecting data
  error info. data, 470
  Windows, 470
COM_RETRY_TIME, 102
COM_RETRY_WAIT, 102
command execution condition, 219
concurrent command execution when backing up an Exchange database, 230
concurrent command execution when backing up and restoring a file system, 227
concurrent command execution when backup is of SQL Server database, 230
concurrent execution of commands, 227

concurrent execution of common commands, 228
concurrent execution of tape-related commands, 229
concurrent execution of utility commands, 229
condition and precaution for dynamic disk configuration, 71
configuration and precaution for data backup and restoration between storage subsystem, 26
configuration definition file for tape backup, 499
configuration file
Protection Products, 100
configuring Protection Manager for linkage with RAID Manager
setting up operation mode when automatically selecting copy groups, 120
settings for using secondary volume dynamic recognition functionality, 119
specifying retry count and retry interval for RAID Manager command busy status retries checking, 115
specifying retry count and retry interval that are appropriate to operation style, 115
CONFIRM_GENERATION_IDENTICAL, 120
contents of list file, 151
contents of transaction log list file, 152
copy group batch definition file, 500
copy group map file, 499
copy group mount file, 501
copy group names, 149
copy parameter definition file, 115
copy parameters definition file, 501
COPY_SIZE, 117
core map file, 499
correspondence between extended command and basic command, 209
creating
copy-group list file, 149
database configuration definition file, 134
database list file or mount point directory list file, 150
dictionary map file, 123, 124
list file, 149
non-cluster configuration, 123
operation ID, 172
transaction log list file, 151
creating an operation definition file
when backup is of SQL server database, 180
creating database configuration definition file
when backup is of SQL server database, 134
creating dictionary map file
creation in Active-Active cluster configuration, 126
creating host environment setting file, 171
creating operation definition file
when backup is of SQL Server database, 177
when the backup is of a file system, 173
creating operation definition files
when backup is of Exchange database, 182

D

data
to collect at problem, 462
data file (for SQL server database), 61
data files (for Exchange database), 42, 70
data management
using a remote site, 9
data to be collected, 462
database configuration definition file (instance name.dat), 499
database subject to backup
requirements and precaution, 61
DB_SERVER_NAME, 174, 179, 184
dealing with problems, 460
DEFAULT.dat, 107, 499, 500
deleting backup data, 268
deletion of unnecessary file, 169
deployment of operation definition file
when backup is of file system, 173
when backup is of SQL Server database, 178
deployment of operation definition files
when backup is of Exchange database, 182
detailed trace log
investigating, 474
detailed trace log information
adjusting number of log files, 472
adjusting output level, 472
determining a mount point and mount target
when mounting a secondary volume, 163
DEVICE_DETACH, 120
dictionary map
application map file, 5
copy group map file, 6
core map file, 5
dictionary map file, 5, 499
precaution regarding update of dictionary map file, 219
recreating, 481
troubleshooting, 477
verifying, 478
dictionary map file directory, 123
dictionary map synchronization option
setting up, 103
disaster recovery, 10
disk group, 46
disk group configuration that does not allow backup (dynamic disk configuration), 221

disk signature, 517
distribution database, 410
distribution target server
  setting, 363
DRM_DB_SYNC, 103
drm_output.log, 461
drm_output.log.old, 461
drm_script.log, 461
drm_script.log.old, 461
drmdbsetup utility, 125, 127
dynamic disk configuration
  using, 75
dynamic disks
  backing up and restoring by recognizing secondary volume dynamically, 54
  backing up and restoring by recognizing secondary volume statically, 55
  using in LDM, 53
  using in VERITAS Volume Manager for Windows, 54
dynamic volumes
  expanding, 55

