# VERITAS NetBackup™ 6.0 for Microsoft SQL Server

## System Administrator’s Guide

for Windows

N152678

September 2005

## Disclaimer

The information contained in this publication is subject to change without notice. VERITAS Software Corporation makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

## VERITAS Legal Notice

VERITAS Software Corporation
350 Ellis Street
Mountain View, CA 94043
USA
Phone 650-527-8000
Fax 650-527-2908
www.veritas.com

## Third-Party Copyrights

For a list of third-party copyrights, see the *NetBackup Release Notes* appendix.

# Contents

# Preface

This guide describes how to install, configure and use the NetBackup for Microsoft® SQL Server extension on a Windows platform.

This document is the same as `NetBackup_AdminGuide_MS-SQL_Win.pdf` distributed with the NetBackup for SQL Server software.

This guide is intended for the:

- SQL Server system administrator responsible for configuring and using SQL Server.
- NetBackup system administrator responsible for configuring NetBackup.

A system administrator is a person with system administrator privileges and responsibilities.

This guide assumes:

- A basic understanding of Windows system administration.
- A working understanding of the NetBackup for Windows client software.
- A working understanding of the NetBackup server and client software.
- A familiarity with the information covered in the following NetBackup manuals:
    - *NetBackup System Administrator's Guide for UNIX*, Volumes I and II, or *NetBackup System Administrator's Guide for Windows*, Volumes I and II.
    - *NetBackup Troubleshooting Guide.*
- A thorough understanding of the SQL Server administration.

# Getting Help

You can find answers to questions and get help from the NetBackup documentation and from the VERITAS technical support web site.

## Finding NetBackup Documentation

A list of the entire NetBackup documentation set appears as an appendix in the *NetBackup Release Notes*. All NetBackup documents are included in PDF format on the NetBackup Documentation CD.

For definitions of NetBackup terms, consult the online glossary.

### ▼ To access the NetBackup online glossary

1. In the NetBackup Administration Console, click **Help** > **Help Topics**.
2. Click the **Contents** tab.
3. Click **Glossary of NetBackup Terms**.

Use the scroll function to navigate through the glossary.

## Accessing the VERITAS Technical Support Web Site

The address for the VERITAS Technical Support Web site is http://support.veritas.com.

The VERITAS Support Web site lets you do any of the following:

- Obtain updated information about NetBackup, including system requirements, supported platforms, and supported peripherals
- Contact the VERITAS Technical Support staff and post questions to them
- Get the latest patches, upgrades, and utilities
- View the NetBackup Frequently Asked Questions (FAQ) page
- Search the knowledge base for answers to technical support questions
- Receive automatic notice of product updates
- Find out about NetBackup training
- Read current white papers related to NetBackup

From http://support.veritas.com, you can complete various tasks to obtain specific types of support for NetBackup:

1. Subscribe to the VERITAS Email notification service to be informed of software alerts, newly published documentation, Beta programs, and other services.

   a. From the main http://support.veritas.com page, select a product family and a product.

   b. Under Support Resources, click **Email Notifications**.

      Your customer profile ensures you receive the latest VERITAS technical information pertaining to your specific interests.

2. Locate the telephone support directory at http://support.veritas.com by clicking the **Phone Support** icon. A page appears that contains VERITAS support numbers from around the world.

   **Note** Telephone support for NetBackup is only available with a valid support contract. To contact VERITAS for technical support, dial the appropriate phone number listed on the Technical Support Guide included in the product box and have your product license information ready for quick navigation to the proper support group.

3. Contact technical support using e-mail.

   a. From the main http://support.veritas.com page, click the **E-mail Support** icon.

      A wizard guides you to do the following:

      - Select a language of your preference
      - Select a product and a platform
      - Provide additional contact and product information, and your message
      - Associate your message with an existing technical support case

   b. After providing the required information, click **Send Message**.

## Contacting VERITAS Licensing

For license information, you can contact us as follows:

- Call 1-800-634-4747 and select option 3
- Fax questions to 1-650-527-0952
- In the Americas, send e-mail to amercustomercare@veritas.com.

  In the Asia and Pacific areas, send email to apaccustomercare@veritas.com.

  In all other areas, send email to internationallicense@veritas.com.

# Accessibility Features

NetBackup contains features that make the user interface easier to use by people who are visually impaired and by people who have limited dexterity. Accessibility features include:

- Support for assistive technologies such as screen readers and voice input (Windows servers only)
- Support for keyboard (mouseless) navigation using accelerator keys and mnemonic keys

For more information, see the *NetBackup Installation Guide*.

