

3PAR Recovery Manager 3.0.0 for Microsoft SQL Server User's Guide

3PAR, Inc.
4209 Technology Drive
Fremont, CA 94538
U.S.A.

Part No. 320-200108 Rev A
December 2008

Revision Notice
This is the first release of this manual. A complete revision history is provided at the end of this manual.

Changes
The material in this document is for information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, 3PAR Inc. assumes no liability resulting from errors or omissions in this document or from the use of the information contained herein.
3PAR reserves the right to make changes in the product design without reservation and without notification to its users.

Updates to the Documentation at 3PAR Central
Any updates to this document, or to other 3PAR technical documents, can be found by logging in to 3PAR Central's Document Control System from 3PAR's Support page at http://support.3PAR.com

3PAR Technical Support and Services
Contact your local service provider for technical support and services at: http://www.3PAR.com/services.html.

Sales and Ordering Information
For sales and ordering information, contact:

3PAR Inc.
4209 Technology Drive
Fremont, CA 94538 U.S.A.
Telephone: 510-413-5999
Fax: 510-413-5699
E-mail: salesinfo@3PAR.com

Reader Comments and Suggestions
Please E-mail your comments and suggestions about this document to: ReaderComments@3PAR.com

Copyright
Printed Material
Copyright © 2005-2008 3PAR Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without the prior written consent of 3PAR Inc., 4209 Technology Drive, Fremont, CA 94538.

Trademarks
3PAR, InServ, InForm, InSpire and Serving Information are registered trademarks of 3PAR Inc.
Microsoft Windows, Microsoft SQL server is a trademark of Microsoft, Inc.
VERITAS is a registered trademark of VERITAS Software Corporation.
All other trademarks and registered trademarks are owned by their respective owners.

Table of Contents

1 Introduction

1.1 Audience 1.1

1.2 Related Documents 1.2

1.3 Organization 1.2

1.4 Typographical Conventions 1.3

1.5 Advisories 1.4

2 Overview of Recovery Manager

2.1 Microsoft Volume Shadow Copy Service 2.1

2.2 How VSS Works with Microsoft SQL Server 2.3

2.3 Recovery Manager Layout 2.4

2.4 Supported VSS Components 2.5

2.5 System Requirements 2.6

2.6 Recovery Manager Requirements 2.6

2.7 Features Supported 2.7

2.8 Features Not Supported 2.8

3 Installation and Deinstallation

3.1 Installing Recovery Manager 3.1

3.2 Verifying the Installation 3.6

3.3 Deinstalling Recovery Manager 3.6

3.4 Understanding the Registry 3.7

3.5 Migrating from Version 2.2.3 3.8

4 Configuring NetBackup for Recovery Manager

4.1 Overview 4.1

4.2 Defining a NetBackup Policy 4.2

5 Configuring Recovery Manager

5.1 Configuring the Database 5.1

5.2 Configuring Recovery Manager 5.2

5.2.1 Installation Location 5.2

5.2.2 Search Path 5.2

5.3 Change Port Procedure 5.2

5.4 Change Protocol Procedure 5.3

5.5 Configuring the Host Computer and the 3PAR InServ Storage Server 5.4

6 Using the Recovery Manager GUI

6.1 Overview of the Recovery Manager GUI 6.2

6.1.1 Menu Bar 6.3

6.1.2 Node Tree Pane 6.3

6.1.3 Informational Right Pane 6.3

6.1.4 Displaying Database Information 6.3

6.2 Starting the GUI 6.5

6.3 Stopping the GUI 6.5

6.4 Server Registration 6.5

6.5 Setting the Recovery Manager Policy Configuration 6.9

6.6 Creating Instance or Database Virtual Copy 6.10

6.7 Creating a Remote Virtual Copy 6.14

6.7.1 Creating a Remote Virtual Copy for an Instance 6.15

6.7.2 Creating a Remote Virtual Copy for a Database 6.15
6.8 Mounting a Virtual Copy 6.17
6.9 Unmounting a Virtual Copy 6.19
6.10 Backing Up a Virtual Copy Using Veritas NetBackup 6.19
6.10.1 Backup at Instance Level 6.19
6.10.2 Backup at Database Level 6.22
6.11 Creating a Backup at the Timestamp Level 6.24
6.12 Deleting Backup from the Repository 6.25
6.13 Restoring an Instance or Database from Virtual Copy using Volume Restore 6.25
6.14 Restoring an Instance or Database from Virtual Copy Using File Copy 6.27
6.15 Restoring a Virtual Copy Using Veritas NetBackup 6.28
6.16 Removing a Virtual Copy 6.30

7 Using the Recovery Manager CLI

7.1 Recovery Manager Command Line Interface (CLI) 7.2
7.2 Recovery Manager Commands 7.3
RMSql backup 7.4
RMSql create 7.6
RMSql deletebackup 7.9
RMSql analyze 7.10
RMSql display 7.12
RMSql mount 7.14
RMSql remove 7.16
RMSql restore 7.17
RMSql unmount 7.20
RMSql policy 7.21
RMSql listbackup 7.23
RMSql report 7.24

RMSql repository 7.25

RMSql port 7.26

RMSql portconfig 7.27

RMSql protocol 7.28

RMSql time 7.29

RMSql timeconfig 7.30

7.3 Running Recovery Manager as a Scheduled Job 7.30

8 Recovery for Microsoft SQL Server

8.1 Recovery 8.1

8.1.1 Point-In-Time Recovery 8.2

8.1.2 Point-Of-Failure Recovery 8.3

8.1.3 SQL Instance Level File Copy Restore in an MSCS Environment 8.5

9 Remote Copy for Recovery Manager

9.1 Purpose of Remote Copy 9.1

9.2 How it works 9.1

9.3 How to Setup the Remote Copy Feature 9.3

9.3.1 Set Up Remote Copy Group for Databases 9.3

A Troubleshooting

IX Index

RH Revision History

1 Introduction

In this chapter

1.1 Audience 1.1

1.2 Related Documents 1.2

1.3 Organization 1.2

1.4 Typographical Conventions 1.3

1.5 Advisories 1.4

This guide provides the information you need to install, deinstall and use the 3PAR Recovery Manager 3.0.0 for Microsoft SQL Server on the Microsoft Windows operating system.

1.1 Audience

This guide is designed for System Administrators and Microsoft SQL Server Administrators who are responsible for backing up Microsoft SQL Server databases and who understand Microsoft™ Windows™ and Microsoft SQL Server™.

1.2 Related Documents

The following documents also provide information related to the InServ Storage Server:

For Information About…	Read the…
Using the InForm Command Line Interface (CLI) to configure and manage the InServ Storage Server	*3PAR InForm OS CLI Administrator's Manual*
Using the InForm Management Console to configure and administer InServ Storage Servers	*3PAR InForm OS Management Console Online Help*
Identifying InServ Storage Server components and fixing problems	3PAR *InForm OS Messages and Operator's Guide*
Understanding the InServ Storage Server and its features	*3PAR InForm OS Concepts Guide*
Shadow copies using 3PAR VSS Provider for Microsoft Windows	*3PAR VSS Provider for Microsoft Windows User Guide*

1.3 Organization

This guide is organized as follows:

This chapter provides an overview of this guide, including intended audience, related documentation, typographical conventions, and advisories.

Chapter 2, *Overview of Recovery Manager*, provides an overview of virtual copies and how Recovery Manager interacts with Microsoft SQL Server.

Chapter 3, *Installation and Deinstallation*, describes how to install, verify, and deinstall Recovery Manager.

Chapter 4, *Configuring NetBackup for Recovery Manager*, contains instructions for configuring Veritas NetBackup for use with the 3PAR Virtual Copy DBA.

Chapter 5, *Configuring Recovery Manager*, contains instructions for configuring the database and the 3PAR Virtual Copy DBA.

Chapter 6, *Using the Recovery Manager GUI*, contains procedures for working with Recovery Manager through the GUI (Graphical User Interface).

Chapter 7, *Using the Recovery Manager CLI*, contains procedures for working with Recovery Manager through the CLI (Command Line Interface).

Chapter 8, *Recovery for Microsoft SQL Server*, discusses methods for backing up data and performing recovery using virtual copies.

Chapter 9, *Remote Copy for Recovery Manager*, contains procedures for configuring the Recovery Manager to work with the 3PAR Remote Copy solution.

Appendix A, *Troubleshooting*, lists some common issues that may arise and suggests solutions to help solve the problems.

This guide also contains an index and revision history for your reference.

1.4 Typographical Conventions

The following typographical conventions are used in this guide:

Typeface	Meaning	Example
ABCDabcd	Used for dialog box elements such as titles and button labels.	Enter your system name in the **Value** box and click **OK**.
ABCDabcd	Used for file names, paths, and screen output, and for text you are to enter.	`Found < 12 > 73G disks.` Enter `cli` at the Windows command prompt.

Typeface	Meaning	Example
ABCDabcd	Used to contrast your input with system output.	`cli% removevv VV1` `Removing vv VV1.`
ABCDabcd **ABCDabcd**	Used for variables in file names, paths, and screen output, and variables in user input.	[root@(*systemID-nodeID*)root] `To continue Enter your system name ==>` **systemname**

1.5 Advisories

To avoid injury to people or damage to data and equipment, be sure to observe the cautions and warnings in this guide. ***Always be careful when handling any electrical equipment.***



NOTE: Notes are reminders, tips, or suggestions that supplement the procedures included in this guide.



CAUTION: Cautions alert you to actions that can cause damage to equipment, software, or data.



WARNING: Warnings alert you to actions that can cause injury to people or irreversible damage to data or the operating system.

2 Overview of Recovery Manager

In this chapter

2.1 Microsoft Volume Shadow Copy Service **2.1**

2.2 How VSS Works with Microsoft SQL Server **2.3**

2.3 Recovery Manager Layout **2.4**

2.4 Supported VSS Components **2.5**

2.5 System Requirements **2.6**

2.6 Recovery Manager Requirements **2.6**

2.7 Features Supported **2.7**

2.8 Features Not Supported **2.8**

2.1 Microsoft Volume Shadow Copy Service

Microsoft includes VSS (Volume Shadow Copy Service) in Windows 2003 and 2008 to simplify the enterprise's storage environment. It provides a framework for creating a point-in-time copy of either a single volume or multiple volumes.

Microsoft Windows Server 2003 and 2008 VSS works with VSS-aware applications that are in use to determine when a volume shadow copy (snapshot) can be made. A snapshot is a copy of some data set, for example, a disk volume, at a point-in-time. VSS then communicates with the

operating system and applications and freezes computing tasks to allow snapshot creation. The freeze function enables an application consistent snapshot to be taken. The shadow copy volume is then used for the actual backup. After the shadow copy volume is saved on the backup device, the shadow copy is deleted.

The 3PAR VSS Hardware Provider coordinates between Requestors (backup applications – Recovery Manager), Writers (applications in windows services – Microsoft SQL Server) and providers (components that create the shadow copies).

VSS Component	VSS Component Description
Requestor	The requestor is backup software that initiates the creation of the shadow copy.
Writer	The writer is any application software that participates in the shadow copy process and whose files and data are included in the process.
Provider	The provider is the storage technology that does the actual shadow copy.

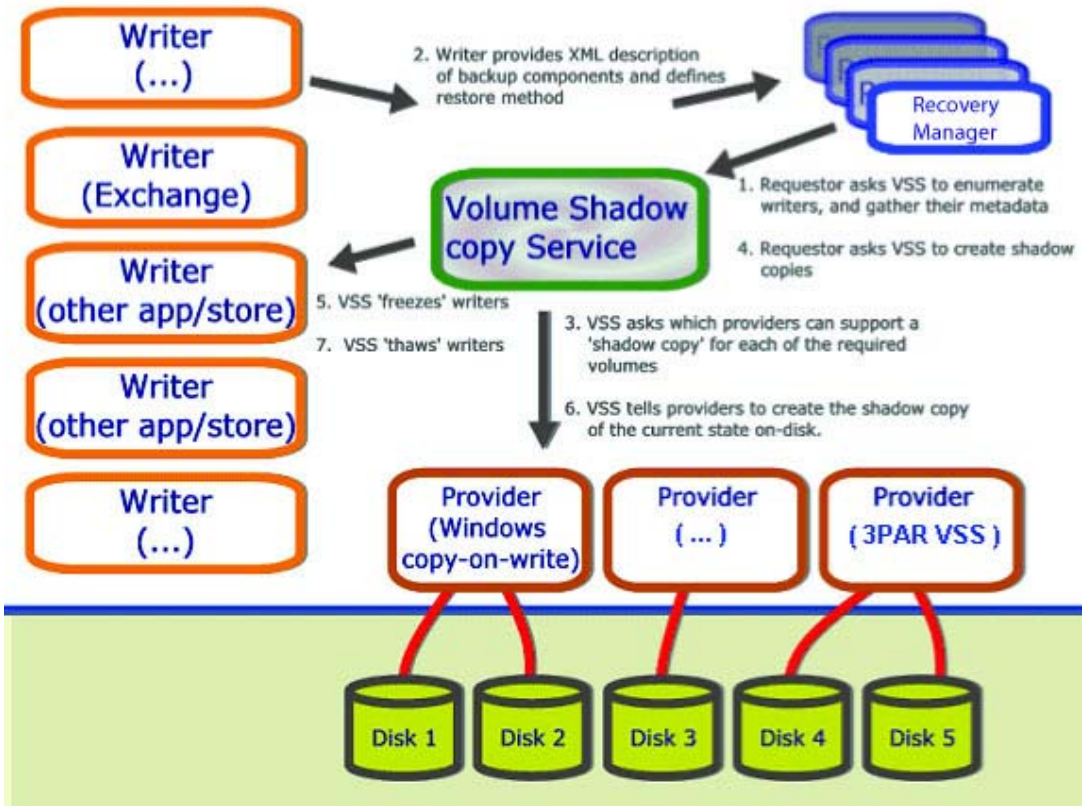


Figure 2-1. 3PAR VSS Provider for Microsoft Windows Operation Overview

2.2 How VSS Works with Microsoft SQL Server

1. The requestor (Recovery Manager) sends a command to the Volume Shadow Copy service to take a shadow copy of the selected Microsoft SQL Server database.
2. The Volume Shadow Copy service communicates with the Microsoft SQL Server with SQL Writer. The Volume Shadow Copy service flushes the current log files to disk and suspends any writes to the transaction log files and to the database file. However, read-only access is still allowed so any data that is already in the SQL Server database can still be accessed from the client.

3 Volume Shadow Copy service communicates with the provider to initiate the shadow copy process for the disk volumes that contain the Microsoft SQL Server data. VSS communicates with the writers and providers to create a shadow copy. The provider actually creates the shadow copy.

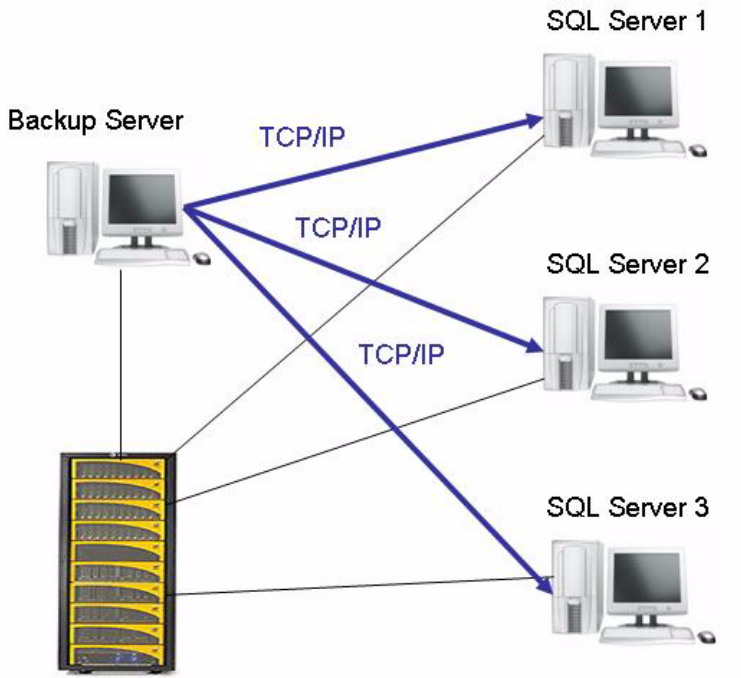
4 The shadow copy typically takes less than a minute.

5 When the shadow copy has been taken, Volume Shadow Copy service communicates with the Microsoft SQL Server with MSDE writer or SQL writer to let it know that it can resume writing to disk.

6 The queues of SQL transactions are written to the logs.

7 The shadow copy is now available to the requestor for backing up.

2.3 Recovery Manager Layout



Recovery Manager 3.0.0 for Microsoft SQL provides off-host backup and mounting capabilities. These features reduce the impact on the production server and also provides the central management for the backup and restore functions across all SQL servers.

- Backup Server

  A backup server is the central control point for the Recovery Manager which controls and manages the backup and restore on different SQL servers.

- SQL Server

  The SQL Server can be a standalone server or a cluster of virtual servers. The release version of SQL server must be 2005 or 2008. Versions prior to SQL Server 2005 are not supported.

- Connectivity

  Both the backup server and SQL Server need to be connected to the same InServ Storage Server.

- Windows Domain

  The backup server and the SQL server must be in the same Windows domain. Running a backup server and SQL server in different Windows domains is not supported.

2.4 Supported VSS Components

VSS components currently supported by 3PAR Hardware Provider are:

- Requestor
  - Recovery Manager
  - Veritas NetBackup™ 6.5 for Microsoft windows
- Writer
  - Microsoft SQL Server 2005 and 2008 with SQL Server Writer
-



For platform-specific release levels, please refer to the 3PAR InForm OS 2.2.4 Configuration Matrix (PN 320-200099) which can be accessed through the 3PAR Document Control System (DCS).

2.5 System Requirements

- Microsoft Windows Server 2003 with SP2 or above, or Windows Server 2008
- Microsoft SQL Server 2005 with SP2 or above, or Microsoft SQL Server 2008
- SQL Server client utilities need to be installed on the backup server
- All SQL instances need to be TCP/IP enabled
- Microsoft .Net Framework 2.0 or above
- Recovery Manager license for the InServ Storage Server
- Snapshot space must be available on 3PAR Virtual Volumes where the SQL server database resided

  In order to take snapshots on 3PAR's base volume, the volume itself must have snapshot space associated during the base volume creation. Please see the *3PAR InForm OS Command Line Reference* manual for creating a base volume with snapshot, or assigning snapshot space after the base volume is created.



NOTE: For platform-specific release levels, please refer to the 3PAR InForm OS 2.2.4 Configuration Matrix (PN 320-200099) which can be accessed through the 3PAR Document Control System (DCS).

2.6 Recovery Manager Requirements

- The Microsoft SQL Server database for which virtual copies will be taken must be online.
- One backup server is required.
- The Microsoft SQL server database for restore must be online. If the database is offline, the recovery manager assumes the database might not exist.
- Backup and Production server(s) must be running the same level OS version, operating mode, and version of 3PAR Recovery Manager. Cross-platform configurations are not supported.
- The SQLServerWriter service must be running.

2.7 Features Supported

- Windows Enterprise Server - For detailed OS version support, see the *3PAR InForm OS 2.2.4 Configuration Matrix* (PN 320-200099).
- Microsoft SQL Server - For detailed SQL server support, see the *3PAR InForm OS 2.2.4 Configuration Matrix* (PN 320-200099).
- Supports Volume Shadow Copy Service "Full" backup type
- Virtual copy creation using 3PAR VSS Provider for Microsoft Windows
- Access 3PAR InServ Storage Server via InForm CLI for Microsoft Windows
- Mount virtual copies locally or remotely
- Mount virtual copies Read-Write or Read-Only
- Unmount virtual copies
- Delete virtual copies
- Environment Report
- File Copy Restore
- Restore from virtual copies
- Backup using Veritas NetBackup – both client and master
- Restore using Veritas NetBackup
- Recovery Manager GUI
- Recovery Manager CLI
- Volumes Analyze
- Volume Restore
- Supports virtual copy creation of local and remote Microsoft SQL Server
- Supports MSCS environment
- Provides the Remote Copy feature for the SQL Servers
- Supports virtual copy operations on SQL instance and database level

2.8 Features Not Supported

- Microsoft SQL Server offline backup.
- VSS backup types "Incremental," "Differential," "Copy," and "Log" are not supported.
- All backup tools excluding Veritas NetBackup.
- Access to 3PAR InServ Storage Server via SSH.
- Management of different versions of Microsoft SQL server from the same Recovery Manager backup server.
- Interoperability between Windows Server 2003 and Windows Server 2008 is not supported. For example, a virtual copy created under Win2003 cannot be mounted to a Win2008 target host. A virtual copy created under Win2008 cannot be mounted to a Win2003 target host.
- Taking snapshot of the SQL servers created with the **SQL Server Authentication** setting is not supported. 3PAR only supports SQL servers created with the **Windows Authentication** setting.
- Taking a snapshot of a database residing on a network mapping volume is not supported.
- Virtual copy ownership transferring is not supported. For example, virtual copies can only be removed by the same user who created them. Virtual copies can only be unmounted by the same user who mounted them.