E

entries in user script file
  [CMD], 191, 374
  [FINISH_PROC], 190, 374
  [POST_PROC], 191
  [PRE_PROC], 190
  [RESYNC_PROC], 190, 374
  [SPLIT_PROC], 190, 374
  CMDLINE, 191, 375
  END_CODE, 192, 375
  ENV, 192, 375
  LOCAL_BACKUP, 190, 374
  LOCATION, 193
  PARENT_STAT, 193
  TIMEOUT, 192, 375
environment configuration example
  creating dictionary map file, 493
environment setup
  for linkage with RAID Manager, 107
  for linkage with the backup management product, 143
  for RAID Manager, 92
  operations of Protection Products, 100
environment variable PATH, 79
Examining the Contents of the drm_inst.log File, 84
example of backing up SQL Server database in multi-target configuration, 350
example of creating dictionary map file, 493
example of creating operation definition file in Windows cluster configuration, 175
example of creating operation definition files when backup is of Exchange database, 185
example of environment configuration, 483
  example of registering SQL Server information, 494
  example setting for copy-group list file, 498
  example setting for host environment setting file, 497
  example setting for linkage to backup management product, 495
  example setting for RAID Manager configuration definition file, 488
  example settings for operation definition file, 497
  example settings for Protection Products configuration definition File (init.conf), 491
  server configuration, 485
  storage subsystem configuration, 486
example of file system operation
  backing up a file system to remote site, 259
  backing up and restoring data between volumes, 239
  recovering a file system from a remote site to local site, 264
  restoring a file system on remote site, 263
  restoring file system to local site, 261
example of operation using transaction log backup (obtaining transaction logs after failure), 319
example of restoring SQL Server database in multi-target configuration, 351
example of the configuration definition file for tape backup, 496
example registering information for linkage to backup management product, 495
example setting
  RAID Manager-linkage definition file (DEFAULT.dat), 492
Exchange database
  products, 69
Exchange database product, 42
executing
  extended commands, 167
executing multiple drmmediabackup or drmmediarestore commands concurrently, 232
EXG_VERIFY_RETRY_COUNT, 140
EXG_VERIFY_RETRY_INTERVAL, 140
extended command trace log file, 461
extended commands, 208
  troubleshooting, 461

F

features of Protection Products, 3
file share
  backing up and restoring a file system, 266
  backing up and restoring storage group, 455
  backing up or restoring SQL Server database, 395
  backing up SQL Server database by using file share, 396
  example of backing up file system, 266
  example of backing up storage group, 455
  example of restoring a file system, 267
  example of restoring database by using file share, 397
  example of restoring storage group, 456
  preparing to back up or restore by using file share, 396
  preparing to back up or restore file system, 266
  preparing to back up or restore storage groups, 455
files configuring database
  allocating, 65
files used in Protection Products, 499
format of operation definition files
  when backup is of Exchange database, 183
  when backup is of file system, 174
  when backup is of SQL Server database, 178
FTP service, 170
FTP_HOME_DIR, 174, 179, 184
FTP_SUB_DIR, 174, 179, 184
function of Protection Products
  clustering support, 8
  generation management, 10
  pair management using copy groups, 7
  reducing operation workload using command, 8
  resource management using dictionary map file, 5
functions of Protection Products, 5

G

generation management
  backing up and restoring in cascade configuration or Multi-target configuration, 365
  example of operation using lock functionality, 331
  locking, 11

H

HORCMINST, 108
HORCMINST_AUX, 108
horcmn.conf, 499
  example settings, 488
Host environment setting file, 500
host.dat, 500
HOST_ROLE, 171

I

incorrect data file
  restoring, 481
Information pertaining to Protection Manager Console, 466
init.conf, 100, 500
  example settings, 491
INST_PATH, 145, 146
installation
  destination of Protection Products, 78
  volume replication functionality required on storage subsystem, 15
installing
  Protection Products, 78
INSTALLPATH, 122
instance number
  setting up, 107
INSTANCE_NAME, 174, 179, 184
introduction to Protection Products, 2
investigating
  collected data, 473
  detailed trace log, 474
  standard output logs, 473
  trace logs, 473

K

KAVX5105-W, 513
KAVX5137-E, action for, 517
key definition file
  restoring, 479
key file
  restoring, 479

L

LDM
  using dynamic disks, 53
list file storage location, 151
local, 10
local site, 9
locking copy group, 332
log file for extended commands, 467
logical volume manager, 14
log-shipping function, 33

M

mail send definition file, 198
mail -sending definition file, 501
mail.conf, 198, 501
MAX_LOG_LINES, 171

messages
actions for KAVX5137-E, 517
meta file, 61
metafile allocation, 67
metafiles
allocating, 66
missing data file
restoring, 480
mount point directory batch definition files, 500
MOUNT_POINT, 145, 146
MSG_OUTPUT, 171
multiple database configuration files to one volume
allocating, 65
multi-target configuration, 346, 365
multi-target configuration (to back up a SQL Server database), 35
mutual standby type (Active-Active)
cluster configuration, 25