# Comment on the Documentation

Let us know what you like and dislike about the documentation. Were you able to find the information you needed quickly? Was the information clearly presented? You can report errors and omissions or tell us what you would find useful in future versions of our manuals and online help.

Please include the following information with your comment:

- The title and product version of the manual on which you are commenting
- The topic (if relevant) on which you are commenting
- Your comment
- Your name

Email your comment to NBDocs@veritas.com.

Please only use this address to comment on product documentation. See “Getting Help” in this preface for information on how to contact Technical Support about our software.

We appreciate your feedback.

# Introduction to NetBackup for SQL Server 1

NetBackup for SQL Server extends the capabilities of NetBackup for Windows to include backing up and restoring SQL Server databases. These capabilities are provided for a Windows client using either a UNIX or Windows NetBackup master server.

NetBackup for SQL Server includes a client-based graphical user interface (GUI) program to perform various activities on SQL Server. These activities include:

- Setting options for NetBackup for SQL Server operations.
- Backing up, restoring and copying databases and database components, which include transaction logs, differentials, files and filegroups.
- Starting NetBackup for SQL Server operations from batch files which you have created.
- Monitoring NetBackup for SQL Server operations.

Microsoft SQL Server will be referred to as SQL Server. NetBackup for Microsoft SQL Server will be referred to as NetBackup for SQL Server.

# Features

This section describes the main features of NetBackup for SQL Server.

## NetBackup Operations

- Full integration with the NetBackup master server and Media Manager.
- Stream-based backup and restore of SQL Server objects to tape or disk using SQL Server's high-speed virtual device interface.
- Snapshot-based backup and restore of SQL Server objects using NetBackup Advanced Client methods.
- Backup and recovery of databases, differentials, files, filegroups, and transaction logs.
- Browse capability for SQL Server objects on the local and remote nodes.
- Support for redirection of SQL Server restores to different locations.
- Support for multiple SQL Server instances.
- Client operation monitoring through the NetBackup Client Job Monitor. Server monitoring is also available through the NetBackup master.
- Performance tuning through user control of backup stripes, transfer size, and buffer usage.
- Job launch is supported through the following options:
    - Immediate launch through the NetBackup Database Extension GUI
    - Scheduled backup in a backup policy
    - Command line launch
- Support for Microsoft Cluster Server and Veritas Cluster Server.
- Recovery of Microsoft SQL images backed up with Backup Exec, through the Backup, Archive, and Restore (NetBackup Client) interface.
- Support for SQL Server 2005 enhancements, including page-level validation and partial database backup and restore.
- Partial recovery
- Optimization for read-only filegroups
- Verify-only restore
- Page verification during backup and restore
- Page-level restore

- Backup and restore of the full text search catalog
- Extended object information in the SQL Server catalog
- Progress statistics during backup and restore
- Copy-only backup

## Graphical User Interface

- GUI capability for browsing:
  - SQL Server databases
  - SQL Server database files and filegroups
- GUI assistance for staging a complete database recovery from backup images created for databases, filegroups, files, database differentials, and transaction logs, as well as read-write filegroup backups and partial database backups.
- GUI assistance for page level restore.
- Assistance for minimizing backup volume by making use of the read-only versus read-write properties of SQL Server data.
- GUI assistance for restoring database objects backed up on one SQL Server client to another SQL Server client.
- GUI assistance for creating and saving a backup script as an alternative to performing an immediate job launch.
- Properties display for SQL Server databases, filegroups, and files.
- Properties display for NetBackup for SQL Server backup images.
- Online help provided through the NetBackup Database Extension GUI.

# Installation Requirements and Registering a License Key 2

This chapter describes how to perform the following tasks:

- Verifying the installation prerequisites for NetBackup for SQL Server. Perform this task before enabling this agent.

  The NetBackup for SQL Server software is installed when you install NetBackup, so verify these prerequisites before enabling NetBackup for SQL Server.

- Registering the license key for NetBackup for SQL Server.

For information on installing NetBackup for SQL Server in a cluster, refer to "Using NetBackup for SQL Server with Clustering Solutions" on page 121.

# Verifying the Installation Prerequisites

Before enabling NetBackup for SQL Server, verify that you are installing the agent on a supported operating system or platform and that requirements are met for the NetBackup software and the database agent. These requirements apply for remote and local installations.