3 Installation and Deinstallation

In this chapter

3.1 Installing Recovery Manager **3.1**

3.2 Verifying the Installation **3.6**

3.3 Deinstalling Recovery Manager **3.6**

3.4 Understanding the Registry **3.7**

3.5 Migrating from Version 2.2.3 **3.8**

This chapter describes how to install, verify, and deinstall Recovery Manager.

3.1 Installing Recovery Manager

Before installing 3PAR Recovery Manager, install the InForm OS Command Line Interface (CLI) for remote clients as well as 3PAR VSS Provider 1.5.1 for Microsoft Windows. For details on installing the CLI, see the *3PAR InForm OS CLI Administrator's Manual*. For details on installing the 3PAR VSS provider, see the *3PAR VSS Provider 1.5.1 User Guide*.

Install 3PAR Recovery Manager on a host computer where the Microsoft SQL Server is running.

NOTE: If you are installing the application through a remote desktop session, the newly added environment variables will not become effective until you log off the session. It is recommended that you install the application directly from system console.

NOTE: If you are installing Recovery Manager for SQL on a system that also has a NetBackup service running in the background, the NetBackup service needs to be restarted after the installation of Recovery Manager for SQL is complete. This is required for the environment path to be registered in the NetBackup service.

To install Recovery Manager:

1. Log on as the system administrator.
2. Insert the Recovery Manager CD into a CD-ROM drive.
3. Open a Windows Explorer window:

   Click **Start** > **All Programs** > **Accessories** > **Windows Explorer**.
4. Click the CD-ROM drive to expand the files.
5. Select the corresponding package for your OS version.

Table 3-1. Operating System Packages

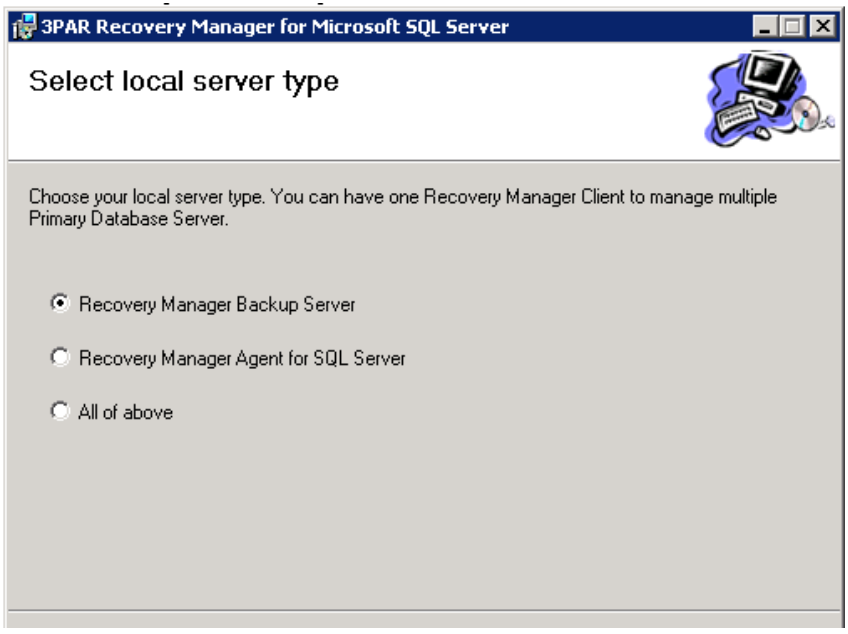
Operating System	Package
Win2003 32-bit	RMS_W2K3_x86.msi
Win2003 64-bit	RMS_W2K3_x64.msi
Win2008 32-bit	RMS_W2K8_x86.msi
Win2008 64-bit	RMS_W2K8_x64.msi


3PAR Recovery Manager for Microsoft SQL Server
Welcome to the 3PAR Recovery Manager for Microsoft SQL Server Setup Wizard
The installer will guide you through the steps required to install 3PAR Recovery Manager for Microsoft SQL Server on your computer.
WARNING: This computer program is protected by copyright law and international treaties. Unauthorized duplication or distribution of this program, or any portion of it, may result in severe civil or criminal penalties, and will be prosecuted to the maximum extent possible under the law.
Cancel
< Back
Next >

6 Click **Next**.



7 Agree to the license agreement, then click **Next**.

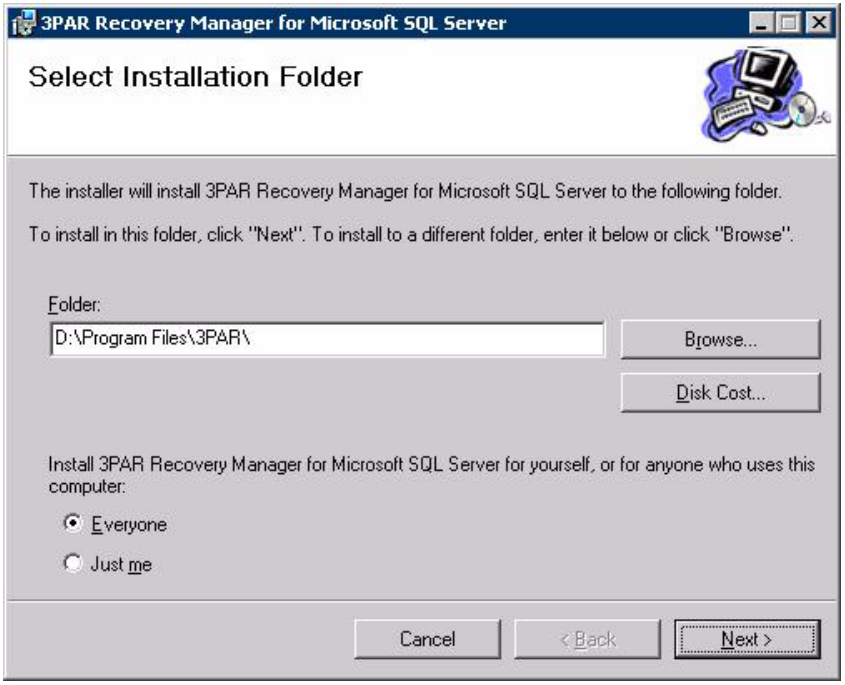


8 Specify the installation type:

- **Recovery Manager Backup Server** is the setup of backup server where Recovery Manager is operated from.

NOTE: The SQL Server client utilities need to be installed on the host where Recovery Manager Backup Server is installed.

- **Recovery Manager Agent for SQL Server** is installed on the production server running Microsoft SQL Server.
- **All of above** is for a single machine setup which means the backup server is also the production server.



9 Specify the installation directory for Recovery Manager



NOTE: The **Disk Cost** button analyzes the amount of available storage space and compares it to the amount required for the Recovery Manager installation. Default options to install for **Everyone** will create the necessary program folder and desktop icon for everyone who has access to the system.

10 Click **Next** to confirm the installation destination.

11 Click **Close** once the installation has completed.

3.2 Verifying the Installation

Verify the Recovery Manager installation as follows:

1 Log on as the system administrator.

2 Click **Start > Control Panel > Add/Remove Programs** on Windows 2003 or programs and features on Windows 2008.

Result: The **Add/Remove Programs** window appears on Windows 2003 or programs and features on Windows 2008.

3 Verify Recovery Manager appears under the **Currently installed programs** column.

3.3 Deinstalling Recovery Manager

To deinstall 3PAR Recovery Manager:

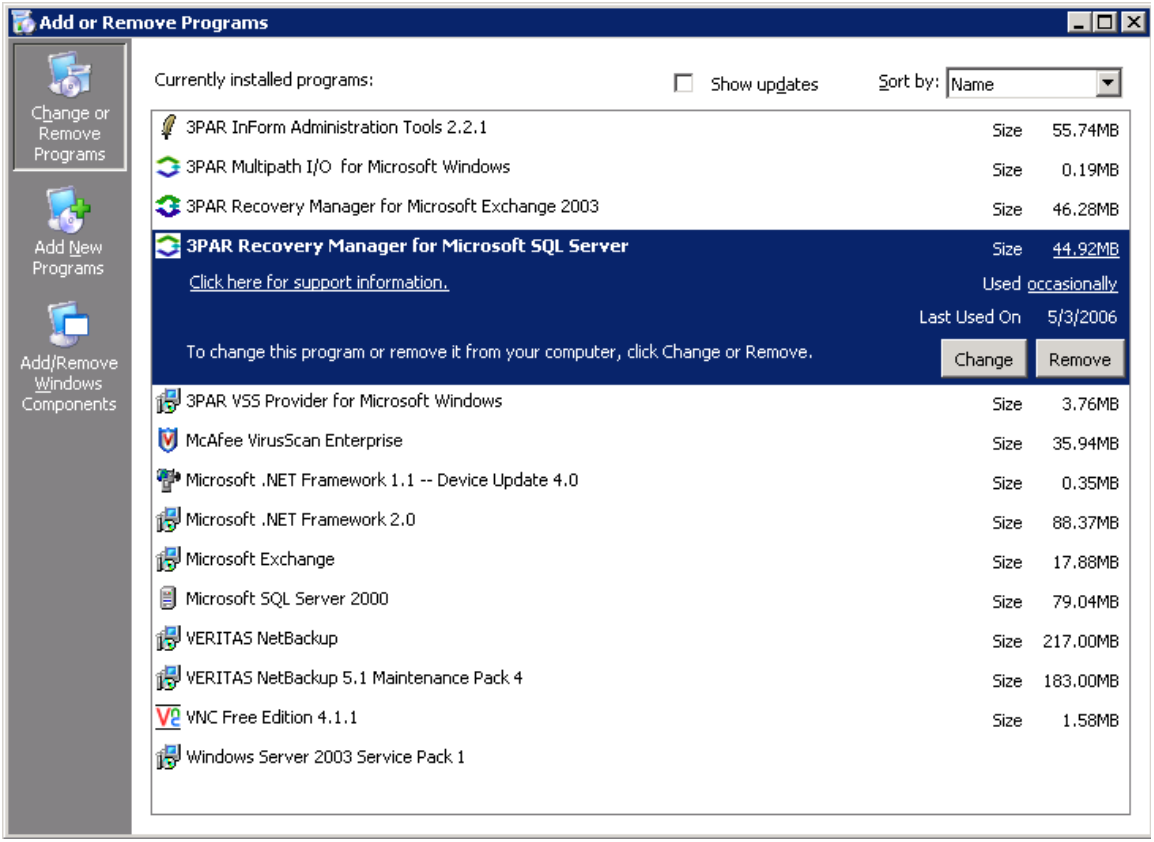
1 Log on as the system administrator.

2 Click **Start > Control Panel > Add/Remove Programs** on Windows 2003 or programs and features on Windows 2008.

Result: The **Add or Remove Programs** window appears on Windows 2003 or programs and features on Windows 2008.

3 Select **3PAR Recovery Manager for Microsoft SQL Server**.

4 Click **Remove**.



3.4 Understanding the Registry

After Recovery Manager is installed successfully, some information is written into the registry database under: `HKEY_LOCAL_MACHINE\SOFTWARE\3PAR\VCDBA_SQL`

CAUTION: DO NOT try to edit the registry key value unless asked by your local service provider for support purposes.

Table 3-2. System Registry Values for Recovery Manager

Name	Value
Data-File-Path	Location of the XML repository.
Install-Path	Recovery Manager for SQL Server location path.
Log-File-Path	Tracing log files.
TraceHistory	Number of Trace files to keep before deleting.
TraceLevel	The trace level for the log file.
Version	The current version of Recovery Manager for the SQL Server.
BackupTimeout	Backup process timeout value in minutes to confirm that NetBackup has received the backup request.
PromoteTimeout	Time interval in seconds to monitor the status of volume restore operation.
Protocol	Sub-registry key that contains value for communication port and protocol type used by Recovery Manager for SQL



NOTE: The Data-File-Path is created in the registry database only after you run the Recovery Manager GUI the first time.

3.5 Migrating from Version 2.2.3

Recovery Manager 3.0.0 for SQL Server does not support a direct upgrade from an earlier version of Recovery Manager for SQL Server. Any existing Recovery Manager for SQL Server installations need to be deinstalled prior to installing Recovery Manager 3.0.0 for SQL Server. Data migration for the virtual copy repository is required in order to upgrade from Recovery Manager 2.2.3 for SQL Server to Recovery Manager 3.0.0 for SQL Server. The virtual copy

repository will remain even if the existing Recovery Manager for SQL Server installation has been deinstalled.



NOTE: Data migration is required to upgrade from Recovery Manager 2.2.3 for SQL Server to Recovery Manager 3.0.0 for SQL Server. Data migration is not required to upgrade from Recovery Manager 2.2.4 for SQL Server to Recovery Manager 3.0.0 for SQL Server.

To migrate the data:

Deinstall Recovery Manager 2.2.3 for SQL:

1. Unmount all the virtual copies
2. Uninstall the Recovery Manager 2.2.3 for SQL.

Install Recovery Manger 3.0.0 for SQL and perform the migration:

1. Install Recovery Manager 3.0.0 for SQL to the original location where the 2.2.3 version was located.
2. Run `RMSql migrate -s SQLserver` to migrate from 2.2.3 to 3.0.0.

The `RMSql migrate` command will migrate the SQL server virtual copies one by one. Migration of all the SQL servers at one time is not currently supported.



NOTE: Migration from version 2.2.2 is not supported using the `RMSql migrate` command. Instead, you must remove all of the 2.2.2 virtual copies and uninstall version 2.2.2 before installing version 3.0.0.

4 Configuring NetBackup for Recovery Manager

In this chapter

4.1 Overview 4.1

4.2 Defining a NetBackup Policy 4.2

This chapter contains instructions for configuring Veritas NetBackup for use with Recovery Manager.

4.1 Overview

Recovery Manager can be used with Veritas NetBackup for backup and restore of virtual copies created by Recovery Manager. The following sections describe setting up Veritas NetBackup for use with Recovery Manager.



NOTE: There is a known limitation from NetBackup regarding case sensitivity. When mounting a virtual copy for backup with NetBackup, the `–m` parameter is case-sensitive and must match the `–m` option in the Windows OS environment. For example, if the mount point you would like mount to appears as `C:\NoticeMyCAPS` in Windows you must use the same case in your NetBackup `–m` parameter.

4.2 Defining a NetBackup Policy

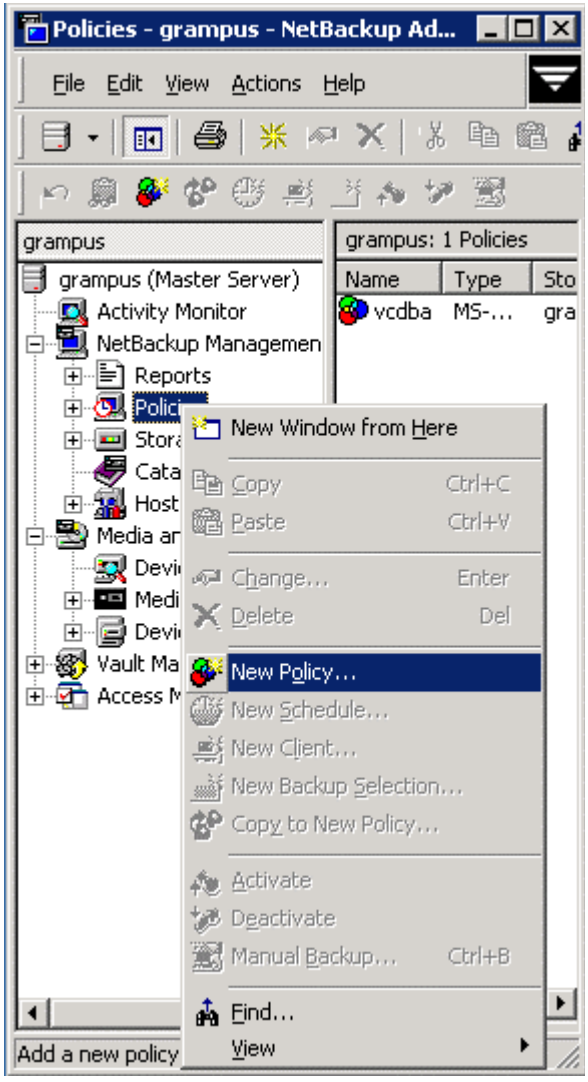
Perform the following steps to set up a NetBackup policy for use with Recovery Manager.



NOTE: A storage unit to store the virtual copy backups should already be set up. See the Veritas NetBackup documentation for setting up a storage unit.

1 Launch the Veritas **NetBackup Administration Console** from the NetBackup Master Server.

2 Select the policies node in the left pane, right-click and select **New Policy...**



3 Give the policy a name.

NOTE: Provide a policy name that easily identifies it as a policy for use with Recovery Manager.

4 Click **OK**.

Result: A dialog box appears.

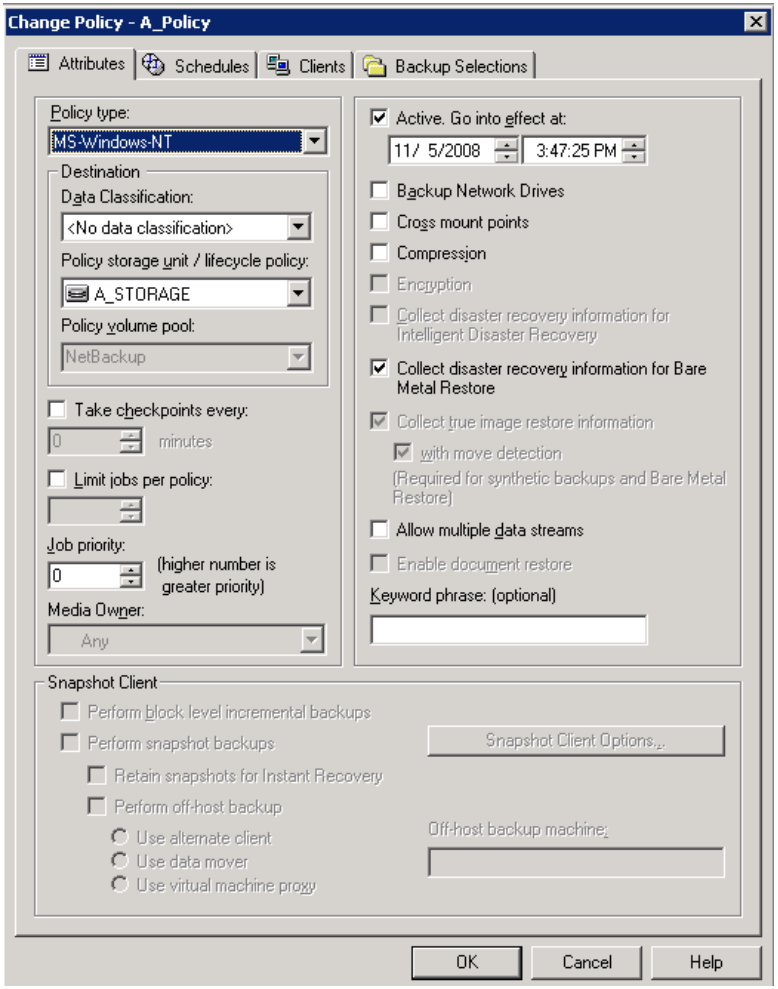
5 On the **Attributes** tab:

a Select **MS-Windows-NT** as the policy type.

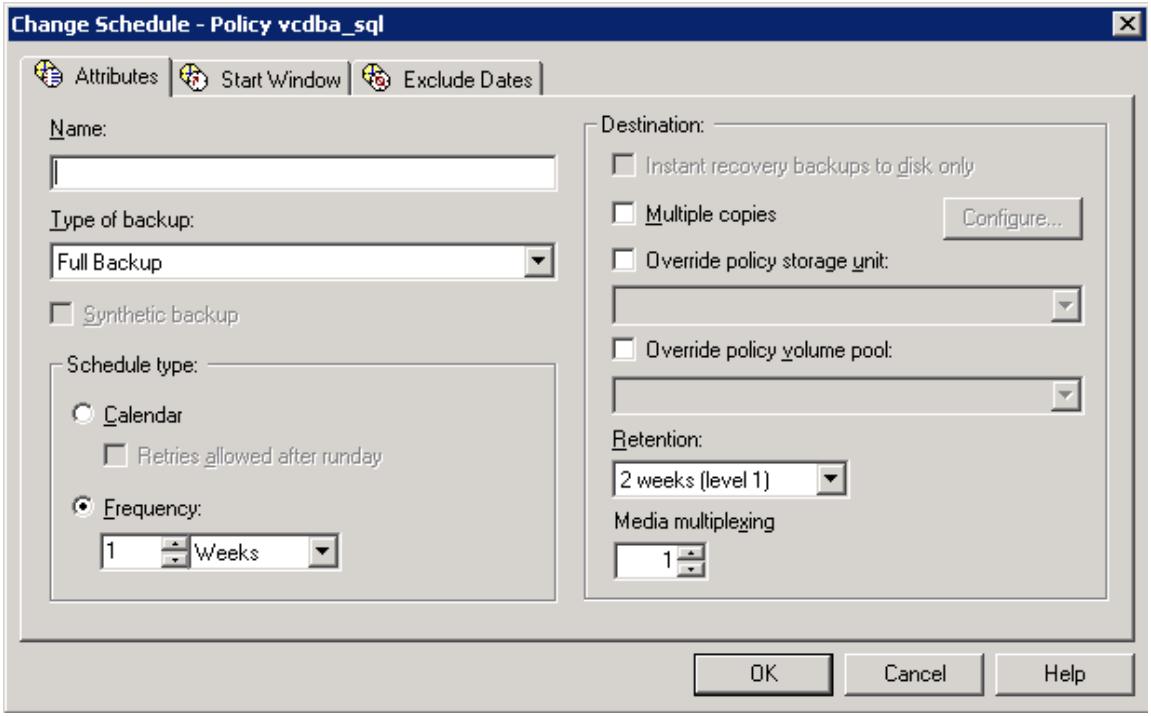
b Select the storage unit for where the virtual copy backups will reside.

c Uncheck **Collect disaster recovery information for Bare Metal Restore** if it is checked by default.

d Select the **Schedules** tab and click **New...**



Result: The Change Schedule Attributes dialog appears.