N

name identifying the generation, 10
note
using VERITAS cluster server for Windows, 69, 75
notes on starting and stopping RAID Manager, 121

O

online backup, 3
operation
for SQL Server database, 269
operation for exchange database, 415
operation for file system, 235
operation definition file, 500
operation details
in Windows, 467
operation example
backing up a file system to tape, 246
backing up and restoring a file system using a backup management product that is unable to link with protection products, 265
backing up file system by using user script, 252
backing up file system to secondary volume, 241
performing backup and restoration between remote site, 257
restoring a file system to primary volume, 241
restoring file system from tape, 249
operation ID, 172
creating, 172
operation precaution, 213
operation standby type (Active-Passive)
cluster configuration, 24
operation using SQL Server transaction log, 311
operations for backing up and restoring a file system, 236
order of applying transaction log backups (1), 283, 321
overview
of Protection Products, 1

P

pair
checking the status, 110
pair volume, 7
PAIR_CREATE, 119
PARALLEL_COUNT, 145
permissions required for users executing commands, 223
POLICY, 145
precaution
executing commands, 223
executing commands in a cluster environment, 217
executing commands in cluster environment, 237
simultaneous execution of commands, 217
precaution for disk group configuration, 49
precaution for dynamic volume expansion, 55
precaution for importing backup information, 219
precaution for installing VERITAS Volume Manager for Windows in Windows LDM environment, 55
precaution for mounting data to backup server, 218
precaution on database verification, 420
precaution on forced termination of command, 238
precaution on operation, 213
precaution on recovery storage group, 420
precaution regarding backup server, 217
precaution when backing up and restoring multiple database, 274
precautions for restoring file system, 237
precautions regarding application of transaction logs, 282
precautions when mounting, 217
preparation
using Protection Products, 89
preparation for performing backing up and restoring in cascade configuration or Multi-target configuration, 367
prerequisite for using dynamic disk in LDM, 53
prerequisite for using dynamic disk in VERITAS Volume Manager for Windows, 54

primary volume
replacing (in Windows), 504
restoring from tape (QuickShadow), 511
restoring from tape in QuickShadow configuration, 509
primary volume disk
replacing (in Windows), 504
primary volume disks
replacing while Protection Products Programs run, 503
procedure for restoring from tape directly to primary volume in QuickShadow configuration, 509
Protection Manager
version of, 15
Protection Manager for Exchange, 2
Protection Manager for SQL, 2
Protection Manager service
local system account, 81, 82
restarting, 234
right to log in as service, 81, 82
starting, 233
stopping, 234
user who executes Protection Manager service, 81
Protection Products
configuration definition file, 100
cooperative software, 17
functions, 5
installation destination, 78
overview, 1
overview of functions, 4
server, 86
system requirements, 14
Protection Products commands, 208
Protection Products configuration definition file, 500
Protection Products log file, 461
Protection Products Operation, 207
publication database, 410

Q

QuickShadow
configuration example for restoration from tape, 509

R

RAID Manager
environment setup, 92
setup for linkage with, 107
RAID Manager configuration definition file, 92, 499
RAID Manager linkage definition file, 500
RAID Manager-linkage definition file, 107
RAID Manager-linkage definition file (DEFAULT.dat)
example setting, 492
recognizing secondary volume dynamically
backing up and restoring dynamic disks, 54
recognizing secondary volume statically
backing up and restoring dynamic disks, 55
recovering operating environment, 204
recovering SQL Server database from last backup data, 282, 321
recovering the SQL Server database from backup data prior to last backup data, 283, 322
recovery option
setting up, 104
recreating
dictionary map file, 481
remote, 10
remote copy functionality, 10
remote site, 9
renaming or discarding once created operation ID, 169
replacing
primary volume (in Windows), 504
primary volume disk (in Windows), 504
secondary volume disk (in Windows), 507
replacing primary and secondary volume disks while Protection Products programs are running, 503
requirement and precaution for dynamic disk configuration, 68
requirement and precaution for subject to backup
Exchange database, 69
SQL Server database, 61
requirement and precaution for using VSS, 57
requirements
for Protection Products host, 14
operating server, 15
requirements and precautions when disk group is configured in Windows, 53
restoration
restoring in units of information stores, 71
restoration while cluster resources are offline, 8
restoration while cluster resources are online, 8
RESTORE_DELAY_RETRY_TIME, 113
RESTORE_DELAY_RETRY_WAIT, 113
RESTORE_RETRY_TIME, 112
RESTORE_RETRY_WAIT, 112
restoring, 5
incorrect data file, 481
key definition file, 479
key file, 479
missing data file, 480
SQL Server database, 388
restoring a file system to primary volume, 241