## Operating System and Platform Compatibility

Verify that NetBackup for SQL Server will be installed on an operating system or platform supported by NetBackup. A compatibility list for database agents is available on the Technical Support web site.

▼ **To verify compatibility**

1. Go to the Technical Support web page: http://support.veritas.com.

2. From the **Select Product Family** list, click **NetBackup Products**.

3. From the **Select Product** list in the right pane, click **NetBackup Enterprise Server**.

4. Under **Support Resources**, click **Compatibility and Reference**.

5. In the list of documents, click **NetBackup Enterprise Server (tm)/ Server** *x.x* **Database Agent Compatibility (updated** *date***)**.

   For *x.x*, look for the current release. For *date*, look for the most recent date.

   For information on supported cluster environments for NetBackup for SQL Server, see **NetBackup (tm)** *x.x* **Cluster Compatibility (updated** *date***)**.

6. Click on the link for the PDF document, which is a downloadable file that enables you to view the supported database spreadsheet for this release.

7. Read the document and verify that the software in your environment is compatible with the NetBackup and NetBackup for SQL Server software.

## NetBackup Software

Verify that the following requirements are met for the NetBackup server and client software. The NetBackup for Windows client relies on a NetBackup Master Server and a NetBackup Media Server, either of which may exist on the same or a different host than the NetBackup client.

❏ The NetBackup server software is installed and operational on the NetBackup server.

NetBackup for SQL Server software is installed along with the NetBackup for Windows client on the server. The NetBackup server platform can be any of those that NetBackup supports. For installation information, see the *NetBackup Installation Guide*.

❏ If the SQL client is on a different host than the master or media server, then the NetBackup client must installed on that host. Installing the NetBackup Client also installs the NetBackup for SQL Server agent.

The version level of the client must be the same as or older (e.g., 6.0 or earlier) than the version of any NetBackup server in its Servers list.

❏ Make sure that you have configured backup media in a Media Manager or disk storage unit. The number of media volumes required depends on the devices used, the sizes of the databases that you are backing up, the amount of data that you are archiving, the size of your backups, and the frequency of backups or archives. For information on using Media Manager, see the *NetBackup Media Manager System Administrator's Guide*.

## Database Software

Verify that the follow requirements are met for the database software.

❏ The SQL Server vendor software is installed and operational on each client.

❏ SQL Server is installed on Windows 2000 or later.

❏ NetBackup supports SQL Server 7.0 with Service Pack 1 or higher, SQL Server 2000 with Service Pack 3 or higher, or SQL Server 2005.

# Registering the License Key

NetBackup for SQL Server is installed with client software. You can use this agent from a NetBackup server or a client. To use this agent, register a valid license key for it on the master or media server.

### ▼ To register a license key

1. On the master or media server, open the NetBackup Administration Console.
2. Choose **Help** > **License Keys**.
3. Click the **New** button.
4. Type in the license key and click **Add**.

   For information on adding license keys, see the *NetBackup System Administrator's Guide, Volume I.*

# Configuration 3

This chapter describes how to configure NetBackup for SQL Server. The following sections in this chapter describe each of these steps in detail.

- "Verifying Installed Components" on page 10
- "Configuring the Maximum Jobs Per Client" on page 12
- "Configuring Backup Policies" on page 13
- "Testing Configuration Settings" on page 25
- "Configuring Multi-Stream Operations" on page 26
- "Configuring for a Multi-Interface Network Connection" on page 28
- "SQL Server Privileges" on page 31
- "Performance Factors" on page 34
- "Backing Up SQL Server in an Environment with Log Shipping" on page 38

## User Interface Terminology Notes

You can perform many of the configuration steps in this chapter from the NetBackup Administration Console on the master server. Depending on your master server's platform, the console is available in one or two forms. NetBackup supports a Java interface for both Windows and UNIX master servers. In addition, NetBackup supports a Windows interface for Windows master servers.

The Java and Windows interfaces are nearly identical, but when there are interface differences in the configuration procedures, this manual uses the following headings to identify the interface being described:

*From the Windows interface:*

*From the Java interface:*

# Verifying Installed Components

**Note** *`install_path`* refers to the directory where you installed the NetBackup software. By default, this directory is `C:\Program Files\VERITAS\`.

The NetBackup SQL Server agent is installed automatically when you perform a NetBackup server or client installation. During the installation, the following actions for the NetBackup for SQL Server Agent were taken.