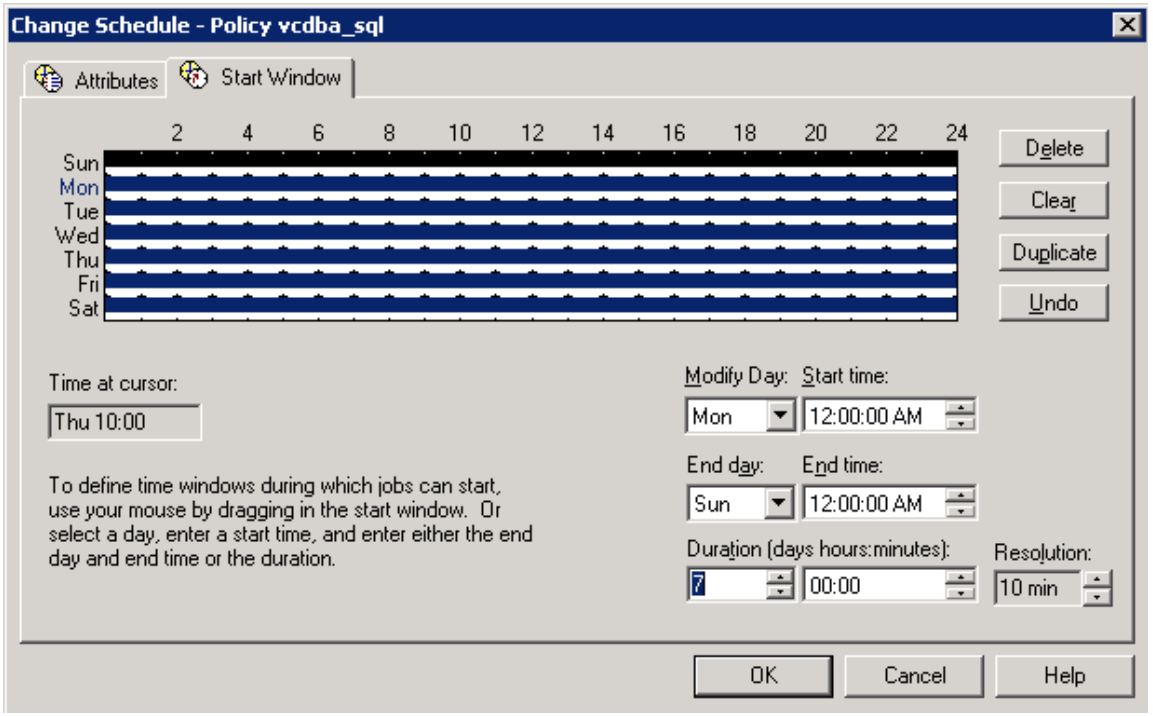


e When the backup server is just running the NBU client interface, select **User Backup** from the drop-down menu as the type of backup.
When the backup server is just running the NBU master interface, select **Full Backup** from the drop-down menu as the type of backup.

f Click **OK**.

g Select the "Start Window" tab and activate the all time slot for schedule by selecting "Mon," "12:00:00AM" as the "Start time" and "Sun," "12:00:00AM" as the "End data/

time" and make sure "7" days are entered in the Duration selection box. Click **OK** to activate the schedule:



h Select the **Clients** tab and enter the machine name where the Recovery Manager is installed.

NOTE: Recovery Manager supports full backups when running with a NetBackup Master server. Recovery Manager also supports the user defined backup type when running with a NetBackup client machine.

i Click **OK**.

Result: Veritas NetBackup is ready for use by Recovery Manager.

NOTE: When the backup server is just running the NBU client interface, the policy validation cannot be done.



NOTE: 3PAR recommends that you reboot the system after the NetBackup Installation so that the NBU path gets added to the PATH env variable. If you do not reboot the system, the backup to the media might end up with Windows API call errors. 3PAR recommends that you check the PATH env variable for the NBU binary path entry.



NOTE: If NetBackup path is added to the system environment PATH variable after Recovery Manager for SQL installation, you need to restart Recovery Manager for SQL service for the changes to take effect.

5 Configuring Recovery Manager

In this chapter

5.1 Configuring the Database **5.1**

5.2 Configuring Recovery Manager **5.2**

5.3 Change Port Procedure **5.2**

5.4 Change Protocol Procedure **5.3**

5.5 Configuring the Host Computer and the 3PAR InServ Storage Server **5.4**

This chapter contains instructions for configuring Microsoft SQL Server and Recovery Manager.

5.1 Configuring the Database

The system supports Microsoft SQL Server database files and transaction logs created on basic disks using the NTFS file system.



NOTE: 3PAR recommends that each database, as well as transaction log files, be allocated to separate volumes. This configuration provides the greatest flexibility when performing individual database backup and recovery, thus reducing the exposure to conflicts between databases. The snap volume restore would corrupt the other databases if there are two databases using the same volume.

5.2 Configuring Recovery Manager

Use the following to assist in the configuration of the Recovery Manager.

5.2.1 Installation Location

By default Recovery Manager is installed in
`C:\Program Files\3PAR\RM\SQL`

5.2.2 Search Path

The Recovery Manager executables are stored in the program folders directory chosen during installation. To use the command line components, the Recovery Manager directory must be added to the Windows search path. During installation, the search path is automatically added to the environment variable PATH.

5.3 Change Port Procedure

Use the `RMSql portconfig` command shows the TCP port of Recovery Manager service is currently using.

`RMSql portconfig` - shows the port the Recovery Manager service is currently using.

`RMSql portconfig -s ServerA` - shows the port the Recovery Manager service is currently using on `ServerA`.

The TCP port used by Recovery Manager agent service must be the same across all servers. Therefore, the same TCP port should apply to all servers where Recovery Manager agent service is installed.

1. Shut down the Recovery Manager for SQL services on the production servers and backup servers on which you want to change the ports.
2. Use `RMSql portconfig -n 8888` to change to local server (backup server) port first.
3. Use `RMSql portconfig -n 8888 -s ServerA` to change `ServerA` port.
4. Repeat these steps if you have more servers you want to change.
5. Start the Recovery Manager for SQL service from the backup server.

6 Start the Recovery Manager for SQL service of ServerA.

7 Repeat step 6 if you have more servers.



NOTE: To have remote access to the other servers, the remote register service must be started on the other servers. In addition, special privileges must be obtained to access the remote server.

5.4 Change Protocol Procedure

Use the RMSql protocol command to show or change the current protocol used for communications between the backup and production servers. The default protocol is TCP, you can also use HTTP.

RMSql protocol - shows the protocol the Recovery Manager service is currently using.

RMSql protocol -s ServerA - shows the protocol the Recovery Manager service is currently using on ServerA.

The protocol should be changed at the network level which means all servers with the Recovery Manager agent installed need to be changed. This command is run from Recovery Manager backup server.

1 Shut down the Recovery Manager for SQL services on the production servers and backup servers on which you want to change the protocol.

2 Use RMSql protocol -pt http to change to local server (backup server) protocol first.

3 Use RMSql protocol -pt http -s ServerA to change ServerA protocol to http.

4 Repeat these steps if you have more servers you want to change.

5 Start the Recovery Manger for SQL service from the backup server.

6 Start the service of serverA.

7 Repeat step 6 if you have more servers.

5.5 Configuring the Host Computer and the 3PAR InServ Storage Server

See the *3PAR InForm OS Concepts Guide* and the *3PAR InForm OS CLI Administrator's Manual* for information about configuring the controller nodes on the InServ storage server to communicate with the host computer.

6 Using the Recovery Manager GUI

In this chapter

6.1 Overview of the Recovery Manager GUI 6.2

6.2 Starting the GUI 6.5

6.3 Stopping the GUI 6.5

6.4 Server Registration 6.5

6.5 Setting the Recovery Manager Policy Configuration 6.9

6.6 Creating Instance or Database Virtual Copy 6.10

6.7 Creating a Remote Virtual Copy 6.14

6.8 Mounting a Virtual Copy 6.17

6.9 Unmounting a Virtual Copy 6.19

6.10 Backing Up a Virtual Copy Using Veritas NetBackup 6.19

6.11 Creating a Backup at the Timestamp Level 6.24

6.12 Deleting Backup from the Repository 6.25

6.13 Restoring an Instance or Database from Virtual Copy using Volume Restore 6.25

6.14 Restoring an Instance or Database from Virtual Copy Using File Copy 6.27

6.15 Restoring a Virtual Copy Using Veritas NetBackup 6.28

6.16 Removing a Virtual Copy 6.30

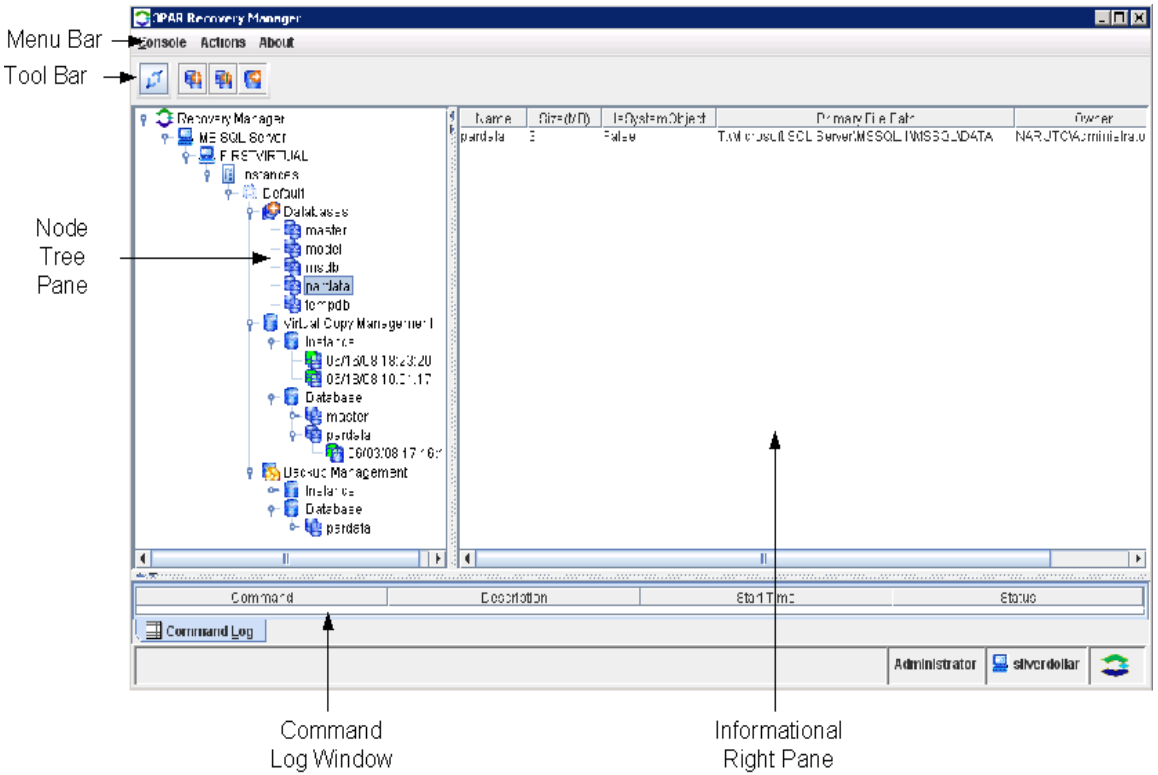
This chapter provides instructions for using the Recovery Manager Graphical User Interface (GUI) to back up and restore an Instance or databases.

6.1 Overview of the Recovery Manager GUI

Recovery Manager provides a Graphical User Interface (GUI) to backup and restore databases at a point-in-time.

Recovery Manager provides the capability to create, remove, mount, unmount, backup and restore the Virtual Copy for the Instance (all databases within the same server) and a database.

The Recovery Manager GUI is a windows application, containing a menu bar, a node tree pane, an informational right pane, and a command log window.



6.1.1 Menu Bar

The Menu Bar consists of the following menu items:

- **Console** menu - allows you to exit the GUI.
- **Actions** menu - allows you to access virtual copy functionality.
- **Help** menu - provides an **About Virtual Copy** screen.

6.1.2 Node Tree Pane

The node tree presents the Microsoft SQL Server objects. Recovery Manager supports Microsoft SQL Server on the local host as well as Servers within the same intranet. When the Recovery Manager starts, it retrieves the instances, databases, virtual copies and backup information, and displays them graphically as a nested tree in the left pane.

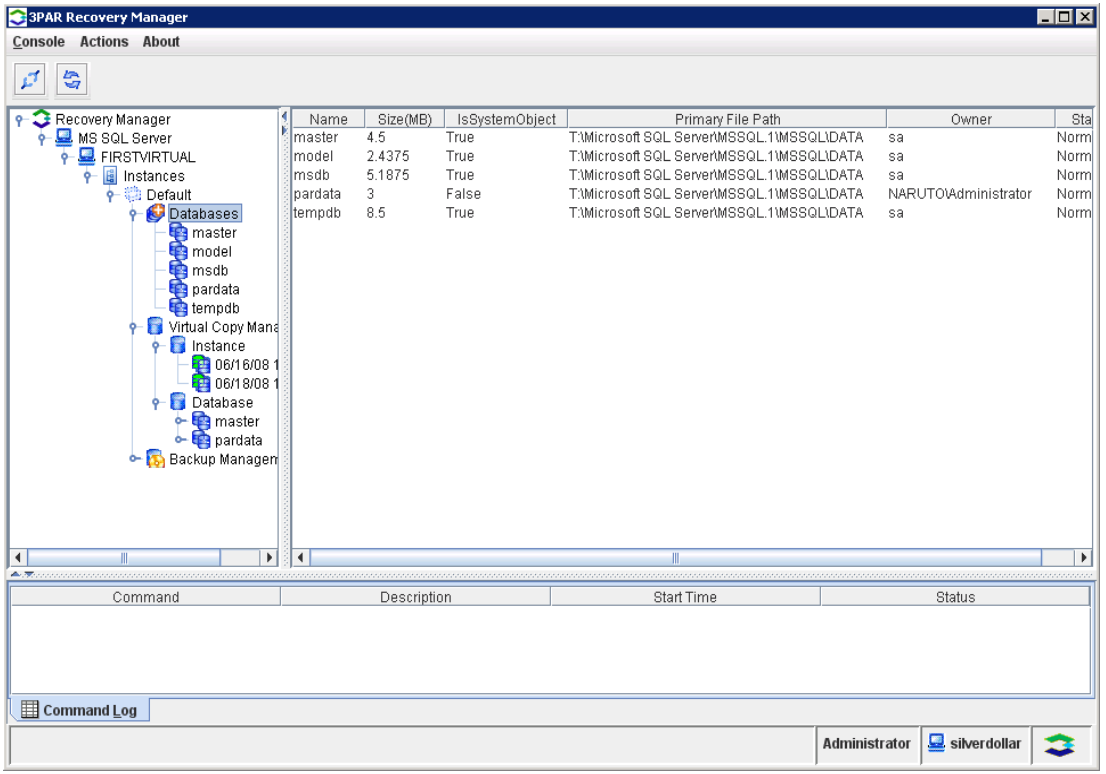
6.1.3 Informational Right Pane

The right pane displays detailed information about the children of the selected node in the left tree pane. The information displayed in the right pane is presented as a table. Scroll bars are provided to shift the table left, right, up and down.

6.1.4 Displaying Database Information

To display information about the databases under a particular instance, click on **Instances** > **Default** > **Databases** in the tree. Detailed information about the databases associated with the instance are displayed in the right pane.

The following screen displays all databases for the selected instance **Default Instance**.



The columns are explained in the following table.

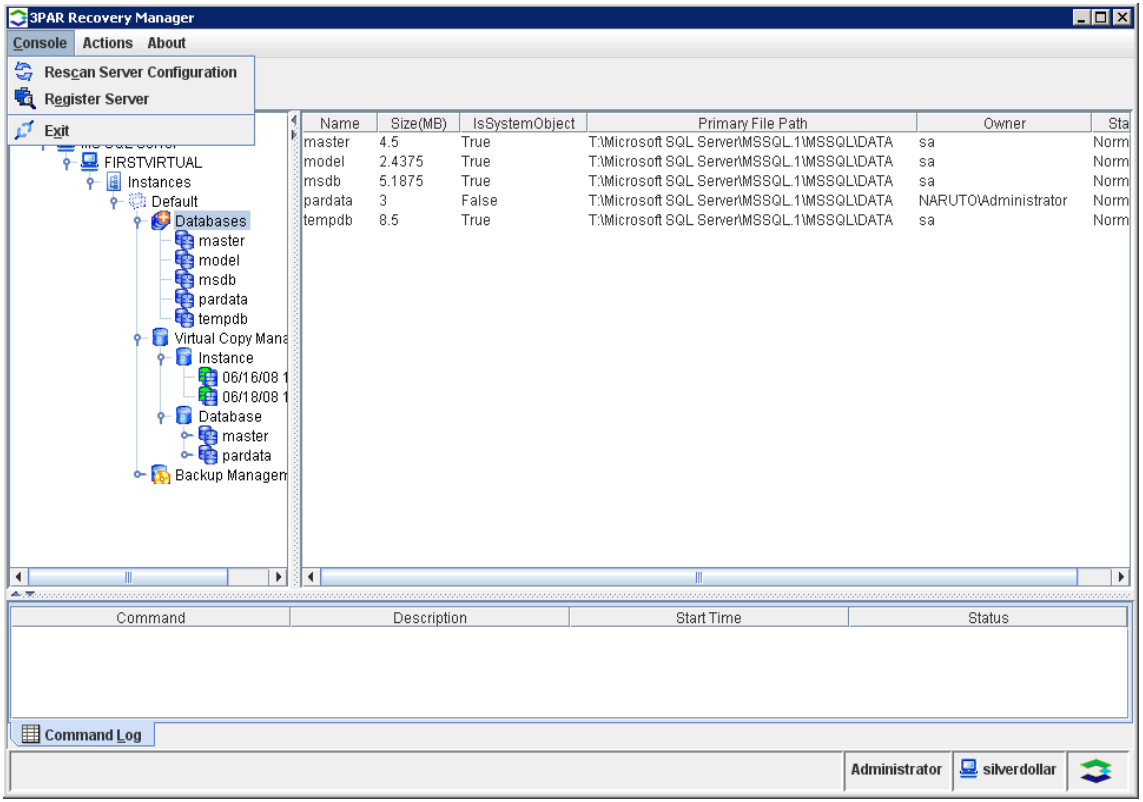
Column...	Provides...
Name	The name of the database.
Size	The size of the database.
IsSystemObject	Is the database a system object?
Primary File Path	The location of the database.
Owner	The owner of the database.
Status	The status of the database.

6.2 Starting the GUI

To start the Recovery Manager GUI, locate and double-click the **Recovery Manager for SQL** icon on your desktop.