restoring backup data obtained using VSS in units of information stores, 71
restoring data in units of information stores, 435
restoring data to different SQL Server instance, 399
restoring data to SQL Server instance that is different from backup source, 399
restoring file system from tape, 249
restoring storage group from tape, 434
restoring storage group on primary volume, 423
RESYNC_RETRY_TIME, 111
RESYNC_RETRY_WAIT, 112
retry count, 100
retry interval, 100
retry time when error occurs, 231
RETRY_TIME, 110
RETRY_WAIT, 110
right to log in as service, 233
roll-forward, 439

S

Sample user script files, 377
saving transaction log that have previously been backed up, to tape or other storage media, 323
SCHEDULE, 145
SCRIPT_FILE, 146
secondary volume disk
  replacing (in Windows), 507
server
  from which Protection Products is deleted, 86
SET_DRM_HOSTNAME, 174, 179, 184
setting as operating server
  distribution target server, 363
setting up environment for Protection Manager, 90
setting up FTP service, 170
setting up linkage with backup management product
  creating configuration definition file for tape backup, 144
  registering information for linkage with backup management product, 143
setting up Protection Manager
  specifying retry count and retry interval for command execution status retries checking, 102
setting up RAID Manager
  backing up multiple primary volumes in single generation, 94
  backing up multiple primary volumes to multiple generations, 95
  backing up one primary volume to multiple generations, 93
  backup or restore between storage subsystems, 96
  setting up multi-target or cascade configuration, 99
setting up restoration while cluster resources are in online state, 104
settings for sending mail, 198
setup
  dictionary map synchronization option, 103
  for checking status of cluster resources, 100
  for checking status of pair, 110
  instance number, 107
  recovery option, 104
  retry count, 110
  retry interval, 110
  track size for copying data, 117
  volume pair creation, 119
Setup GUI, 2
SOL Server database
  precaution for Allocating database file, 65
  precaution for allocating metafile, 66
  precaution to take when multiple database configuration files are allocated to volume, 65
specifying how to mount a secondary volume, 154
specifying how to mount a secondary volume (method to make OS dynamically recognize secondary volume), 155
specifying how to mount a secondary volume (method to make OS statically recognize secondary volume), 160
specifying how to start extended commands, 170
specifying installation path for RAID Manager, 122
specifying settings for a cluster configuration, 129
specifying settings for uning VSS
  when RM Shadow Copy Provider version is 01-02-03/03 or later, 141
specifying settings for using VSS
  when RM Shadow Copy Provider version is earlier than 01-02-03/03, 142
specifying settings for using VSS (when file system or exchange database), 139
specifying settings related to linkage with SQL Server, 136
specifying settings when the file server or database server is in a cluster configuration, 129
specifying SQL Server automatic recovery time, 136
specifying SQL Server login timeout option, 136
SPLIT_RETRY_TIME, 112
SPLIT_RETRY_WAIT, 112