- An icon was added to the NetBackup Program Group

  **Veritas NetBackup** > **NetBackup Agents** > **NetBackup MS SQL Client**

- The following new directories were created, if they did not already exist on your Windows host:

  *`install_path`*`\NetBackup\DbExt\Mssql`

  The default directory for batch files and miscellaneous items pertaining to installation.

  *`install_path`*`\NetBackup\logs\user_ops\Mssql\jobs`

  The directory containing NetBackup master server logging files for each operation that you perform.

  *`install_path`*`\NetBackup\logs\user_ops\Mssql\logs`

  The directory containing NetBackup client logging files for each operation that you perform.

- Two sample *batch* files were placed in *`install_path`*`\NetBackup\DbExt\Mssql`:

  `bkup.bch`

  `rest.bch`

  Batch files are described in "Using Batch Files" on page 100.

- The following lists binaries created for NetBackup for SQL Server. The path for these components is *`install_path`*`\NetBackup\bin\`.

| | |
|---|---|
| `DBbackmain.dll` | NetBackup client DLL which backs up and restores SQL Server databases. |
| `dbbackup.exe` | Graphical user interface to `dbbackmain`. |

`dbbackex.exe` A command-line interface to `dbbackmain`. It is also called internally by the NetBackup client (through `bphdb`), and by NetBackup for SQL Server for performing NetBackup scheduling operations.

`xbsa.dll` Common NetBackup database client functions

`dbbackup`*`XX`*`.dll` Resource localization DLL, where *`XX`* indicates the language. For example, `dbbackupEN.dll` is the resource localization DLL for English.

# Configuring the Maximum Jobs Per Client

### ▼ To configure the Maximum jobs per client

1. In the left pane of the NetBackup Administration Console, expand **Host Properties**.

2. Select **Master Server**.

3. In the right pane, double-click on the server icon.

   The Master Server Properties dialog box displays.

4. In the Master Server Properties dialog box, click **Global Attributes**.

5. Change the **Maximum jobs per client** value to 99.

   The **Maximum jobs per client** specifies the maximum number of concurrent backups allowed per client. The default is 1. You can use the following formula to calculate a smaller value:

   Maximum jobs per client = `number_of_streams` X `number_of_policies`

*`number_of_streams`* The number of backup streams between the database server and NetBackup. If striping is not used, each separate stream starts a new backup job on the client. If striping is used, each new job uses one stream per stripe.

*`number_of_policies`* The number of policies of any type that can back up this client at the same time. This number can be greater than one. For example, a client can be in two policies in order to back up two different databases. These backup windows can overlap.

# Configuring Backup Policies

A NetBackup policy defines the backup criteria for a specific group of one or more clients. These criteria include:

- Storage unit and media to use
- Policy attributes
- Backup schedules
- Clients to be backed up
- The batch file files to be executed on the clients

To use NetBackup for SQL Server, you need to define at least one MS-SQL-Server policy with the appropriate schedules. A configuration can have a single policy that includes all clients, or there can be many policies, some of which include only one client.

Most requirements for MS-SQL-Server policies are the same as for file system backups. In addition to the attributes described here, there are other attributes for a policy to consider. For configuration instructions and information on all the attributes available, see the *NetBackup System Administrator's Guide, Volume I*.

## Adding a New Policy

### ▼ To add a new policy

**Note** If you are going to perform multi-streamed backups and restores, see "Configuring Multi-Stream Operations" on page 26.

1. Log on to the master server as administrator (Windows) or root (UNIX).

2. Start the NetBackup Administration Console.

3. If your site has more than one master server, choose the one on which you want to add the policy.

4. *From the Windows interface*: In the left pane, right-click **Policies** and choose **New Policy**.

   *From the Java interface*: In the left pane, click **Policies**. In the **All Policies** pane, right-click the master server, and click **New**.

   The **Add a New Policy** dialog box displays.

5. In the **Policy name** field, type a unique name for the new policy.

6. Click **OK**.

   A dialog box displays in which you can specify the general attributes for the policy.

7. From the **Policy Type** box, select the **MS-SQL-Server** policy type.

**Note** The MS-SQL-Server policy type does not appear in the drop-down list unless your master server has a license key for NetBackup for SQL Server.

8. Complete the entries on the **Attributes** tab. For more information, see "Description of Attributes" on page 15.

9. Add other policy information.

   ◆ To add schedules, see "Adding Schedules" on page 15.

   ◆ To add clients, see "Adding Clients" on page 21.