6.3 Stopping the GUI

To stop the Recovery Manager GUI, on the Recovery Manager menu bar, click **Console** > **Exit** as shown in the following screen. For additional information, see *Menu Bar* on page 6.3

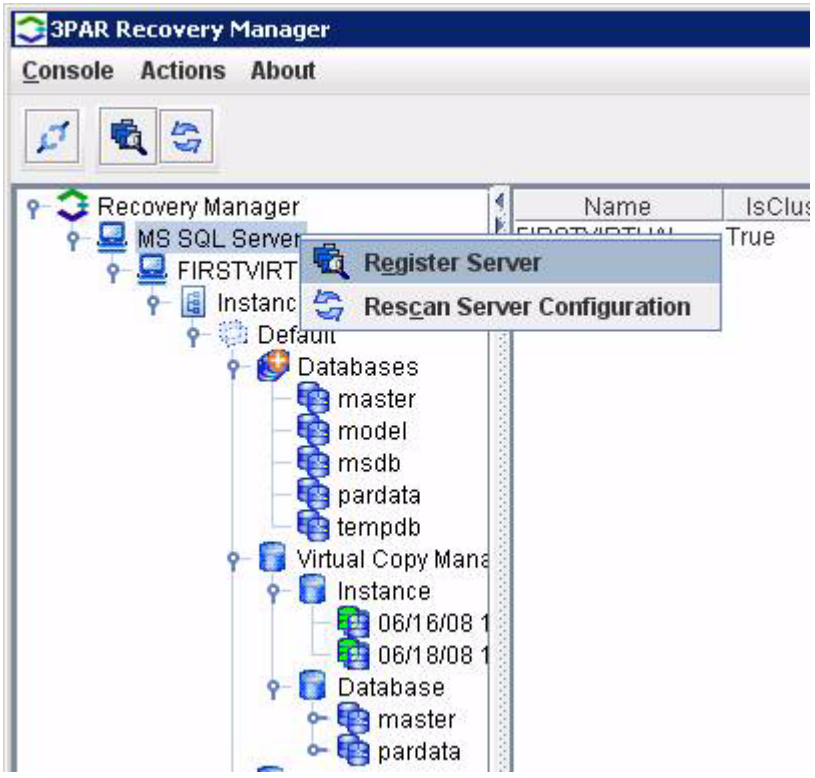


6.4 Server Registration

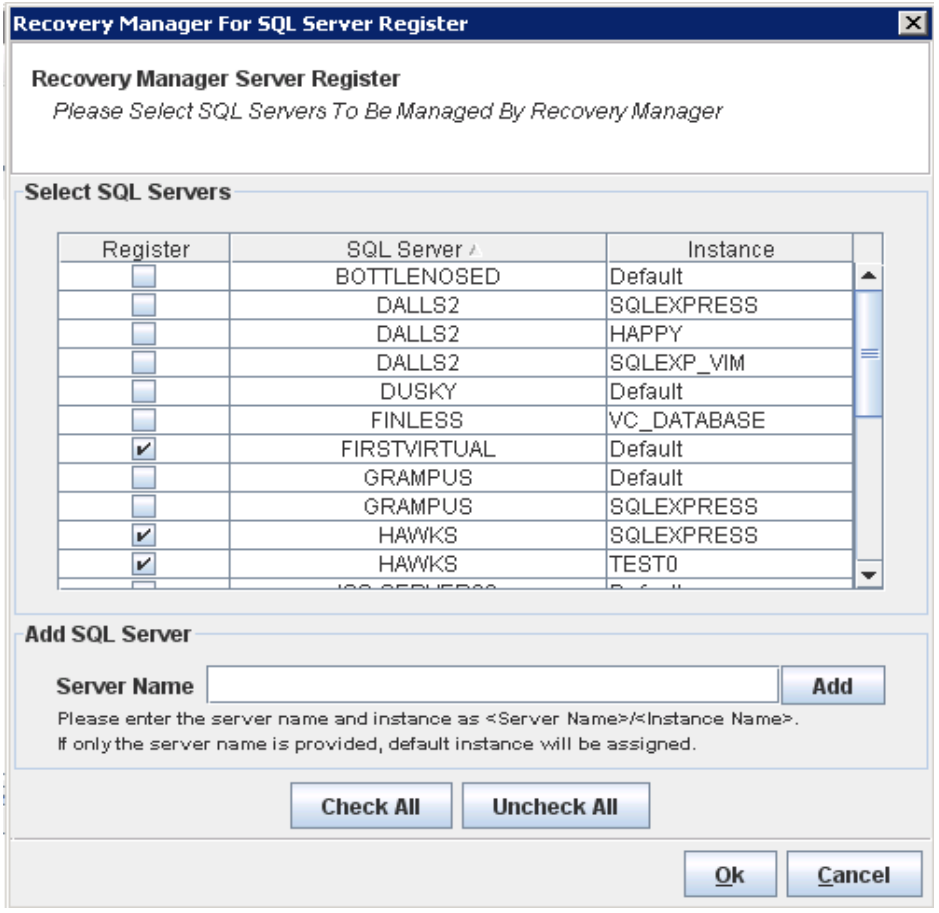
Before the GUI can manage MS SQL Servers, all servers and their corresponding instances must be registered.

To register a server instance:

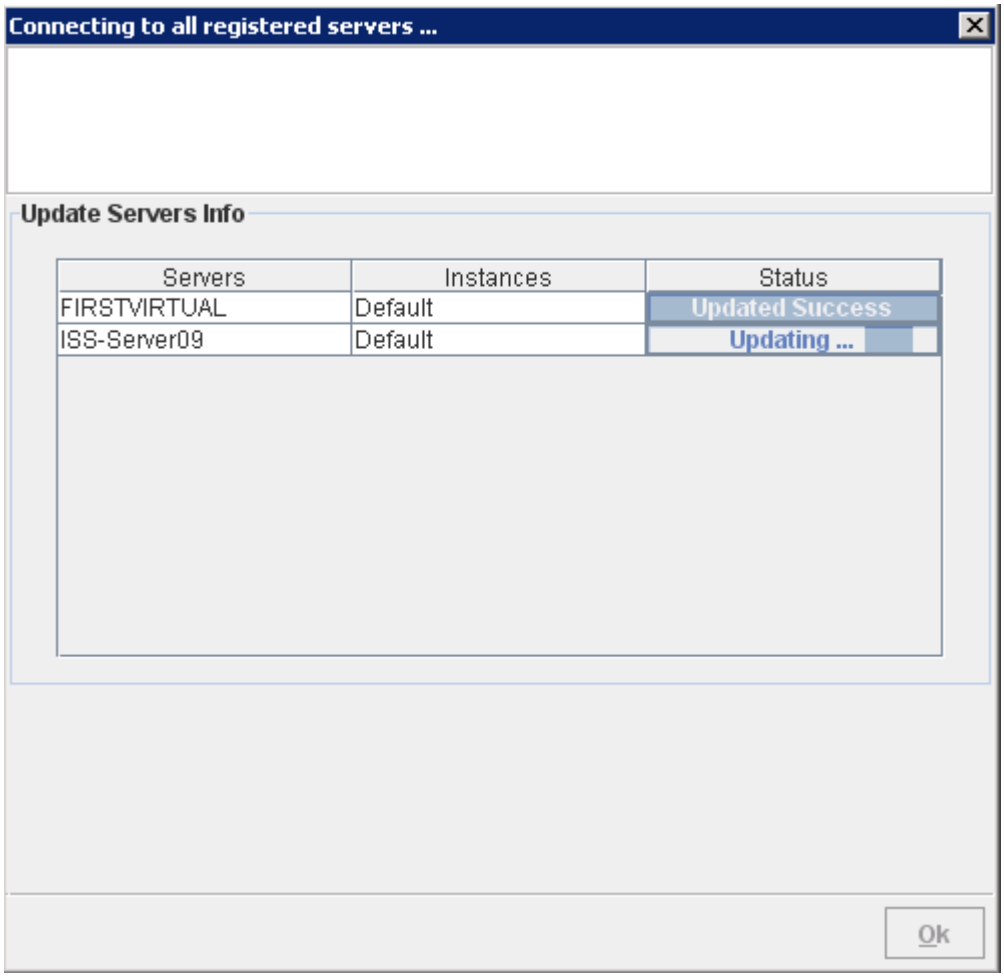
1 Right click on **MS SQL Server** node to bring up the popup menu and select **Register Server**.



2 All SQL Servers within the same active directory should be shown, and all SQL service services must be up and running. Toggle the checkbox in the Register column to add or remove server instances to the Recovery Manager GUI for SQL. Click on the **OK** button to commit the change.



To register or un-register all server instances, simply click on the **Check All** or **Uncheck All** button. The server can also be manually added by entering the *Server Name/Instance Name* and then clicking on the **Add** button. If only the *Server Name* is provided, the default instance is assigned.



NOTE: Only the server node name or virtual server name is allowed. Only the virtual server name will be shown.

NOTE: Depending on the environment, it will take several minutes for the GUI to discover and register the server instances on the network.



NOTE: When a new SQL instance or a new SQL database is created on the system after the server has been registered, server configuration has to be rescanned before it can be managed by Recovery Manager. To rescan, right click on **MS SQL Server** node to bring up the popup menu and select **Rescan Server Configuration**.

6.5 Setting the Recovery Manager Policy Configuration

By default, Recovery Manager creates a maximum of 10 virtual copies for instance and per database. When the 11th Virtual Copy creation is attempted, the default behavior is to replace the oldest Virtual Copy with a newly created Virtual Copy.

Recovery Manager provides a policy setting feature that allows you to set the maximum number of virtual copies permitted for the Instance or per database of the selected Microsoft SQL Server. The policy setting is applied at the instance level allowing each instance to have its own policy configurations.

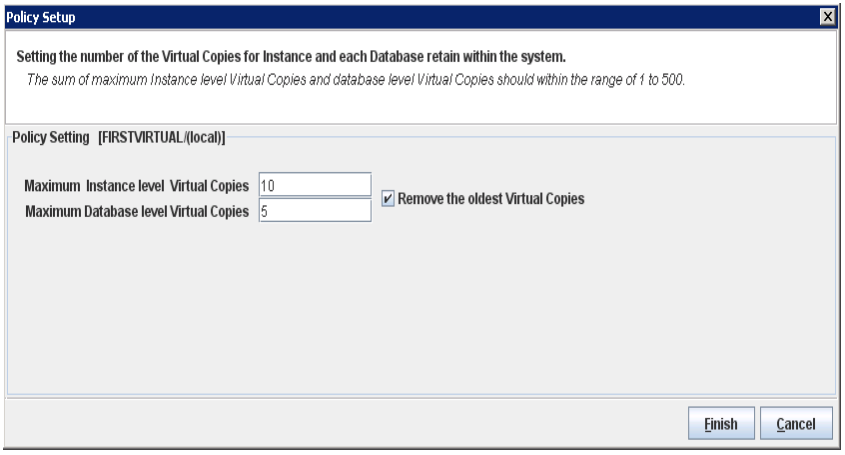
When setting the virtual copy limit, you can also optionally select the **Remove the oldest virtual copies** checkbox indicating that the oldest virtual copy for each database is automatically removed once the set maximum limit of virtual copies is reached.

If the **Remove the oldest virtual copies** checkbox is unselected, once the set maximum limit is met, a new virtual copy cannot be created until you manually delete any old virtual copy.

To set Recovery Manager policies:

1. Start the Recovery Manager application (see *Starting the GUI* on page 6.5).
2. Select the Instance that is to be configured in the node tree pane.
3. Click on the policy icon that appears on the tool bar. Alternatively, you can right-click on the SQL server and select policy, or select **Policy** from the **Virtual Copy** drop-down menu from the menu bar.

Result: The **Policy Setup** dialog box appears.



4 Enter a number (between 1 and 500) in the text fields for the Maximum Instance level Virtual Copies and Maximum Database level Virtual Copies.

5 If you want to remove the oldest virtual copy, select **Remove the oldest virtual copies** option. If you want to keep the oldest virtual copy, uncheck the **Remove the oldest virtual copies** option.

6 Click **OK**.



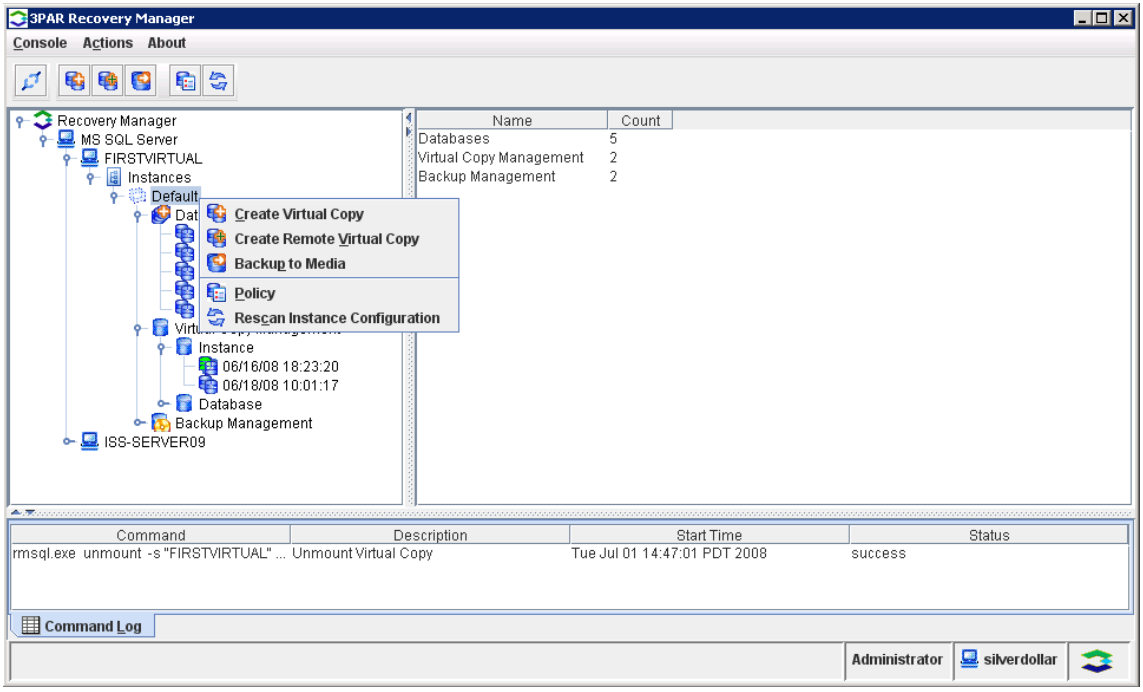
NOTE: The virtual copy policy is effective immediately.

6.6 Creating Instance or Database Virtual Copy

Virtual Copies are created at the instance or database level for Microsoft SQL Server. When Recovery Manager creates a Virtual Copy for an instance, it creates a Virtual Copy of all databases under that instance including the instance's log and system files. Similarly, when Recovery Manager creates a Virtual Copy for a database, it creates a Virtual Copy of all virtual volumes where the database, log, and system files reside to preserve the data for consistency.

To create a virtual copy of an instance:

1. Navigate through the node tree view on the left pane and locate the instance for which a virtual copy is to be created.
2. Click the instance name and the **Create Virtual Copy** icon appears on the toolbar. You can select to create a virtual copy by:

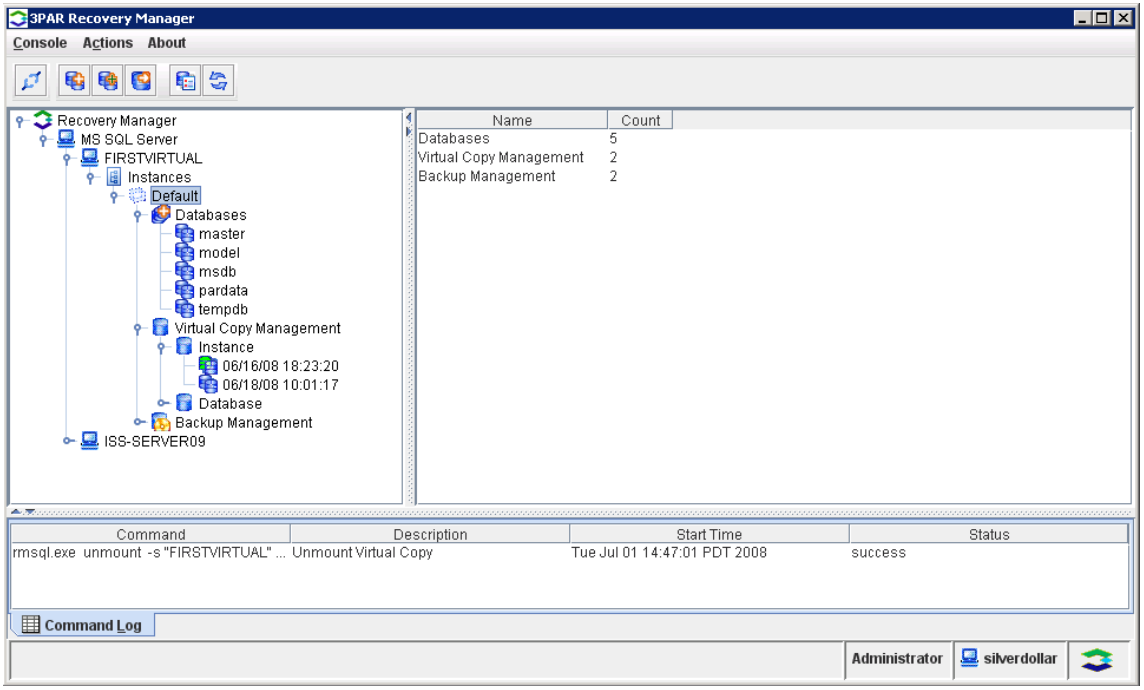


   a. Click on the **Create Virtual Copy** icon on the toolbar.

   b. Right-click on the database node and selecting **Create Virtual Copy** from the pop-up menu.

   c. Select **Create Virtual Copy** from the **Actions** drop-down menu on the menu bar.

Once a virtual copy is created, it appears under the **Virtual Copy Management** node under the relevant instance name (see the following figure). The virtual copy is given a timestamp that is used as its name.

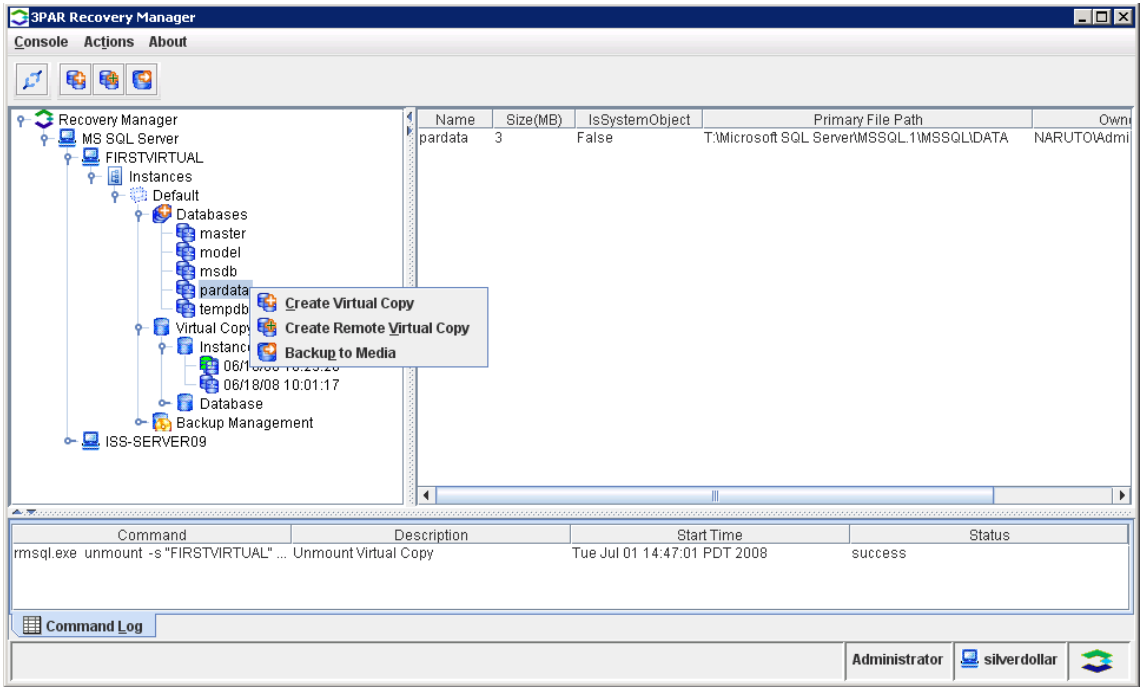


Result: A newly created timestamp will be added to Virtual Copy Management->instance tree node in the Node Tree Pane.

NOTE: Creating a Virtual Copy of an Instance creates a Virtual Copy of all the volumes where all the databases reside. Therefore, it is recommended that only database data files be stored in the volume.