SQL Server database
backing up, 385
backing up and restoring data between volumes, 290
backing up and restoring in Multi-target configuration, 346
backing up SQL Server database by using user script, 308
backing up SQL Server database to remote site, 338
backing up SQL Server database to secondary volume, 292
backing up SQL Server database to tape, 301
database status when restoring and recovering, 276
example of operation using transaction log backup, 313
loading status, 276
operations for backing up and restoring an SQL Server database, 270
performing backup and restoration using backup management product that is unable to link with protection products, 393
performing backup and restore operation between remote sites (remote copying), 335
performing restoration and recovery on host other than the backup host, 353
precaution regarding backing up transaction log, 278
precaution regarding sequence of transaction log, 280
precaution when backing up and restoring database, 271
precaution when recovering database, 273
precautions when restoring database, 272
recovering data from remote site to local site, 344
requirement for transaction log backup, 278
restoring, 388
restoring SQL Server database from tape, 304
restoring SQL Server database to local site, 341
restoring SQL Server database to primary volume, 292
restoring SQL Server database to remote site, 343
standby status, 276
using Log-Shipping function of SQL Server database, 358
when there is any problem in the latest backup data, 283, 322
SQL Server database product, 61
SQL Server database requirement for executing command, 284
SQL Server database status for executing command, 285
SQL Server database type for executing command, 289
SQL Server replication function, 409
SQL Server service status for executing command, 284
SQL_AUTORECOVERY_TIME, 138
SQL_CHECKDB, 104
SQL_LOGIN_TIMEOUT, 138
SQL_QUICK_BACKUP, 102
stopping retry a operation caused by commands that cannot execute concurrently, 231
storage group
backing up and restoring between remote sites (remote copy), 444
backing up and restoring data between volumes, 421
backing up storage group by using user script, 436
backing up to remote site, 446
restoring to local site, 450
storage subsystem
required volume replication functionality, 15
storage subsystem requirements, 15
subscription database, 410
SVC_RETRY_TIME, 101
SVC_RETRY_WAIT, 101
syntax conventions, 18
system configuration, 21
system requirements, 14
system to back up and restore data between volumes
configuring, 22
system to restore data on host that differ from one used for backup (to back up a SQL Server database)
configuring, 31
system to store backed-up data to tape, and restore the data from tape
configuring, 23
system to use SQL Server's log-shipping function (to back up a SQL Server database)
configuring, 33

T

tape
backup to tape on backup server, 510
restoring to primary volume, 511
to primary volume in QuickShadow configuration, 509
tape device
backup and restoration (Exchange), 425
backup and restoration (file system), 243

backup and restoration (SQL Server), 296
TARGET_NAME, 174, 179, 184
template catalog, 371
temporary directory for extended commands, 187
temporary file, 500
track size for copying data
setting up, 117
transaction log batch definition file, 500
transaction log file, 500
transaction log file (for SQL server database), 61
transaction log files (for Exchange database), 42, 70
transaction log list file storage location, 151
troubleshooting, 459
dictionary map file error, 477
extended commands, 461
troubleshooting installation errors, 84
type of setup, 80

U

uninstalling
Protection Products, 86
unlocking copy group, 334
updating
dictionary map file, 153
user script
coding rules for, 189
creating, 189
creating, to perform backup in cascade configuration, 372
overview of, 189
user script example (for file system), 254
user script example (for SQL Server database), 310
user script example (for storage group), 437
user script file, 372, 501
using drmsqlrecovertool dialog box to recover SQL Server database, 317
using Protection Products
preparation, 89

V

verifying
dictionary map file, 478
VERITAS Cluster Server for Windows
configuration
using, 69, 75
VERITAS Volume Manager for Windows
using dynamic disks, 54
VERITAS Volume Manager for Windows in Windows LDM environment
installing, 55
volume configuration used by Protection Products when backing up or restoring data, 44
volume copy function between storage subsystems, 10
volume pair creation
setting up, 119
volume replication functionality, 15
required on storage subsystems, 15
VSHTCHOMRCF_MUN, 142
VSHTCHORCMINST_LOCAL, 142
VSHTCHORCMINST_REMOTE, 142
VSS
requirements and precautions, 58
restoring in units of information stores, 71
used for backing up data, 21, 56
using, 58
VSS definition file, 139, 501
vsscom.conf, 139, 501

W

when installation fails, 84
when RM Shadow Copy Provider version is 01-02-03/03 or later
specifying settings for using VSS, 141
when RM Shadow Copy Provider version is earlier than 01-02-03/03
specifying settings for using VSS, 142
when VERITAS cluster server is in use
creating cluster information configuration definition file (when VERITAS Cluster Server is in use), 131
WRITER_TIMEOUT_RETRY_COUNT, 140
WRITER_TIMEOUT_RETRY_INTERVAL, 140