   ◆ To add batch files to the Backup Selections list, see "Adding Backup Selections" on page 22.

10. When you have added all the schedules, clients, and Backup Selections you need, click **OK**. The new policy is created.

### Description of Attributes

With a few exceptions, NetBackup manages a database backup like a file system backup. The following table shows the policy attributes that are different for SQL Server backups. This information is used when you are adding a new policy.

Other policy attributes vary according to your specific backup strategy and system configuration. For more information on policy attributes, see the *NetBackup System Administrator's Guide, Volume I.*

Description of Policy Attributes

| Attribute | Description |
|---|---|
| **Policy type** | Determines the types of clients that can be in the policy and, in some cases, the types of backups that NetBackup can perform on those clients. To use NetBackup for SQL Server, you must define at least one policy of type MS-SQL-Server. |
| **Keyword phrase** | Although you can create a keyword phrase for MS-SQL-Server policies, it is not used by Netbackup for SQL Server. |
| **Advanced Client** | See "Using NetBackup for SQL Server with Advanced Client" for information on configuring policies for advanced backup methods. |

## Adding Schedules

Each policy has its own set of schedules. These schedules initiate automatic backups and specify when a user can initate operations.

A SQL Server backup requires an Application Backup schedule, which is created automatically when you create a SQL Server policy. The Application Backup schedule manages the backup operation. You also need one or more automatic backup schedules if you plan to have NetBackup perform automatic scheduled backups or if you are using Advanced Client features.

### ▼ To configure an Application Backup schedule

1. In the Policy dialog, click the **Schedules** tab.

   To access the Policy dialog, double-click the policy name in the Policies list in the NetBackup Administration Console.

2. Double-click on the schedule named **Default-Application-Backup**.

   A dialog box displays. The title bar shows the name of the policy to which you are adding the schedule.

   All SQL Server backup operations are performed through NetBackup for SQL Server using an Application Backup schedule. This includes backups that start automatically.

   Configure an Application Backup schedule for each MS-SQL-Server policy you create. If you do not do this, you cannot perform a backup. To help satisfy this requirement, an Application Backup schedule named **Default-Application-Backup** is automatically created when you configure a new MS-SQL-Server policy.

3. Specify the other properties for the schedule as explained in "Schedule Properties" on page 18.

   The backup window for an Application Backup schedule must encompass the time period during which all NetBackup for SQL Server jobs, scheduled and unscheduled, can occur. This is necessary because the Application Backup schedule starts processes that are required for all NetBackup for SQL Server backups, including those started automatically.

   For example, assume the following:

   - Users perform MS-SQL-Server backup operations during business hours, 0800 to 1300.
   - Automatic backups that use this policy commence between 1800 and 2200.

   The Application Backup schedule must have a start time of 0800 and a duration of 14 hours.

Example Settings for an Application Backup schedule

| **Type of Backup** | **Schedule settings** | **Description** | **Settings** |
|---|---|---|---|
| Application Backup | Retention | The length of time the backup images are retained in the NetBackup catalog for restore. | 2 weeks |
| | Backup Window | The time during which a NetBackup operation can be initiated. | Sunday through Saturday<br>00:00:00 - 24:00:00 |

**Tip** Set the time period for the Application Backup schedule for 24 hours per day, seven days per week. This ensures that your NetBackup for SQL Server operations are never locked out due to the Application Backup schedule.

### ▼ To configure an automatic backup schedule

1. On the Policy dialog, click the **Schedules** tab.

2. Click **New**.

   A dialog box displays. The title bar shows the name of the policy to which you are adding the schedules.

3. Specify a unique name for the schedule.

4. Select the **Type of backup**.

   For information on the types of backups available for this policy, see "Types of Backup Schedules" on page 20.

**Tip** If you put multiple batch files in the same MS-SQL-Server policy, they will execute during each Automatic Backup session for that MS-SQL-Server policy. If you have a variety of SQL Server backup operations that you wish to have executed on different schedules, then consider creating multiple MS-SQL-Server policies with differing Automatic Backup schedules; and then assign each batch file to the policy that uses the desired Automatic Backup schedule.

5. Specify the other properties for the schedule as explained in Schedule Properties, which follows this procedure.

**Tip** For policies used for SQL Server 2005 read-only filegroups, consider creating a schedule with a retention set to infinity. This may enable you to avoid redundant backups. See "Reducing Backup Size and Time By Using Read-Only Filegroups (SQL Server 2005)" on page 45.