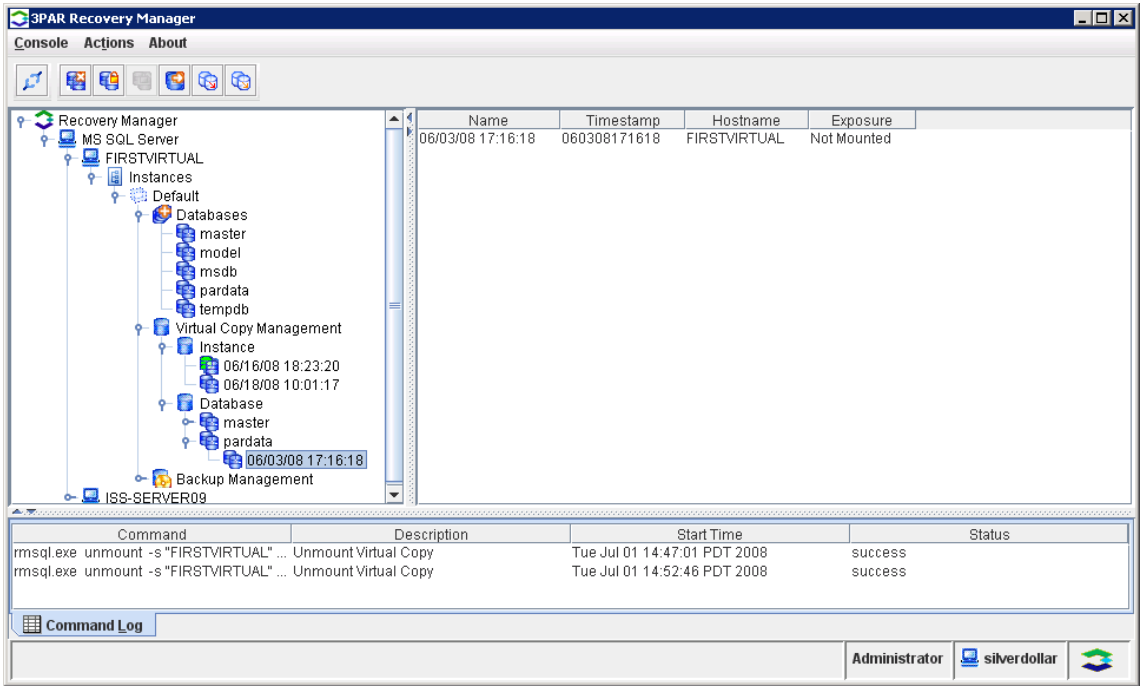
To create a virtual copy of a database:

1. Navigate through the node tree view on the left pane and locate the database for which a virtual copy is to be created.
2. Click the database name and the **Create Virtual Copy** icon appears on the toolbar. You can select to create a virtual copy by:



   a. Click on the **Create Virtual Copy** icon on the toolbar.
   b. Right-click on the database node and selecting **Create Virtual Copy** from the pop-up menu.
   c. Select **Create Virtual Copy** from the **Actions** drop-down menu on the menu bar.

Once a virtual copy is created, it appears under the **Virtual Copy Management** node under the relevant database name (see the following figure). The virtual copy is given a timestamp that is used as its name.



Result: A newly created timestamp will be added to Virtual Copy Management->instance tree node in the Node Tree Pane.



NOTE: Creating a Virtual Copy of an Instance creates a Virtual Copy of all the volumes where all the databases reside. Therefore, it is recommended that only database data files be stored in the volume.

6.7 Creating a Remote Virtual Copy

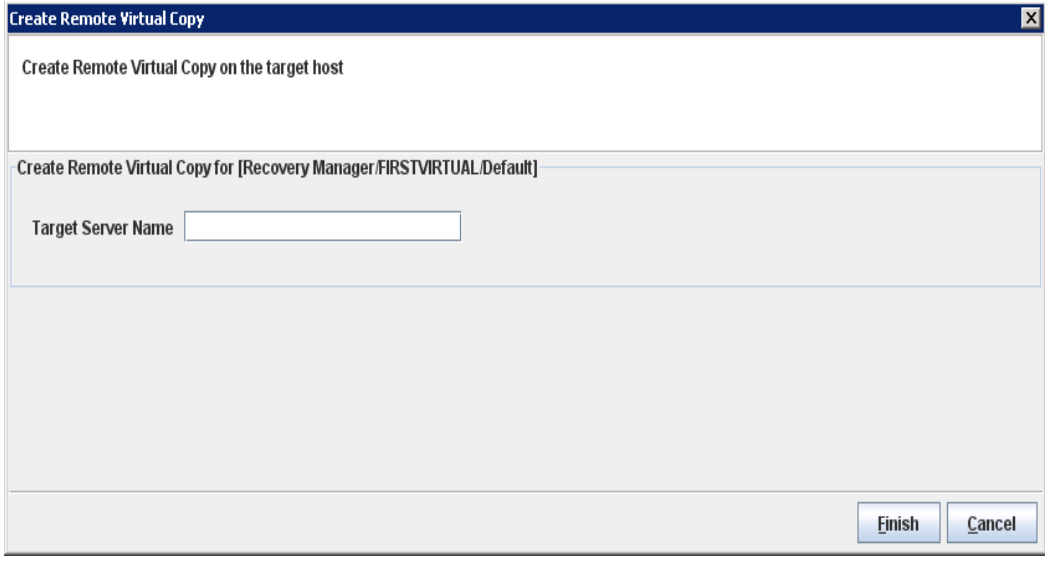
Virtual copies are created at the database level for Microsoft SQL Server and are sent to the Target Server through the Inserv Remote Copy Technology.

Before creating a remote virtual copy, please refer to "Remote Copy Integration" for Remote Copy Requirements and setup.

6.7.1 Creating a Remote Virtual Copy for an Instance

To create a Remote Virtual Copy for an Instance:

1. Navigate through the node tree view on the left pane and locate the instance for which a Remote Virtual Copy is to be created.
2. Click the instance name and the Create Remote Virtual Copy icon appears on the toolbar. You can create a remote virtual copy as follows:
   a. Click on the **Create Remote Virtual Copy** icon on the toolbar.
   b. Right-click on the storage group node and select **Create Remote Virtual Copy** from the pop-up menu.
   c. Select **Create Remote Virtual Copy** from the **Actions** drop-down menu on the menu bar.



6.7.2 Creating a Remote Virtual Copy for a Database

To create a Remote Virtual Copy for a Database:

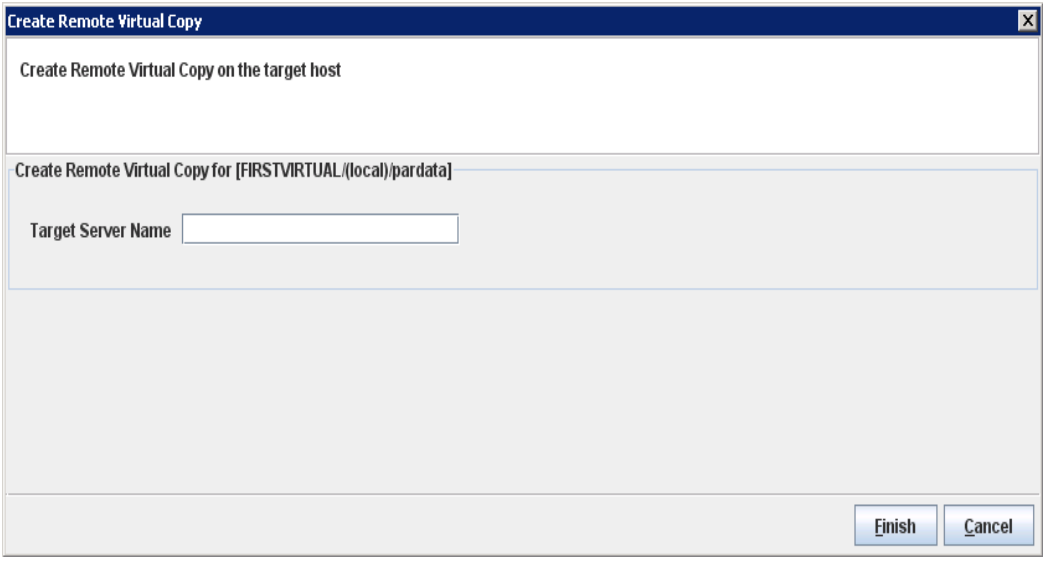
1. Navigate through the node tree view on the left pane and locate the database for which a Remote Virtual Copy is to be created.

2 Click the database name and the Create Remote Virtual Copy icon appears on the toolbar. You can create a remote virtual copy as follows:

a Click on the **Create Remote Virtual Copy** icon on the toolbar.

b Right-click on the storage group node and select **Create Remote Virtual Copy** from the pop-up menu.

c Select **Create Remote Virtual Copy** from the **Actions** drop-down menu on the menu bar.



NOTE: A Target Server name is required. Target Server is a remote server which is a Recovery Manger backup server with the same version of Recovery Manager for SQL installed.

NOTE: Remote Virtual Copy will not be seen on the GUI from Recovery Manager Backup server. However, Virtual Copy can be mounted at an instance or database level from the remote target server which also contains the Recovery Manager Backup server component. To check the Remote Virtual Copy, please use the Recovery Manager GUI or run the CLI `RMSql display` command from the remote target server.

6.8 Mounting a Virtual Copy

A virtual copy can be mounted at an Instance or Database level.

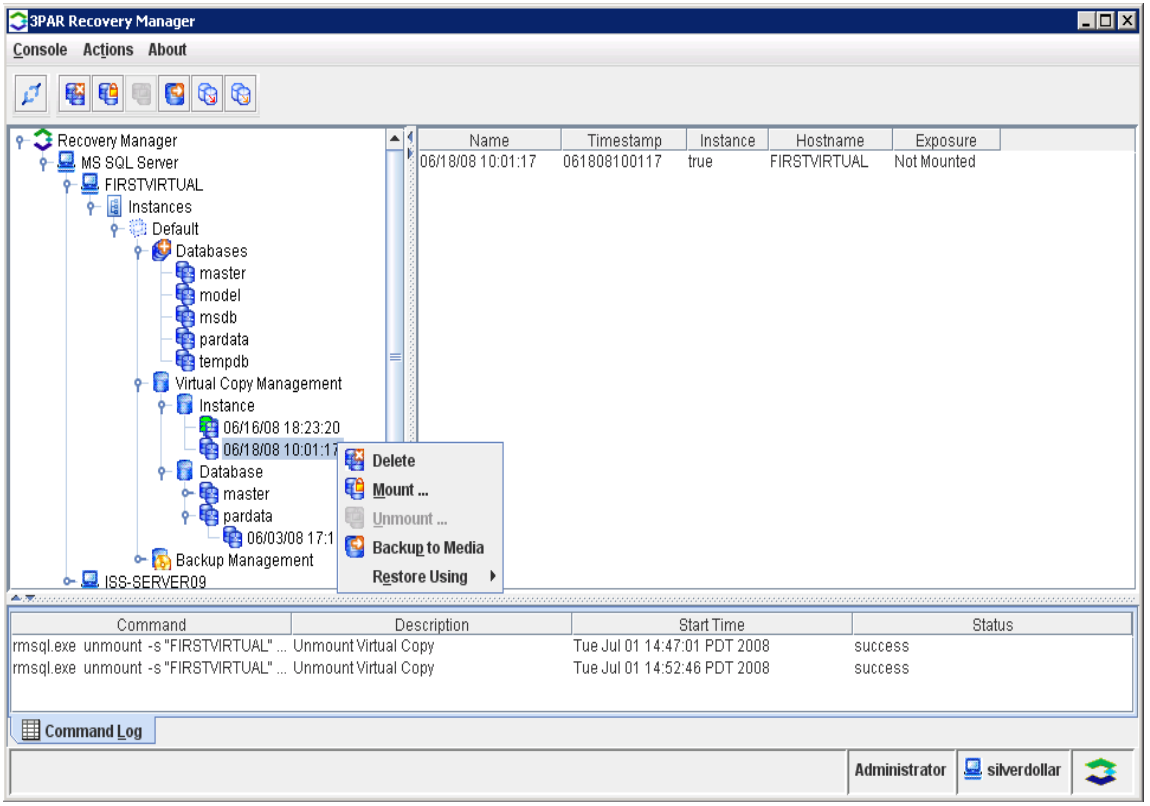
To mount a Virtual Copy under Instance:

1 Click on the timestamp in the left pane tree view, then either:

   a Right-click on the timestamp name and select **Mount ...** .

   b Click on the **Mount** icon in the tool bar.

   c Select **Mount ...** from the **Actions** drop-down menu on the menu bar.

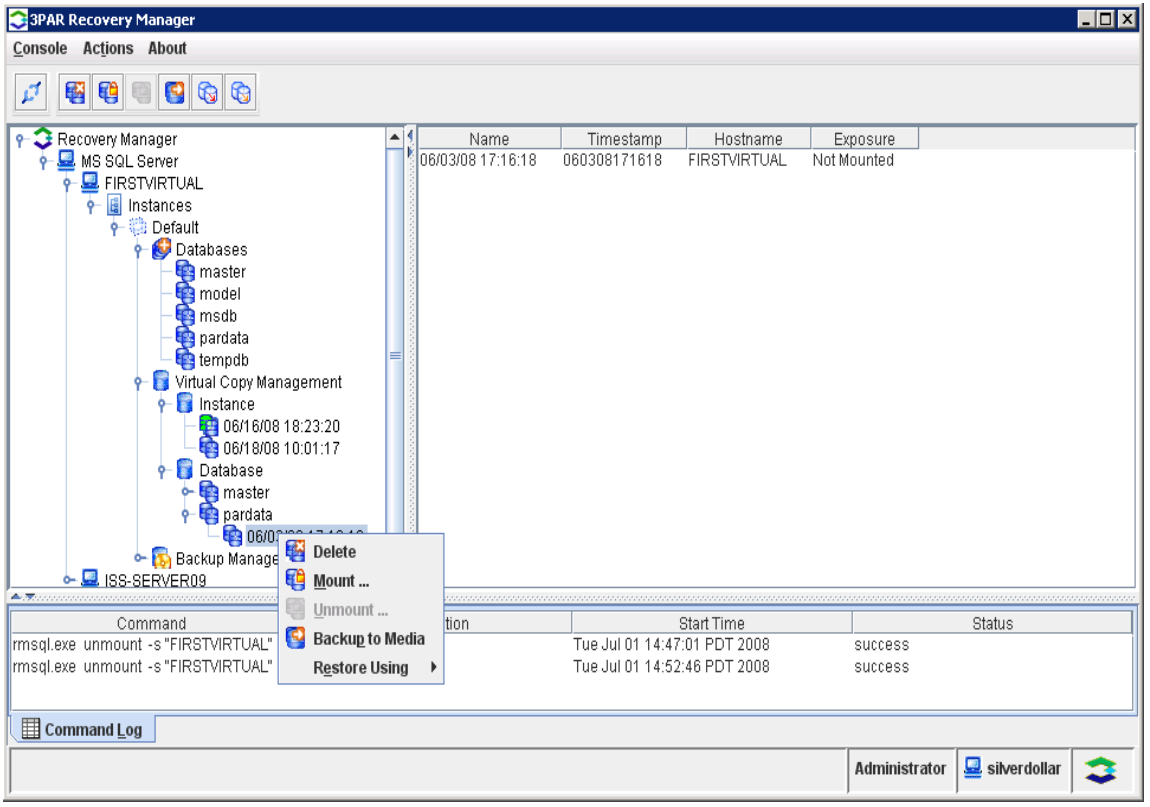


Result: The select mount point dialog appears. Select the mount point to mount the virtual copy by file chooser or enter the mount point directly.

Result: The list view in the right pane is updated for the mounted virtual copy.

The virtual copy status (Exposure column) shows **Exposed locally** and the mount point column shows the absolute path to which the virtual copy was mounted.

Result: The icon for the mounted VC turns green.



To mount a Virtual Copy under Database:

1. Click on the timestamp in the left pane node tree view, then either:
   a. Right-click on the timestamp name and select **Mount** ...
   b. Click on the **Mount** icon in the tool bar.
   c. Select **Mount** ... from the **Actions** drop-down menu on the menu bar.



NOTE: Mounting to the drive letter is not currently supported.

6.9 Unmounting a Virtual Copy

A virtual copy can be unmounted at the Instance or database level.

To unmount a Virtual Copy:

1 Click on the timestamp in the left pane node tree view, then either:

   a Right-click on the database name and select **Unmount ...**

   b Click on the **Unmount** icon in the tool bar.

   c Select **Unmount ...** from the **Actions** drop-down menu on the menu bar.

   Result: The list view in the right pane is updated for the unmounted virtual copy.

   The virtual copy status (Exposure column) shows **Not exposed**.

   Result: The icon for the unmounted Instance or database Virtual Copy turns blue.

6.10 Backing Up a Virtual Copy Using Veritas NetBackup

Recovery Manager allows backup Virtual Copies to media using Veritas NetBackup.

To backup a virtual copy using NetBackup, you must first configure Veritas NetBackup as described in Chapter 4, *Configuring NetBackup for Recovery Manager*.

A backup can be done on Instance or database levels or at the timestamp level. A new virtual copy will be created to backup on either instance or database level. An existing virtual copy can also be backed up with the timestamp associated to the virtual copy.

6.10.1 Backup at Instance Level

This is a fully automatic backup. It creates a virtual copy for a instance, mounts it, backs it up to the media, unmounts it and removes it automatically.

To backup at the instance level:

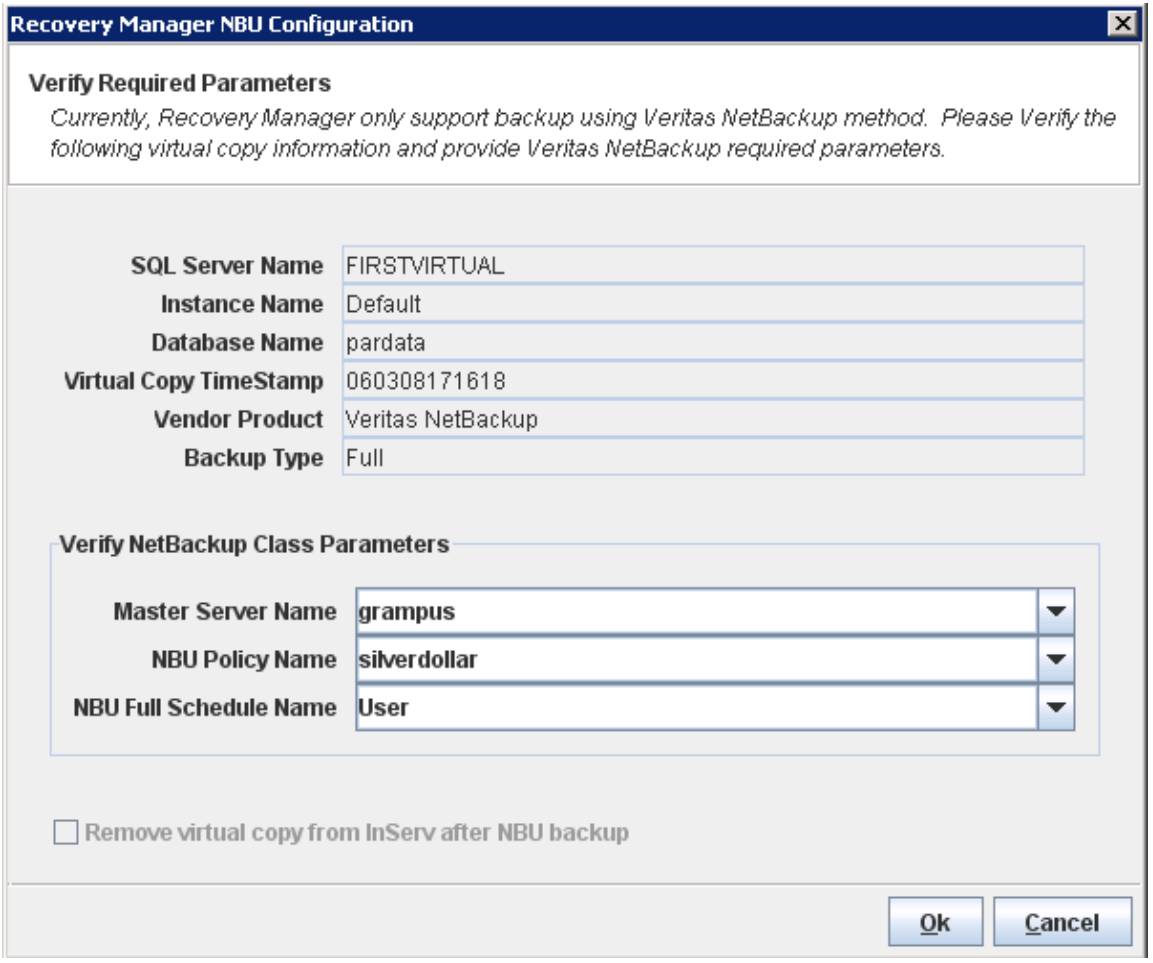
1 Locate the instance node you want to backup in the left pane node tree, then select Backup to Media.

   a Right-click on the Instance Node and select **Backup to Media**.

   b Click on the Backup icon in the tool bar.

   c Select **Backup to Media** from the Virtual Copy drop-down menu on the menu bar.

   Result: The Recovery Manager NBU Configuration dialog box appears:



Fill in the **Master Server Name**, the **NBU Policy Name** and **NBU Full Schedule Name** text boxes. The policy and schedule names should match the names of the policy and schedule that

were set up for use with Recovery Manager in Chapter 4, *Configuring NetBackup for Recovery Manager*.

Check the **Remove virtual copy from InServ after NBU backup** for removing the newly created virtual copy after backup.



NOTE: The Master Server Name, the NBU policy name and NBU Full Schedule Name information will be saved and appear as default values in the next backup and is at the instance level, meaning that each instance can have its own default value set.