The following table shows example settings for an automatic backup schedule:

Example Settings for an Automatic Backup Schedule

| Type of Backup | Schedule settings | Description | Settings |
|---|---|---|---|
| Automatic Backup | Retention | The length of time to store the record of a backup, which NetBackup uses to determine if the schedule needs to be run. | 2 weeks |
| | Frequency | Frequency determines how often a backup should be performed | every week |
| | Backup Window | The time during which a NetBackup operation can be initiated. | Sunday, 18:00:00 - 22:00:00 |

**6.** If this is the last schedule, click **OK**. To add other schedules, repeat step 1 through step 6.

## Schedule Properties

Some of the schedule properties have a different meaning for database backups than for a regular file system backup. The following table explains the schedule properties:

Description of Schedule Properties

| Property | Description |
|---|---|
| **Type of backup** | Specifies the type of backup that this schedule controls. The selection list shows only the backup types that apply to the policy you are configuring.<br>For more information, see "Types of Backup Schedules" on page 20. |
| **Frequency** | This setting is used only for scheduled backups and not for user-directed backups. **Frequency** specifies the period of time that can elapse until the next backup or archive operation begins on this schedule. For example, if the frequency is seven days and a successful backup occurs on Wednesday, the next full backup does not occur until the following Wednesday.Typically, incremental backups have a shorter frequency than full backups. |
| **Calendar** | This setting is used only for scheduled backups. It is not used for user-directed backups. The **Calendar** option allows you to schedule backup operations based on specific dates, recurring week days, or recurring days of the month. |

Description of Schedule Properties (continued)

| Property | Description |
|---|---|
| **Retention** | **Frequency-based scheduling**<br>The retention period for an Application Backup Policy schedule refers to the length of time that NetBackup keeps backup images.<br>For example, if your database is backed up once every Sunday morning, you should select a retention period of at least 2 weeks.The retention period for an Automatic Backup schedule controls how long NetBackup keeps records of when scheduled backups have occurred.<br>Set a retention period that is longer than the frequency setting for the schedule. For example, if the frequency setting is set to one week, set the retention period to be more than one week. The NetBackup scheduler compares the latest record of the Automatic Backup schedule to the frequency of that Automatic Backup schedule to determine whether a backup is due. This means that if you set the retention period to expire the record too early, the scheduled backup frequency is unpredictable. However, if you set the retention period to be longer than necessary, the NetBackup catalog accumulates unnecessary records.<br>**Calendar-based scheduling**<br>The retention period for an Application Backup Policy schedule refers to the length of time that NetBackup keeps backup images.<br>The retention period for an Automatic Backup schedule controls how long NetBackup keeps records of when scheduled backups have occurred. However, this setting is not significant for calendar-based scheduling. |
| **Multiple copies** | If you want to specify multiple copies for your MS-SQL-Server policy, configure **Multiple copies** on the Application Backup schedule. |

Other schedule properties vary according to your specific backup strategy and system configuration. For more information on schedule properties, consult the *NetBackup System Administrator's Guide, Volume I*.

## Types of Backup Schedules

You can perform backups with the following types of schedules:

SQL Server Backup Types

| | |
|---|---|
| Application Backup | The Application Backup schedule enables user-controlled NetBackup operations from the client, both those initiated from the client and those initiated by an automatic schedule on the master server. NetBackup uses the Application Backup schedule when the MS-SQL-Server user starts a backup manually. Configure at least one Application Backup schedule for each MS-SQL-Server policy. The Default-Application-Backup schedule is configured automatically as an Application Backup schedule. |
| Automatic Backup | An Automatic Backup schedule specifies the dates and times for NetBackup to automatically start backups by running the SQL Server scripts in the order that they appear in the file list. If there is more than one client in the SQL Server policy, the SQL Server scripts are run on each client. |

## Adding Clients

The client list is the list of hosts on which your batch files are run during an automatic backup. A NetBackup client must be in at least one policy but can be in more than one.

The following software must be installed on the client:

- SQL Server
- NetBackup client or server
- The backup or restore script(s)

**Note** Each batch file must be present on each client.

### ▼ To add clients to a policy

1. In the Policy dialog, click the **Clients** tab.

   To access the Policy dialog, double-click the policy name in the Policies list in the NetBackup Administration Console.

2. Click **New**.

3. Enter the name of the client you want to add.

   If SQL Server is installed in a NetBackup cluster, specify the virtual SQL Server name as the client name.

   *From the Windows interface*:

   - Type the name into the client list and press Enter.

     If NetBackup cannot detect the hardware and operating system, a dialog box displays so you can specify this information.

     *OR*

   - Click the **Browse for Computer** button to choose the client from the network.

   *From the Java interface:*

   a. The Add Client dialog box displays. In the **Client name** field, type the name of the client you are adding.

   b. Choose the **Hardware and operating system** type and click **Add**.

4. To add another client, repeat step 2 and step 3.

5. If this is the last client, click **OK**.

## Adding Backup Selections

The backup selections list in a database policy has a different meaning than for non-database policies. For example, in an Windows policy, the list contains files and folders to be backed up. In a NetBackup for SQL Server database policy, you can specify batch files to be run. NetBackup for SQL Server scripts are called batch files and are identified by the .bch extension. Batch files describe the backup operations you wish to start through manual or scheduled operations initiated from the NetBackup server. These files reside on the client and direct the operation of NetBackup for SQL Server and SQL Server. Refer to “Overview of Batch Files” on page 100 for details on how to create a batch file.

Add batch files only if you are setting up a policy for automatic scheduling. All batch files listed in the backup selections list are run for manual backups and for Automatic Backup schedules as specified under the **Schedules** tab. NetBackup runs the batch files in the order that the batch files appear in the Backup Selections list.

### ▼ To add batch files to the backup selections list

The following steps show how to perform this procedure from both the Java interface and from the Windows interface.

**Caution** Be sure to specify the correct batch file names in the Backup Selections list to prevent an error or possibly a wrong operation. Make sure that the batch file resides on the client before you try to add it to the Backup Selections list.

*From the Java interface*:

1. Open the Policy dialog.

   To access the Policy dialog, double-click the policy name in the Policies list in the NetBackup Administration Console.

2. Click the **Backup Selections** tab.

3. Click **New**.

   A dialog box displays.

4. Specify the names of the batch files that you want NetBackup to use.

   a. In the **Script:** box, type the full path name of a batch file on the client.

      For example, a path name for a batch file that will back up a database may be:

      *`install_path`*`\NetBackup\DbExt\Mssql\bkup.bch`

**Note** *`install_path`* refers to the directory where you installed the NetBackup software. By default, this directory is `C:\Program Files\Veritas\`.

      You can also specify just the file name if the batch file resides in the *`install_path`*`\NetBackup\Dbext\Mssql\` directory.

   b. Click **Add** to add the batch file to the list.

   c. Repeat step a and step b until all batch files are added.

5. Click **OK**.

*From the Windows interface*:

1. In the Policy dialog, click the **Backup Selections** tab.

   To access the Policy dialog, double-click the policy name in the Policies list in the NetBackup Administration Console.

2. Click **New**.

3. Specify the names of the batch files you want NetBackup to use.

   Use one of the following methods:

   ◆ Type the full path name of the batch file on the client.

     For example, a path name for a batch file that will back up a database may be:

     *`install_path`*`\NetBackup\DbExt\Mssql\bkup.bch`

**Note** *`install_path`* refers to the directory where you installed the NetBackup software. By default, this directory is `C:\Program Files\Veritas\`.

     You can also specify just the file name if the batch file resides in the *`install_path`*`\NetBackup\Dbext\Mssql\` directory.

- Click the **Remote Folder** button.

  The Browse window displays the hosts in the client list. Navigate to and select the batch file file. Click **OK**.

4. Click **OK**.

# Testing Configuration Settings

After configuring the master server for NetBackup for SQL Server, test the configuration settings by performing a manual backup (or backups) using the automatic backup schedules you have created.

### ▼ To test the configuration settings

1. Log onto the master server as administrator (Windows) or root (UNIX).

2. Start the NetBackup Administration Console.

3. In the left pane, click **Policies**.

   *From the Windows interface:* The policy list appears in the right pane.

   *From the Java interface:* The right pane splits into an **All Policies** pane and a details pane.

4. Click the policy you want to test.

5. Choose **Actions** > **Manual Backup**.

   The Manual Backup dialog box appears.

   The Schedules pane contains the name of an automatic schedule (or schedules) configured for the policy that you are going to test. The Clients pane contains the name of the client(s) listed in the policy that you are going to test.

6. Follow the directions in the Manual Backup dialog box.

7. Click **Activity Monitor** on the NetBackup Administration Console.

   If the manual backup does not exit with a successful status, see the Troubleshooting chapter.

For a description of status codes and other troubleshooting information, see the *NetBackup Troubleshooting Guide*.