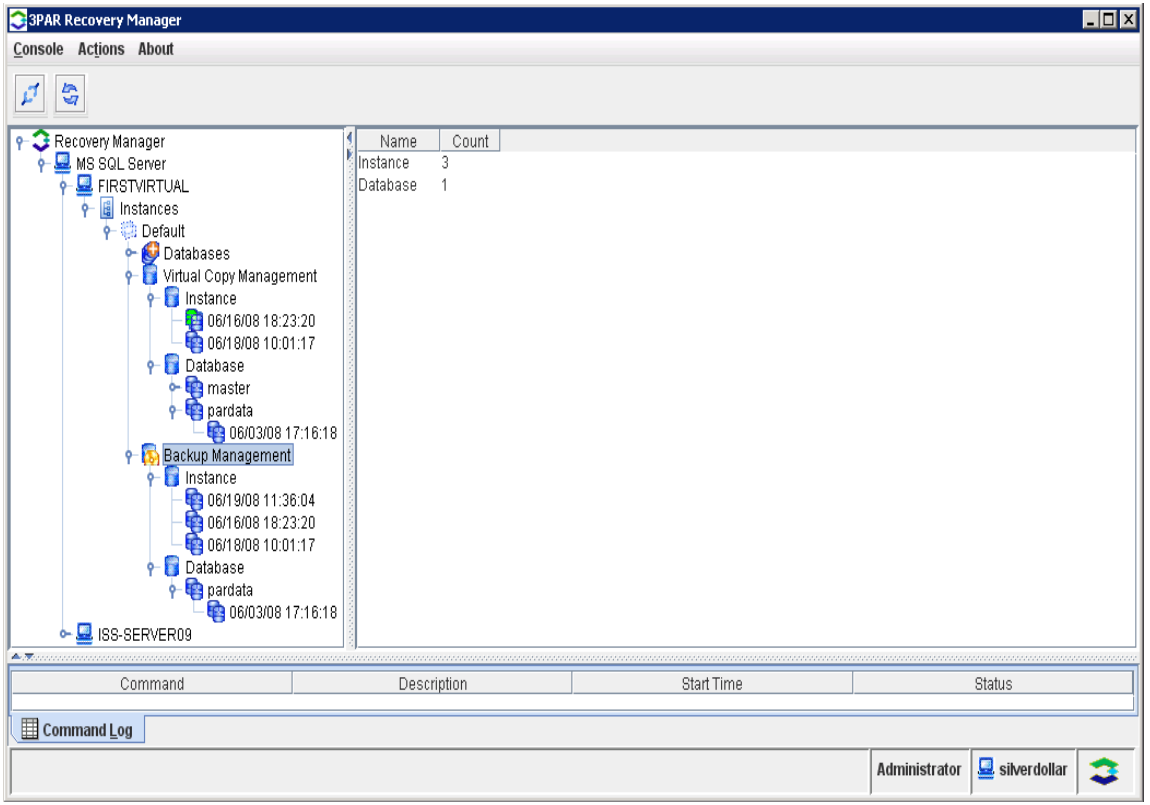


NOTE: While creating a backup at the instance level, a Virtual Copy will be created automatically and will be mounted prior to, and during the backup. After the backup is completed, the Virtual Copy will be dismounted. If a Remove virtual copy is performed from the InServ after the NBU backup, the virtual copy will be retained after backup. Otherwise, it will be removed by default.



NOTE: If the number for the virtual copy is set to the maximum, the oldest virtual copy will be removed if the "Remove the oldest virtual copies" option is checked in the policy before the new virtual copy for the backup is created. If "Remove the oldest virtual copies" option is not checked, an error is generated indicating that the Virtual Copy policy has exceeded its limit.

Result: A newly created timestamp will be added to Backup Management > instance tree node in the Node Tree Pane.



6.10.2 Backup at Database Level

This is a fully automatic backup. It creates a virtual copy for a database, mounts it, backs it up to the media, unmounts it and removes it automatically.

To backup at the database level:

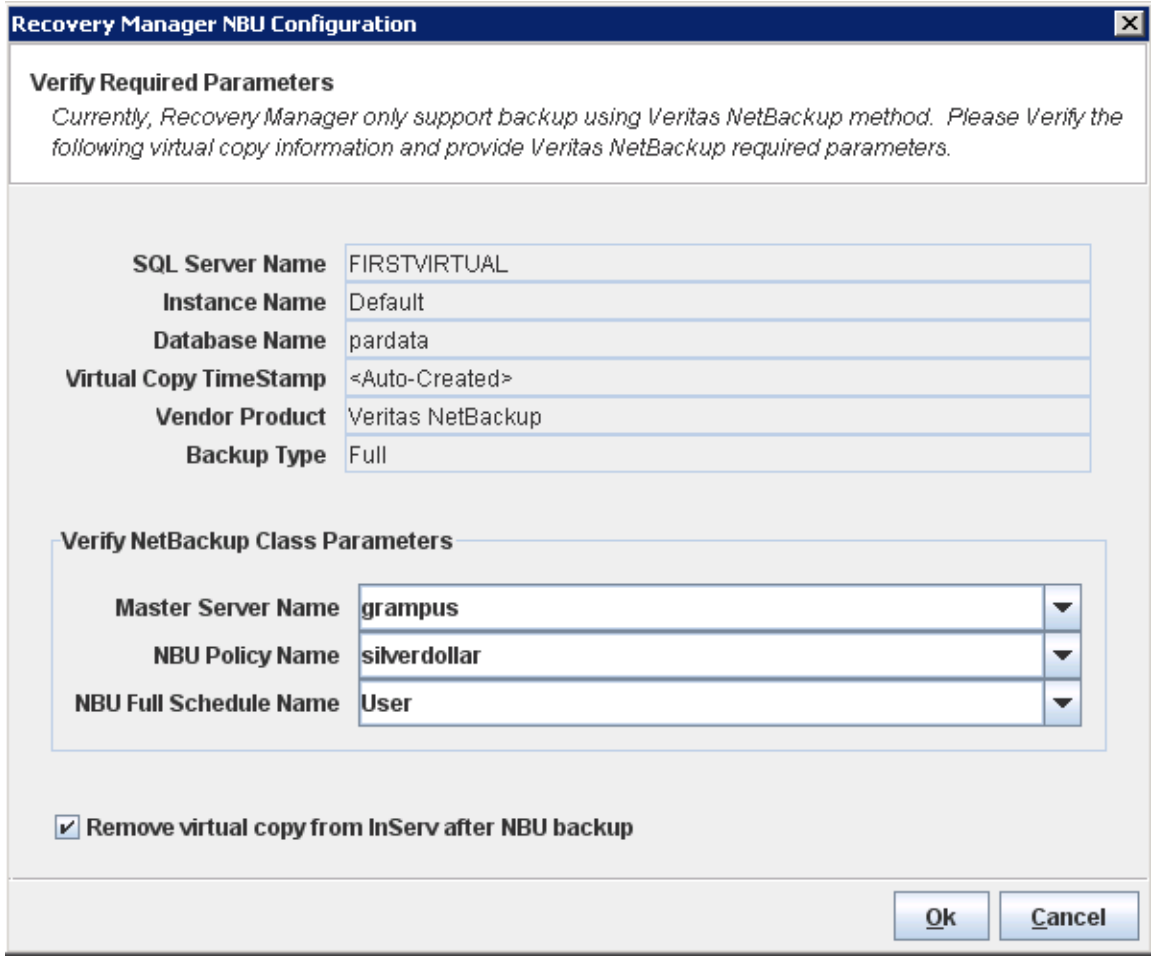
1 Locate the database node you want to backup in the left pane node tree, then select Backup to Media.

   a Right-click on the Database Node and select **Backup to Media**.

   b Click on the Backup icon in the tool bar.

   c Select **Backup to Media** from the Virtual Copy drop-down menu on the menu bar.

Result: The Recovery Manager NBU Configuration dialog box appears:



Fill in the **Master Server Name**, the **NBU Policy Name** and **NBU Full Schedule Name** text boxes. The policy and schedule names should be the names as the policy and schedule that were set up for use with Recovery Manager in Chapter 4, *Configuring NetBackup for Recovery Manager*.

Check the **Remove virtual copy from InServ after NBU backup** for removing the newly created virtual copy after backup.

Result: A newly created timestamp will be added to Backup Management ->Database-><Database Name> tree node in the Node Tree Pane.



NOTE: The Master Server Name, the NBU policy name and NBU Full Schedule Name information will be saved and appear as default values in the next backup and is at the database level, meaning that each database can have its own default value set.



NOTE: While creating a backup at the database level, a Virtual Copy will be created automatically and will be mounted prior to, and during the backup. After the backup is completed, the Virtual Copy will be dismounted. If a Remove virtual copy is performed from the InServ after the NBU backup, the virtual copy will be retained after backup. Otherwise, it will be removed by default.

6.11 Creating a Backup at the Timestamp Level

This is a fully automatic backup for existing virtual copies to the media.

To backup at the timestamp level:

1. Locate the timestamp node you wish to backup in the left pane node tree, then select **Backup to Media**.
   a. Right-click on the **Timestamp Node** and select **Backup to Media**.
   b. Click on the **Backup** icon in the tool bar.
   c. Select **Backup to Media** from the Actions drop-down menu on the menu bar.

Result: The backup dialog appears. However, the check box for **Remove virtual copy from InServ after NBU backup** will be disabled.



NOTE: No virtual copy will be created before the backup is performed or will be removed after backup is completed.



NOTE: After the backup, the status of the virtual copy remains the same as before the backup started.

Result: Once a backup is started, it appears under the **Backup Management** node under the relevant instance or database name with a tape icon. The backup is given a timestamp that is used as its name. When backup is done, the tape icon will be replaced by a normal database icon and the backup time stamp will be shown up on the right information pane.

6.12 Deleting Backup from the Repository

To remove a backup repository from the GUI, use one of the methods described below:

1. Right-click on the backup timestamp node and select **Delete Backup**.
2. Click on the **Delete Backup** icon in the tool bar.
3. Select **Delete Backup** from the **Actions** drop-down menu on the menu bar.

Result: The virtual copy is removed from the Recovery Manager repository only.

6.13 Restoring an Instance or Database from Virtual Copy using Volume Restore



NOTE: Restoring the volume from the snapshot is not supported in a shared volume environment such as a Microsoft Cluster environment. However, you can manually set the cluster resource to the maintenance mode, remove LUNs exported to all cluster hosts, and promote the virtual copy to its parent base volume with the InForm CLI commands. For more information, see the *3PAR InForm OS CLI Administrator's Manual*.

The volume restore feature provides a quicker method for restoring an instance or a database from a virtual copy.

Through this process, all snapshots will be promoted to the base volume. However, when using this method, all data on the original volume will be overwritten.

To perform a volume restore from a virtual copy, follow these steps:

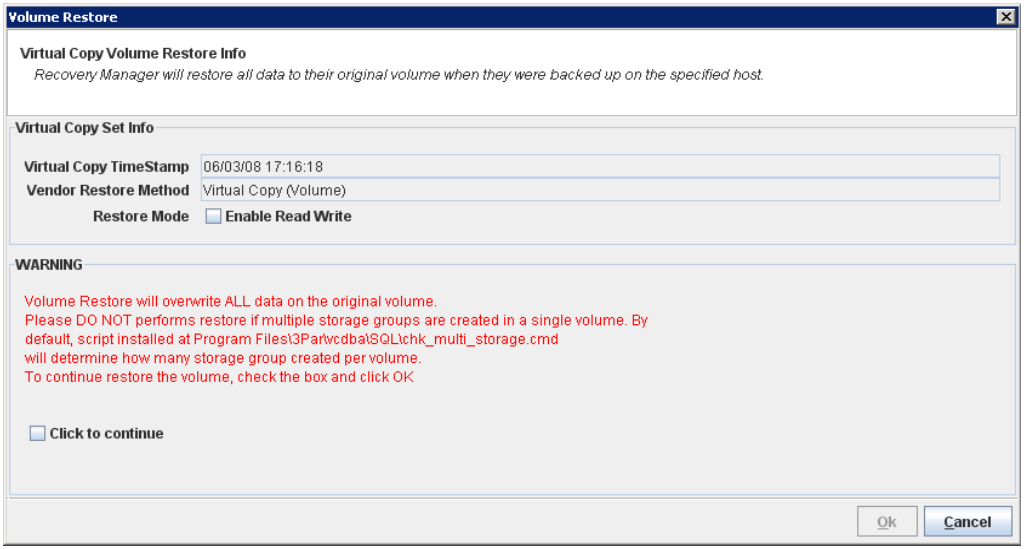
1. Locate the virtual copy timestamp node you wish to restore under Virtual Copy Management in the left pane node tree, then select the volume restore using one of the following methods:

a. Right-click on the virtual copy timestamp node and select **Restore Using->Volume**.

b. Click on the **Volume Restore** icon in the tool bar.

c. Select **Restore Using->Volume from** the **Actions** drop-down menu on the menu bar.

Result: The Restore Instance or Database from Virtual Copy dialog appears.



WARNING: When performing a Virtual Copy Volume Restore, all data on the original volume will be overwritten. Please DO NOT perform restore if multiple storage groups are created in a single volume. By default, script installed at Program Files/3Par/vcdba/SQL/chk_multistorage.cmd will determine how many storage groups created per volume.

6.14 Restoring an Instance or Database from Virtual Copy Using File Copy



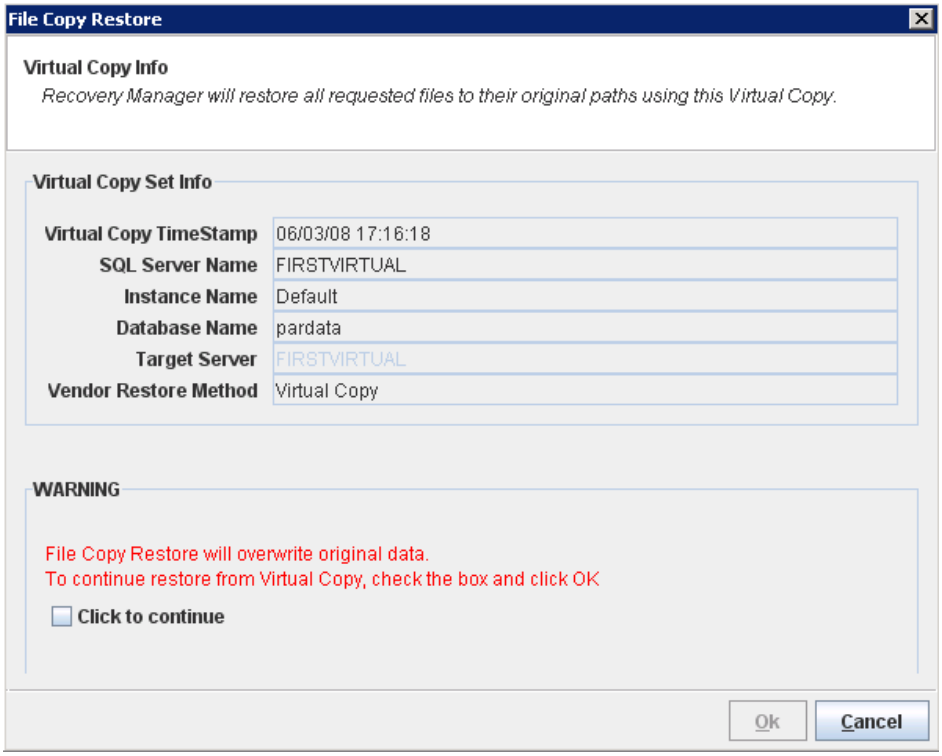
NOTE: Restoring the volume from the snapshot is not supported in a shared volume environment such as a Microsoft Cluster environment. However, you can manually set the cluster resource to the maintenance mode, remove LUNs exported to all cluster hosts, and promote the virtual copy to its parent base volume with the InForm CLI commands. For more information, see the *3PAR InForm OS CLI Administrator's Manual*.

Once a virtual copy is created, the Restore copy option is enabled in the timestamp node and virtual copies can be restored to their original location.

To restore from a virtual copy:

1. Locate the virtual copy timestamp node you wish to restore under **Virtual Copy Management** in the left pane node tree, then select Restore using one of the following methods:
   a. Right-click on the virtual copy timestamp node and select **Restore Using->File Copy**.
   b. Click on the **File Copy Restore** icon in the tool bar.
   c. Select **Restore Using->File Copy** from the **Actions** drop-down menu on the menu bar.

d **Result:** The Restore Instance or Database from Virtual Copy dialog appears



NOTE: Virtual Copy (**File Copy**) will be displayed at **Vendor Restore Method**.

NOTE: When restoring from a virtual copy, it will only restore to the original location. Therefore, the **Alternate Location** will be disabled.

6.15 Restoring a Virtual Copy Using Veritas NetBackup

Once a virtual copy is backed up to media, the **Restore** menu option is enabled. Backups can be restored back to their original location (the volume(s)) or directory locations of the virtual copies that were backed up, not original SQL data locations, or an alternate location can be specified.

To restore a virtual copy that has been backed up using NetBackup:

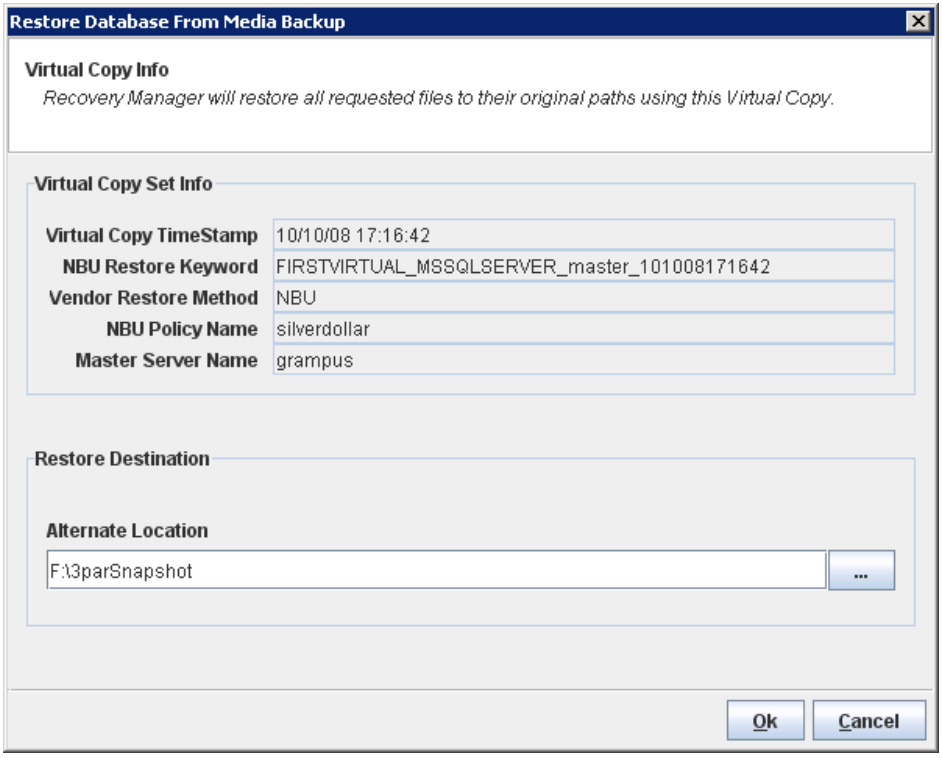
1 Locate the backup timestamp node under **Backup Management Node** you wish to restore in the left pane node tree, then select **Restore** using one of the following methods:

   a Right-click on the backup timestamp node under **Backup Management Node** and select Restore.

   b Click on the **Restore** icon in the tool bar.

   c Select **Restore** from the **Actions** drop-down menu on the menu bar.

   Result: The Restore dialog appears.



2 To restore to an alternate location, enter the location in the text box provided or select from file chooser.

3 Click **OK**.

6.16 Removing a Virtual Copy

To remove a virtual copy from the InServ Storage Server and the Recovery Manager repository, the virtual copy timestamp must have a status of **Available**. To remove a virtual copy, use one of the methods described below:

1. Right-click on the virtual copy timestamp node and select **Remove**.
2. Click on the **Remove** icon in the toolbar.
3. Select **Remove** from the **Actions** drop-down menu on the menu bar.

Result: The virtual copy will be removed from both InServ Storage Server and the Recovery Manager repository.



NOTE: A virtual copy cannot be removed if it is mounted on a host server.



NOTE: If the Backup server has a mixture of instance level snapshots and database level snapshots within the same instance, user is expected to see out-of-sync VV list caused by the snapshots taken at different level.

7 Using the Recovery Manager CLI

In this chapter

7.1 Recovery Manager Command Line Interface (CLI) 7.2

7.2 Recovery Manager Commands 7.3

RMSql backup 7.4

RMSql create 7.6

RMSql deletebackup 7.9

RMSql analyze 7.10

RMSql display 7.12

RMSql mount 7.14

RMSql remove 7.16

RMSql restore 7.17

RMSql unmount 7.20

RMSql policy 7.21

RMSql listbackup 7.23

RMSql report 7.24

RMSql repository 7.25

RMSql port 7.26

RMSql portconfig 7.27

RMSql protocol 7.28

RMSql time 7.29

RMSql timeconfig 7.30

7.3 Running Recovery Manager as a Scheduled Job 7.30

7.1 Recovery Manager Command Line Interface (CLI)

Recovery Manager for Microsoft SQL provides a Command Line Interface (CLI). The CLI allows users to:

- Create virtual copies
- Remove virtual copies
- Mount virtual copies
- Unmount virtual copies
- Backup virtual copies to media using Veritas NetBackup
- Restore virtual copies
- Restore backed up virtual copies using Veritas NetBackup
- Display virtual copies currently in the repository
- List virtual copies currently in the backup pool
- Delete virtual copies backup entry currently in the backup pool
- Query and change repository location
- Query and change the port location
- Query and change the protocol
- Change the backup time and volume restore time interval
- Gather the environmental statistics and create a log file for diagnosis

- Analyze and synchronize the volumes in the Inserv and the virtual copies in the repository

The following sections describe how to administer Recovery Manager for Microsoft SQL using the CLI.

7.2 Recovery Manager Commands

The following commands can be run from the Command Line Interface:

- `RMSql analyze`
- `RMSql backup`
- `RMSql create`
- `RMSql deletebackup`
- `RMSql display`
- `RMSql mount`
- `RMSql remove`
- `RMSql report`
- `RMSql restore`
- `RMSql unmount`
- `RMSql policy`
- `RMSql listbackup`
- `RMSql repository`
- `RMSql port`
- `RMSql portconfig`
- `RMSql protocol`
- `RMSql time`
- `RMSql timeconfig`

COMMAND

`RMSql backup`

SYNTAX

```
RMSql backup -s <SQL Server> [-i <SQL instance>]
-d <database> [-t <timestamp> | -r ] [-ts <Target Server>]
-p <policyName> -u <fullScheduleName> -M <masterServer>
```

DESCRIPTION

The `RMSql backup` command backups a virtual copy to the specified media. The backup can be made of an existing virtual copy or one that is created through this command. The virtual copy will be mounted to the default destination, if not mounted by user.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - SQL server associated with the database.
- `-i <SQL instance>` - SQL server instance associated with the database.

  This is an optional argument: MSSQLSERVER (local) will be used as the default if it is not specified.
- `-d <database>` - The database for the virtual copy.
- `-t <timestamp>` - The timestamp for the virtual copy to backup.

  This is an optional argument: If not specified, a point-in-time virtual copy will be created for backup purposes, and then removed after the backup has completed. The user specified timestamp will be retained after the backup is completed.
- `-r` - This is an optional argument: Retain the virtual copy created without the -t option.
- `-ts <Target Server>` - This is an optional argument: Target Server for the backups.
- `-p <policyName>` - NetBackup policy name.
- `-u <fullScheduleName>` - Name of the schedule for a full or user backup.
- `-M <masterServer>` - NetBackup master server name.

EXAMPLE:

The following command will perform a backup of virtual copy 020405162103 with the necessary NetBackup parameters.

```
RMSql backup -s inst1 -d test_db -t 020405162103 -p test -u full -M host1
```

.



NOTE: Use quotes (" ") where spaces are included



NOTE: During backup, snapshot volumes of the virtual copy will be surfaced to the operating system with a default mount point under the `%SystemDrive%\3parsnapshot` directory. Therefore `%SystemDrive%` must use NTFS in order to support the volume mount point.

COMMAND

`RMSql create`

SYNTAX

```
RMSql create -s <SQL Server> [-rc <target server> [-w <waittmie>
[-i <SQL instance>] -d <database>
```

DESCRIPTION

The `RMSql create` command creates a virtual copy of a SQL database or databases under an instance on the primary host where the SQL server instance is running. The virtual copy will then stay online and can be mounted to a host later on for off-host processing. The `RMSql create` command also allows the user to create a synchronization to a Remote Copy volume and create another virtual copy at the secondary site as a backup. As with Recovery Manager for SQL Server, this command only supports the online backup of an entire SQL database. Therefore, you must make sure the database is online before continuing with the virtual copy creation.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - SQL server to which the database belongs.

Optional: uses the current server name as the default.

The following example will create a snapshot of the databases under the default instance on SQL server (SQLServer).

```
RMSql create -s SQLServer
```

- `-i <SQL instance>` - SQL server instance for which database belongs.

Optional: Uses MSSQLSERVER as the default if not specified.

- `-d <database>` - The database from which to create a virtual copy.

Optional: Without the `-d` parameter, the instance level of virtual copy is used.

- `-w <waittime>` - Wait time unit in seconds. By default, the wait time is 120 seconds. This wait time is only used by remote copy service to poll the status of the remote copy before creating a virtual copy at the secondary site.

- `-rc <target server>` - For the Remote Copy Service, a Recovery Manager SQL server with Recovery Manager SQL installed must be located at the secondary site.

EXAMPLES

The following command will create a snapshot for the database test on SQL server (SQLServer).

```
RMSql create -s SQLServer -d test
```

The following command will create a snapshot for the database test on Virtual SQL server(SQLVirtualServer) on a cluster.

```
RMSql create -s SQLVirtualServer -d test
```

The following command will create a RC for the database test on Virtual SQL server(SQLVirtualServer) on a cluster to a secondary site server SecondaryServer.

```
RMSql create -s SQLVirtualServer -d test -rc Secondaryserver
```



NOTE: Use quotes ("") for names that include spaces.



NOTE: VSS waits up to 10 seconds for the writers to freeze I/O, and this interval is not configurable. In most cases 10 seconds is a sufficient window to successfully create a virtual copy. However, in some circumstances under very heavy loads, 10 seconds may not be enough time and the virtual copy might fail on this attempt, which is "by design." VSS does not automatically retry the shadow copy creation, you must retry the virtual copy creation using the Recovery Manager GUI or Command Line Interface.



NOTE: A backup document in XML format is created during the virtual copy creation. The document contains the current up to date system information and can be used for restore purposes.



NOTE: The snapshot volume set is limited up to 100 volumes which means each database can spread over up to 100 volumes.



NOTE: The Instance level operation is only granted if the current user is given the **sysadmin** server role for security permission in SQL Server.



NOTE: Database level operation is only granted if the database owner or user is given **sysadmin** server role for security permission.

COMMAND

`RMSql deletebackup`

SYNTAX

```
RMSql deletebackup -s <SQL Server> [-i <SQL instance>] [-d <database>]
-t <timestamp>
```

DESCRIPTION

The `RMSql deletebackup` command only removes the specified backup entry in the Recovery Manager repository. It does not remove NBU's entry and does not remove the actual images backed up by NBU.

PARAMETERS

The following options are supported:

`-s <SQL Server>` - SQL server to which the backup belongs.

`-i <SQL instance>` - Optional: SQL instance associated with the database. MSSQLSERVER (local) will be used if not specified.

`-d <database>` - The database level backup to be removed. If not specified, instance level backup is removed.

`-t <timestamp>` - The backup entry timestamp.

COMMAND

`RMSql analyze`

SYNTAX

```
RMSql analyze -s <SQL Server> [-f] [-i <SQL instance>] [-d <database>]
```

DESCRIPTION

The `RMSql analyze` command verifies whether the corresponding snapshot volumes in the InServ match those managed by Recovery Manager. If you are running at instance level, it also lists the database snapshot volume that is managed by the Recovery Manager.

PARAMETERS

The following options are supported:

`-s <SQL Server>` - SQL server for which database or instance belongs.

Optional: Uses current server name as default.

`-i <SQL instance>` - This is an optional argument: The SQL server instance to which the database belongs. MSSQLSERVER (local) will be used as the default if not specified.

`-d <database>` - Optional: To validate the virtual copy created for a given database.

`-f` - Force the clean-up of any inconsistencies between the InServ snapshot created by the Recovery Manager and the repository.



NOTE: Only use the `-f` parameter after verifying that the volume is not managed by Recovery Manager.



NOTE: If the Backup server has a mixture of instance level snapshots and database level snapshots within the same instance, user is expected to see out-of-sync VV list caused by the snapshots taken at different level.

EXAMPLES

The following example validates the snapshots in the InServ that belong to db1, and verifies that they match the snapshots in the Recovery Manager repository.

```
RMSql analyze -s SQLserver -d db1
```

COMMAND

`RMSql display`

SYNTAX

```
RMSql display -s <SQL Server> [-rc <target server>] [-i <SQL instance>]
[-d <database>] [-t <timestamp>]
```

DESCRIPTION

The `RMSql display` command displays all virtual copies in the server at a local site or remote site.

PARAMETERS

The following options are supported:

`-s <SQL Server>` - SQL server associated with the database.

`-i <SQL instance>` - SQL instance associated with the database.

Optional: Uses MSSQLSERVER as the default if not specified.

`-d <database>` - Optional: Display the virtual copy created for a given database.

`-rc <target server>` - For the Remote Copy Service, a Recovery Manager SQL server with Recovery Manager SQL installed must be located at the secondary site.

`-t <timestamp>` - Optional: To display detailed information about an individual virtual copy existing in the local repository.



NOTE: `RMSql display -t` for virtual copy from a secondary site is not supported.

EXAMPLES

The following command will display the virtual copies for server1.

```
RMSql display -s server1
```



NOTE: Use quotes (" ") where spaces are included.

COMMAND

`RMSql mount`

SYNTAX

```
RMSql mount -s <SQL Server> [<-ro>] -t <timestamp>
[-m <mount point>] [-ts <host name>]
```

DESCRIPTION

The `RMSql mount` command mounts a virtual copy created by the `RMSql create` command. In order to mount the virtual copy devices, an available mount point must be created on the NTFS volume before the command is executed.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - The SQL server associated with the database.
- `-t <timestamp>` - The timestamp for the virtual copy to be mounted.
- `-ro` - Specifies that the volume is to be mounted as READ-ONLY. By default, the volume is mounted in the READ-WRITE mode.
- `-m <mount point>` - This is an optional argument: The mount point for virtual copy devices. The default mount point: %SystemDrive%\3parSnapshot.
- `-ts <hostname>` - This is an optional argument: The host name where the virtual copy devices have been mounted. The default value will use the current host.



WARNING: The proper way to dismount a mounted volume that has been mounted by the `RMSql mount` command is with the `RMSql unmount` command. Do not manually dismount a mounted volume that has been mounted by the `RMSql mount` command.



NOTE: When mounting in read-write mode, the VDS API is used to make the volume read-writeable. However if there is another operation on the disk and the device is locked, the volume can not be unmounted. If this happens, the following error message is displayed:

```
ERROR: COM call "pVolumeMF->Dismount(TRUE,FALSE)" failed.
Returned HRESULT = 0x80042413
Error text: <Unknown error code>
Error: There was an expected HRESULT error catched:
0x80042413 <Unknown error code>
```

This means VDS_E_DEVICE_IN_USE. You could restart the VDS service or reboot the machine to resolve the problem.



NOTE: 3PAR only supports snapshot exposure to a host.

COMMAND

`RMSql remove`

SYNTAX

```
RMSql remove -s <SQL Server> -t <timestamp>
```

DESCRIPTION

The `RMSql remove` command removes the virtual copy from the InServ Storage Server and the backup XML document. If the virtual copy is mounted, it will not be removed.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - The SQL server to which the database belongs.
- `-t <timestamp>` - The timestamp for the virtual copy to remove.



NOTE: After the virtual copy has been removed, it cannot be used to restore from the backup XML document, but it can still be restored from the backup media if the virtual copy has been backed up with the `RMSql backup` command and backup entry still remains available.

COMMAND

`RMSql restore`

SYNTAX

```
RMSql restore -s <SQL Server> [-snap [-rw]] [-i <instance>]
[<-d database>] <-t timestamp> [[-ts <target server>] -M <masterServer>
[-a <altLocation>]]
```

DESCRIPTION

The `RMSql restore` command restores a virtual copy from the Netbackup media server when the virtual copy resides on an InServ server. When restoring a virtual copy, you have the option of file copy restore and snap volume restore. To restore from a virtual copy, a previously saved backup document in XML format can be used to restore data back to the original locations. To restore from a media, restore the image from the media where it was previously backed up.

As with Recovery Manager for SQL Server, file copy restore only supports the online restore of an entire SQL database, therefore you must make sure the database is online before continuing with the file copy restoration. If the database is no longer online, you need to mount the virtual copy and perform the file copy manually.

The Netbakup policy and the Master server information are required in order to retrieve the backup image and an alternate location can be specified for the destination or the original snapshot mount volume's location will be used to restore the data.



NOTE: For SQL instance level File copy restore in a MSCS environment see *SQL Instance Level File Copy Restore in an MSCS Environment* on page 8.5.

PARAMETERS

The following options are supported:

`-s <SQL Server>` - SQL server to which the database belongs.

Optional: Uses the current server name as the default.

`-i <SQL instance>` - SQL server instance to which the database belongs.

Optional: Uses MSSQLSERVER as the default if not specified.

`-d <database>` - The database from which to create a virtual copy.

Optional: Without the `-d` parameter, the instance level restore will be performed.

`-t <timestamp>` - The timestamp for the virtual copy to restore

`-ts <target server>` - Target Server to do the restore (Media Restore only).

Optional: Uses the current server name as the default.

`-M <masterServer>` - Specifies the NBU master server name.

`-a <altLocation>` - Specifies an alternate restore location.

`-snap` - Promotes the snapshot back to the base volume for faster restore.



NOTE: The `-snap` option for restoring the volume from the snapshot is not supported in a shared volume environment such as a Microsoft Cluster environment. However, you can manually remove LUNs exported to all host cluster nodes and promote the virtual copy to its parent base volume with the InForm CLI commands. Refer to the *3PAR InForm OS CLI Administrator's Manual* for instructions on removing VLUNs and promoting the snapshot.

`-rw` - Use the snap volume restore to restore the ReadWrite volume for the snapshot. By default, ReadOnly volume is used.



NOTE: Using ReadWrite snapshot volumes to do the restore is necessary to restore the latest changes on the snapshot to the original volumes. Using ReadOnly snapshot volumes to do the restore is necessary for point-in-time restores.



NOTE: For the `-snap` option for volume restore, you can only promote a base volume that is exposed to a host. For VLUNs exposed through the port (VLUN `type: matched set`), you must remove them manually by issuing InServ `removevlun` command before promoting.

The syntax is: `removevlun vvname vlunNumber hostname port`
For example:

```
root@InServ:server058# removevlun basevv1 108 Myhost 0:3:2
```

Refer to the *3PAR InForm OS CLI Administrator's Manual* for instructions on removing VLUNs.

EXAMPLES

The following command performs a restore of virtual copy 020405162103 to drive j:\.

```
RMSql restore -s server1 -d test_db -t 020405162103 -M server1 -a J:\
```

The following command performs a file copy restore of virtual copy 020405162103.

```
RMSql restore -s server1 -d test_db -t 020405162103
```

The following command performs a snap volume restore of virtual copy 020405162103.

```
RMSql restore -snap -s server1 -d test_db -t 020405162103
```



NOTE: Use quotes (" ") where spaces are included.

COMMAND

`RMSql unmount`

SYNTAX

```
RMSql unmount -s <SQL Server> -t <timestamp>
```

DESCRIPTION

The `RMSql unmount` command dismounts a currently mounted virtual copy that was mounted by the `RMSql mount` command. The virtual copy's mounted status must not be "Available" before it can be unmounted.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - The SQL server to which the database belongs.
- `-t <timestamp>` - The timestamp of the virtual copy to dismount.



NOTE: During the dismount operation, the Recovery Manager removes the mount point from the mounted volumes, and then removes the snapshot from the host.

COMMAND

`RMSql policy`

SYNTAX

```
RMSql policy -s <SQL Server> [-q] [-i <SQL instance>]
[-n <number of VC>] [<-r>] [-n <database level policy>]
[-in <instance level policy>]
```

DESCRIPTION

The `RMSql policy` command defines the policy for the virtual copy.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - The SQL server to which the database belongs.
- `-i <SQL instance>` - This is an optional argument: The SQL server instance to which the database belongs. MSSQLSERVER (local) will be used, if the default is not specified.
- `-r` - Optional: Retain the oldest Virtual Copy when the number of Virtual Copies exceeds the limit. The default is to remove them.
- `-q` - query current policy status
- `-n <database level policy>` - Maximum number of database level virtual copies to be retained.
- `-in <instance level policy>` - Maximum number of instance level virtual copies to be retained.



NOTE: The sum of `-n` and `-in` can not exceed 500.

EXAMPLE

The following command maintains the maximum number of virtual copies to 5 at all times for the default SQL instance:

```
RMSql policy –n 5 -s <SQL Server>
```



NOTE: The `-r` option is used to retain the oldest Virtual Copy when the number of Virtual copies exceeds the limit set by policy. Therefore, no new Virtual Copies will be created when the limit is exceeded and the `-r` option specified.

COMMAND

`RMSql listbackup`

SYNTAX

```
RMSql listbackup -s <SQL Server> [-i <SQL instance>]
[-d <database>] [-t <timestamp>]
```

DESCRIPTION

The `RMSql listbackup` command displays all virtual copies being backed up through the NetBackup.

PARAMETERS

The following options are supported:

- `-s <SQL Server>` - The SQL server to which the database belongs.
- `-i <SQL instance>` - This is an optional argument: The SQL server instance to which the database belongs. MSSQLSERVER (local) will be used as the default if not specified.
- `-d <database>` - This is an optional argument: The database from which to create a virtual copy.
- `-t <timestamp>` - This is an optional argument: The timestamp of the virtual copy to list the backup record.



NOTE: When a timestamp is provided, the detailed information about the virtual copy backed up will be displayed.

COMMAND

`RMSql report`

SYNTAX

`RMSql report -s <WindowsHost> -output <OutputDirectory>`

DESCRIPTION

The `RMSql report` command collects the environmental information from a specified Windows host and creates a report for diagnosis purposes.

PARAMETERS

The following options are supported:

- `-output <OutputFolder>` - The full path to the folder where the report will be generated.
- `-s <WindowsHost>` - Name of the specified Recovery Manager enabled host.

EXAMPLE

The following command retrieves the platform information from the host `Grampus` and saves the report in `c:\MyReport`.

```
RMSql report -s Grampus -output c:\MyReport
```

COMMAND

```
RMSql repository
```

SYNTAX

```
RMSql repository [-a <alter location>] [-r]
```

DESCRIPTION

After Recovery Manager for SQL is installed, by default, the repository directory is located under the Recovery Manager install directory. To provide the highest availability access to the repository, you can change the repository destination to the location where the 3PAR virtual volume resides.

PARAMETERS

The following options are supported:

- `-a <alternateLocation>` - An alternate location where the Recovery Manager stores the data.
- `-r` - Retains the current data location, and only makes changes to the registry values. This option is used when the Recovery Manager gets reinstalled after operating under alternate destinations.

EXAMPLE

The following command moves the repository directory to `d:\3parvv` and updates the registry database as well.

```
RMSql repository –a d:\3parvv
```

COMMAND

`RMSql port`

SYNTAX

```
RMSql port [-n <no>] [-s <SQL Server>]
```

DESCRIPTION

After Recovery Manager for SQL is installed, by default, the port used for communication between the backup server and production server is 9932. If necessary, you can specify an alternate port.

PARAMETERS

The following options are supported:

- `-n <no>` - Port number (default is 9932).
- `-s <SQL Server>` - This is an optional argument: The SQL server to which the backup server is going to manage. The current server name will be used as the default.

EXAMPLES

The following command will change the local port to 9999.

```
RMSql port -n 9999
```

The following command will change the server SQLserver port to 9999.

```
RMSql port -n 9999 -s <SQL Server>
```



NOTE: After executing the `port` command, you must restart the Recovery Manager for SQL service.



NOTE: In Recovery Manager version 3.0, both the `portconfig` and `port` commands are supported. The `port` command is deprecated and will not be available in the next release. Please use the `portconfig` command.

COMMAND

`RMSql portconfig`

SYNTAX

`RMSql portconfig [-n <no>] [-s <SQL Server>]`

DESCRIPTION

After Recovery Manager for SQL is installed, by default, the port used for communication between the backup server and production server is 9932. If necessary, you can specify an alternate port.

PARAMETERS

The following options are supported:

- `-n <no>` - Port number (default is 9932).
- `-s <SQL Server>` - This is an optional argument: The SQL server to which the backup server is going to manage. The current server name will be used as the default.

EXAMPLES

The following command will change the local port to 9999.

```
RMSql portconfig -n 9999
```

The following command will change the server SQLserver port to 9999.

```
RMSql portconfig -n 9999 -s <SQL Server>
```



NOTE: After executing the `portconfig` command, you must restart the Recovery Manager for SQL service.

COMMAND

`RMSql protocol`

SYNTAX

`RMSql protocol [<-pt protocol>] [<-s Server>]`

DESCRIPTION

After Recovery Manager for SQL is installed, by default, the protocol used for communication between the backup server and production server is TCP. As an alternative, you can change the TCP to HTTP.

PARAMETERS

`-pt <protocol>` - Specifies alternate protocol (default is TCP).

`-s <Server>` - Specifies the SQL server.

EXAMPLES

The following command will query the protocol that is currently in use.

```
RMSql protocol
```

The following command will change the local protocol from TCP to HTTP.

```
RMSql protocol -pt http
```

The following command will change the server SQLserver protocol to HTTP.

```
RMSql protocol -pt http -s SQLserver
```

COMMAND

`RMSql time`

SYNTAX

`RMSql time [<-prt promotetime>] [<-bkt backuptime]`

DESCRIPTION

After Recovery Manager for SQL is installed, by default the wait time for snap volume restore and the backup is 30 seconds and 5 minutes respectively. You can use this command to change this value as needed.

PARAMETERS

`-prt <promotetime>` -Promote Time to query the status (default is 30 seconds).

`-bkt <backuptime>` - Backup Time to query the status (default is 5 minutes.)

EXAMPLES

The following command will query the time value for each setting.

```
RMSql time
```

The following command will change the promote time from 30 to 50 seconds.

```
RMSql time -prt 50
```

The following command will change the backup time from 5 to 7 minutes.

```
RMSql time -bkt 7
```



NOTE: In most cases, there will not be a need to change these values unless the backup and restore are taking an abnormally long time to complete.



NOTE: In Recovery Manager version 3.0, both the `timeconfig` and `time` commands are supported. The `time` command is deprecated and will not be available in the next release. Please use the `timeconfig` command.

COMMAND

`RMSql timeconfig`

SYNTAX

`RMSql timeconfig [<-prt promotetime>] [<-bkt backuptime]`

DESCRIPTION

After Recovery Manager for SQL is installed, by default the wait time for snap volume restore and the backup is 30 seconds and 5 minutes respectively. You can use this command to change this value as needed.

PARAMETERS

`-prt <promotetime>` -Promote Time to query the status (default is 30 seconds).

`-bkt <backuptime>` - Backup Time to query the status (default is 5 minutes.)

EXAMPLES

The following command will query the time value for each setting.

```
RMSql timeconfig
```

The following command will change the promote time from 30 to 50 seconds.

```
RMSql timeconfig -prt 50
```

The following command will change the backup time from 5 to 7 minutes.

```
RMSql timeconfig -bkt 7
```



NOTE: In most cases, there will not be a need to change these values unless the backup and restore are taking an abnormally long time to complete.

7.3 Running Recovery Manager as a Scheduled Job

In addition to creating virtual copies using the Command Line Interface, you can also schedule virtual copies to be created for a specific database using Windows Scheduled Tasks.

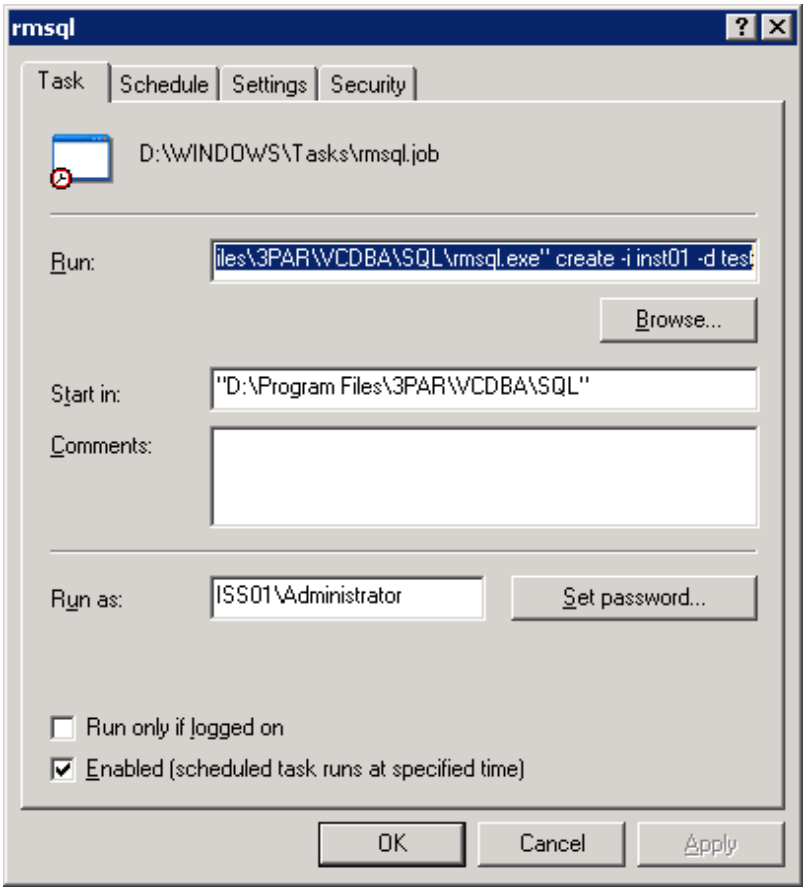
To create virtual copies using Windows Scheduled Tasks:

1. Click **Start** > **Programs** > **Accessories**. The click **System Tools** > **Scheduled Tasks**.
2. Double-click **Add Scheduled Tasks**.
3. Click **Browse** and navigate to the Recovery Manager's install location.

   The default location is `C:\Program Files\3PAR\VCDBA\SQL`
4. Click `RMSql.exe`.
5. Select the frequency of this task. Click **Next** to continue.
6. Select time and date that this task will start and click **Next**.
7. Enter the user name, password, and password confirmation, and then click **Next**.
8. Select the Open advanced properties... checkbox and click **Finish** to access advanced setup for the scheduled task.
9. On the **Task** tab, at the end of the path displayed in the **Run** field, add: `RMSql.exe create –s <SQL Server> –i <SQL Instance> -d <SQL database>`.
   If the name has spaces, use double-quotes ("") to surround the name. See the `RMSql CREATE` command for detailed information.

For example:

```
"C:\Program Files\3par\VCDBA\SQL\RMSql.exe" create –i inst01
–d test -s SQLserver
```



10 On the Schedule tab, further refine the scheduled task by selecting the frequency and start time of the task. Click **Apply** and **OK** when finished.

- Virtual copies are created as you have scheduled. Depending on the policy set with the Recovery Manager GUI or CLI command, older virtual copies on the system might automatically be removed if the number of virtual copies exceeds your specified maximum threshold (see *Setting the Recovery Manager Policy Configuration* on page 6.9).

8 Recovery for Microsoft SQL Server

In this chapter

8.1 Recovery 8.1

8.1.1 Point-In-Time Recovery 8.2

8.1.2 Point-Of-Failure Recovery 8.3

8.1.3 SQL Instance Level File Copy Restore in an MSCS Environment 8.5

8.1 Recovery

The SQL Server offers three disaster recovery models:

Simple Recovery Model

Full Recovery Model

Bulk-logged Recovery Model

With simple recovery it is not possible to restore a database to a specific point in time, you can only restore a database to the exact time when a backup occurred. Therefore, any data modifications made between the time of the most recent backup and the time of the failure are lost. This recovery model uses circular logging to minimize the amount of space used for transaction logs.

The full recovery model allows you to restore a database to a specific point in time. For example, you can restore a database to a specific time just before a failure occurs. With the full recovery model, no data is lost because the transaction logs are saved until the failure occurs.

The bulk-logged recovery model is used for high-performance operations and involves bulk copying and bulk data modification.

The Recovery Manager supports both recovery from Tape and from Virtual Copy directly. Point-in-time recovery and point-of-failure recovery are only supported on recovery from Virtual Copy.

The Recovery Manager does the restore through Volume Shadow Copy Service (VSS) framework. Because the VSS framework takes the database offline and allows the requestor to do a restore operation, placing the database in a single-user mode would no longer be necessary.



WARNING: When taking the database offline, the SQL server writes a checkpoint record in the database ID 9. When the log file is full, the database cannot be taken offline. You need to backup the transaction log for the database to free up some log space or you can detach the database and then do the restore.

8.1.1 Point-In-Time Recovery

A point-in-time recovery can be achieved only when a full backup of the SQL database has been performed. It can be useful for recovering from logical errors. For example, if you accidentally dropped a table, a virtual copy backup can return the database to a consistent, known prior state (the state at the time of the last backup). Any changes made after that last backup would need to be redone.

There are two methods for performing a point-in-time restore: file copy restore and snap volume restore. File copy restore operations take longer to finish than snap volume restore operation.

When the `-restore` command is executed, Recovery Manager performs the following steps:

1. Mount the virtual copy backup.
2. Take the database offline.
3. Restore the virtual copy backup to the database.
4. Bring the database back online.

Now, the database is restored to the stage of the last backup. The logical errors after the last point-in-time backup are now gone.

To perform a point-in-time recovery:

1. Find the timestamp that has a consistent full backup, for example `010106000000`.
2. Find the name of database for example, `northwind`.
3. Find the name of the SQL instance containing the database, for example `SQLInstan01`.
4. Find the SQL server name, for example `SQLserver01`.
5. To do a file copy restore, use the following command as an example:

```
RMSql restore -s SQLserver01 -i SQLInstan01 -d
northwind -t 010106000000
```

6. To do a volume restore, use the `-snap` option. Use the following command as an example:

```
RMSql restore -s SQLserver01 -snap -i SQLInstan01 -d
northwind -t 010106000000
```



NOTE: Volume restore brings the entire volume back to the point-in-time. If you have not allocated databases into different volumes, snap volume restore is not supported.

8.1.2 Point-Of-Failure Recovery

A point-of-failure recovery can be achieved when a full backup of the SQL database has been performed and has been followed by a log backup. This type of recovery would be useful for customers who cannot afford to lose any data in the event of a system failure. As long as the transaction logs were not damaged, this function can be used to recover the database up to the point of the failure.



NOTE: Point-of-failure recovery only supports the database level restore and the file copy restore.

Recovery Manager provides the `-manrollforward` option to perform the point of failure recovery. The log backup needs to be taken by the DBA, using the SQL backup utility before this option can be exercised. Applying the transactions log to snapshot backups, allows a database restoration from a snapshot backup to be rolled-forward to an arbitrary point in time using the STOPAT option. To roll forward the transaction log, the DBA can use either the SQL Server Management Studio, or a command line tool such as OSQL. For more information on how to roll forward with transaction log, see http://msdn.microsoft.com/en-us/library/ms189596.aspx.

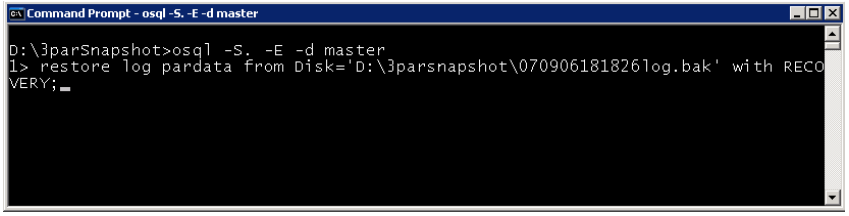


Figure 8-1. OSQL Command Line Tool

NOTE: A transaction log backup is a copy of all the log records that were written to the live transaction log after the last full database backup or the last transaction log backup. By missing any transaction log backups within the sequence would cause the full recovery to fail.

Using the `-manrollforward` command, the following steps will be performed by the SQL Recovery Manager:

1. Mount the virtual copy backup.
2. Put the database into non-operation mode.
3. Take the database offline.
4. Restore the virtual copy backup to the database.
5. Bring the database back online.

   At this stage, the database is at the `no_recovery` mode and the DBA can now apply the log backups that took place after the virtual copy backup was taken.

To perform a point-of-failure recovery:

1. Find the timestamp that has a consistent full backup, (such as *010106000000*).

2 Find the server name (such as *SQLserver01*).

3 Find the SQL instance name (such as *SQLInstan01*).

4 Find the database name (such as *northwind*).

At the command prompt, type:

```
RMSql.exe restore -s SQLserver01 -i SQLInstan01 -d northwind -manrollforward  -
t 010106000000
```

After the command has finished, the DBA can apply the logs.

For example:

RESTORE log northwind from Disk='G:\log1.bak' with NO_recovery;

RESTORE log northwind from Disk='G:\log2.bak' with NO_recovery;

RESTORE log northwind from Disk='G:\log3.bak' with recovery; -stopat "Jun 28, 2006 5:44 PM"

8.1.3 SQL Instance Level File Copy Restore in an MSCS Environment

To perform a SQL instance level file copy restore in an MSCS environment:

1 From **Failover Cluster Management** on Windows 2008 or Cluster Administrator or Windows 2003:

a Offline **SQL Server Agent** resource.

b Offline **SQL Instance Service** resource.

2 Run the `rmsql restore` command with an instance level virtual copy.

3 From **Failover Cluster Management** on Windows 2008 or Cluster Administrator or Windows 2003:

a Online **SQL Server Agent** resource.

b Online **SQL Instance Service** resource.

9 Remote Copy for Recovery Manager

In this chapter

9.1 Purpose of Remote Copy 9.1

9.2 How it works 9.1

9.3 How to Setup the Remote Copy Feature 9.3

9.1 Purpose of Remote Copy

Remote Copy can be used to safeguard against disasters that may occur at one location by keeping a copy of valuable data at another location. For example, if there was a fire or earthquake at a particular location, all of the data stored there could be lost unless a backup copy is maintained somewhere where the data can be recovered. The Remote Copy feature supported through the InServ can prevent this kind of disastrous scenario. When using Remote Copy, two InServs are utilized: a primary which is used for production, and secondary that is used for backup purposes. When disaster strikes in one location, there is always a backup at the secondary site.

9.2 How it works

When users want to synchronize the data from the production site to secondary site, Recovery Manager for Microsoft SQL will leverage the VSS framework to freeze the I/O at the

application level, and then allows the VSS provider to take a snapshot of the database and log volumes. Once the snapshot is taken, Remote Copy performs a synchronization on the data changes to the target volumes at the secondary site. After the synchronization is complete, the target server connected to the Inserv at the secondary site, creates a Virtual copy on the remote copy target volumes, and saves the information in its repository at the secondary site.



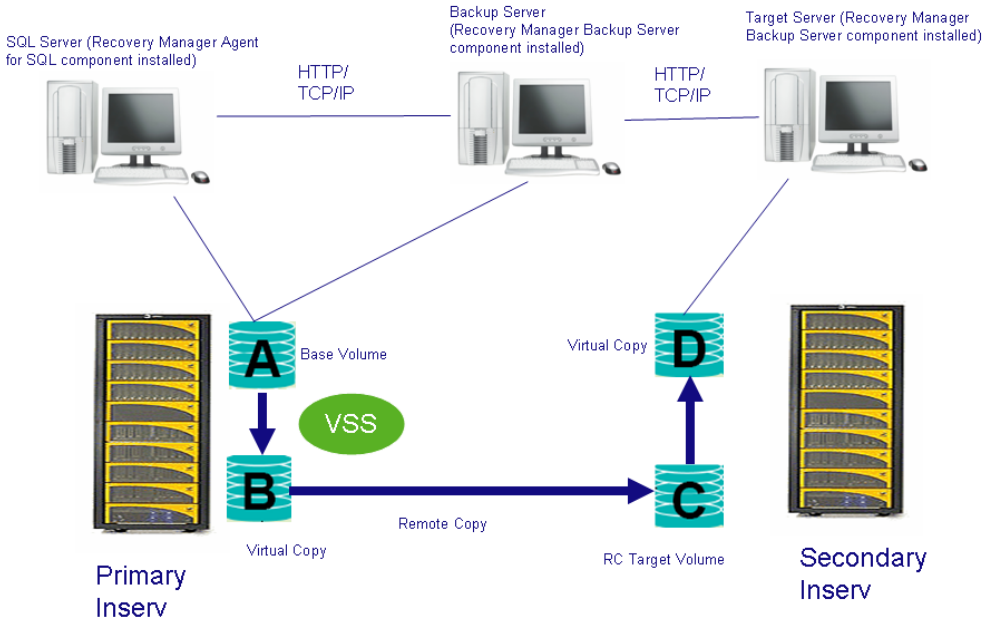
NOTE: The Target Server serves as the backup server at the secondary site. After synchronization has completed, a virtual copy will be created and managed by the Target Server at the secondary site.



NOTE: Virtual copy created on the primary site for Remote Copy purpose will be removed after synchronization has completed.



NOTE: In order for Recovery Manager GUI to manage virtual copies form the remote target server, the exact SQL server, instance, and database must be created from the remote site in order to register the SQL server successfully from Recovery Manager GUI.



9.3 How to Setup the Remote Copy Feature

In order to utilize the Remote Copy feature, a remote copy environment set up needs to be completed.

NOTE: See the 3PAR Remote Copy User's Guide for details on how to set up Remote Copy.

Once you have Remote Copy set up between two Inservs, you should create a remote copy group for testing before going to the next step.

9.3.1 Set Up Remote Copy Group for Databases

Follow these steps to set up Remote Copy for a given database:

1 Identify the volumes that contain the database and log files to be copied.

2 Create one remote copy group for one database.

3 Add the volumes belonging to the database files and log files into the remote copy group with remote copy target volumes.

4 Synchronize the remote copy group.

Every database's log and database file should be added in one remote copy group. Two or more databases in one remote copy group are not supported. DB files from the same database and log files from the same database located in the different Remote copy group are not supported.

If the database is not in the remote copy group the remote copy feature cannot be used.

A Troubleshooting

Trouble-shooting

This appendix is designed to be used like a Frequently Asked Questions section. It will help find solutions to some of the common issues that may arise when running the Recovery Manager for Microsoft SQL.

Problem:
Error message "VSS_E_PROVIDER_VETO" appears. Look at the application event log and VSS provider log from the production server, and if there is a "Maximum of 28 writable virtual copies per volume is allowed", this may indicate that your snapshot capacity is used up.

Suggestion:
This error usually occurs during the mounting operation. Unmount some of the virtual copies to continue the operation.

Problem:
Recovery Manager validates configuration during operation, the validation might fail without proper notification. During Recovery Manager InServ checking, it exits without any messages.

Suggestion:
Restart the RMSql agent service on the backup and production servers.

Problem:
When backing up a virtual copy using NetBackup through the Recovery Manager GUI, if a directory is specified for backup, the directory must be entered in exactly the same case as it resides on the file system. If this is not done then a restore of this backup will fail. This is a limitation of Veritas NetBackup.

Suggestion:
When backing up a directory, specify the directory in exactly the same case as it exists on the file system.

Problem:
A virtual copy creation fails with error "Error-VSS_E_HOLD_ WRITES_ TIMEOUT" when creating a virtual copy on a SAN boot disk, the in-box system writer which enumerates all operating system and driver binaries will be included as well. The error mostly occurs because of the I/O cannot hold on the system writer.

Suggestion:
Database located on the san boot disk is not recommended. All the important databases should be located on the separate san disks instead.

Problem:
As data blocks are copied per the copy-on-write technology, the snapshot space of a virtual volume may become 100% full if it has been statistically allocated as a percentage of the virtual volume, or if there is no additional available space for the CPG to auto-grow. In the event that the snapshot space of a virtual volume becomes 100% full, the snapshots for the virtual volume may become invalid and subject to the configured Snapshot Policy.

Suggestion:
Allocate more snapshot space when creating virtual volumes during base volume creation or use CPGs to manage data generated by copy-on-write technology.

Problem:
Recovery Manager always fails to issue command from the user interface and command line right after the installation.

Suggestion:
If the software is installed through remote desktop session, the newly added environment variables will not become effective until user logs off the session. If the software is installed directly from system console, these variables should be available immediately.

Problem:
NetBackup takes a substantial amount of time to complete a backup process.

Suggestion:
Make sure you have the latest maintenance package installed for your NetBackup application.

Index

A

advisories 1.4
audience for this guide 1.1

C

cautions, defined 1.4
configuring
  host computer and controller nodes 5.4
  Recovery Manager 5.2
  the database 5.1
controller nodes and host computer,
    configuring 5.4
conventions
  typographical 1.3

D

database
  configuring 5.1
database files supported 5.1
database, configuring 5.1
documents, related 1.2

G

GUI
  menu bar 6.3

H

host computer and controller nodes,
    configuring 5.4

I

installation location of Recovery Manager 5.2
installation of Recovery Manager, verifying 3.6
installing Recovery Manager 3.1

M

menu bar 6.3
Microsoft Volume Shadow Copy Service (VSS) 2.1
Microsoft VSS 2.1

N

notes, defined 1.4

O

organization of this guide 1.2

R

Recovery Manager
  commands 7.3
  default installation location 5.2
  installation 3.1
  verifying installation 3.6
related documents 1.2
revision history RH.1

S

search path 5.2
server maintenance

required tools A.1
setting virtual copy limit 6.9

T

Troubleshooting
troubleshooting A.1
typographical conventions 1.3

U

understanding the registry, registry 3.7

V

verifying installation of Recovery Manager 3.6
virtual copy
as a scheduled task 7.30
VSS 2.1

W

warnings, defined 1.4
Windows scheduled tasks 7.30

Revision History

Release Level	Revision Summary
320-200108 Rev A December 2008	First release of this manual to support product release 3.0.0.