# Configuring Multi-Stream Operations

NetBackup supports multi-striped and multiplexed backups of SQL Server. Backup streams can each be written to separate storage units or multiplexed to a single storage unit.

## Configuring Multi-Striped Backups

SQL Server supports backing up databases through multiple data streams, which are called *stripes*. NetBackup stores each stripe as a separate image. The purpose of this feature is to speed up the the rate of data transmission by using multiple tape devices in case the tape device is a bottleneck.

VERITAS does not recommend enabling multiplexing for a schedule used for backing up multiple stripes.

When restoring from a multi-stream backup, all backup images must be available simultaneously. See the section"Restoring Multi-Streamed Backups" on page 111 for instructions on performing multi-stream restores.

### ▼ To configure a multi-striped backup policy

1. In your *Application Backup* policy, set **Media Multiplexing** to **1** in the schedule you will use for striped backups or restores.

   This disables multiplexing for this schedule. This will make all streams simultaneously available so restore operations will be successful.

2. Ensure that the storage unit has as many drives as you will have stripes.

3. Configure schedules for backups and restores so that enough drives are available at the time you will be performing striped backups or restores.

   Select a **Stripes** setting greater than 1 in the Backup Microsoft SQL Server Objects dialog.

## Configuring Multiplexed Backups

NetBackup provides a multiplexing feature that allows you to overlay multiple backups to the same tape. This is useful if you have many simultaneous backups using the same tape drive. *Do not* use this feature for performing multi-striped SQL Server backups.

See the section "Restoring Multi-Streamed Backups" on page 111 for information on how to perform striped restores.

### ▼ To configure multiplexed backups

1. In the schedule for your *Application Backup* policy, set **Media Multiplexing** to the number of backup stripes that you plan to use.

2. In the storage units associated with this schedule, set **Limit Multiplexing per Drive** to the number of stripes that you plan to use.

3. Select a **Stripes** setting equal to **1** in the Backup Microsoft SQL Server Objects dialog.

# Configuring for a Multi-Interface Network Connection

Many administrators desire to reserve a separate network interface for their SQL Server host machines that are used for routing backup traffic. This type of configuration requires some special considerations for configuring both the NetBackup master server and the NetBackup client that backs up SQL Server. There are also special instructions for browsing backup images across a required network interface.

See "Performing Restores with a Multi-Nic Connection" on page 95 for information on browsing across multi-nic connections. See also "Backing Up SQL Server in a Cluster with a Multi-Interface Network Connection" on page 127 provides a description of how to protect SQL Server in a cluster.

## Master Server Configuration

Two configuration changes must be made on the master server to allow for backups and restores over a private interface: the backup policies must include the private interface name in the Clients list of the policy and permissions must be added to allow for browsing of backups across the private interface.

In the following examples, the NetBackup client machine that backs up SQL Server is referred to as SQLHOST and the private interface used for backing up SQLHOST is referred to as SQLHOST-NB.

### Adding Clients to the Policy

The private name of the client must be added to the Clients list of the policy. The NetBIOS or public name of the client should not be used.

▼ **To add clients to the policy**

1. Open the NetBackup Administration Console.

2. Create a new policy or open an existing policy.

   In the Clients list for the policy, add a new client. Specify the Client name as the private interface name (in this example, SQLHOST-NB).

### Adding Permissions That Allow for Browsing of Backups Across the Private Interface

The administrator can allow all clients or allow single clients to browse and restore backups performed over the multi-nic connection.

▼ **To allow all clients to browse for backups and perform restores**

❖ Add the empty file `NB_INSTALL\db\altnames\No.Restriction` on the master server.

This option allows any NetBackup client to browse backups made by the SQL Server host machine.

▼ **To allow a single client to browse for backups and perform restores**

❖ Add the empty file, `NB_INSTALL\db\altnames\`*`SQLHOST`* on the master server.

This only allows the SQL Server host machine, SQLHOST, to access backups managed by the master server.

## Client Configuration

To configure backups of a client in a multi-nic environment, the private and public interface names must be provided for NetBackup. The private interface name of the SQL Server host machine is specified in the Backup, Archive, and Restore interface on the client. This can be set either during the installation of the NetBackup client, or it can be changed in the NetBackup Client Properties dialog in the Backup, Archive, and Restore interface, as described below.

In the following examples, the NetBackup client machine that backs up SQL Server is referred to as SQLHOST and the private interface used for backing up SQLHOST is referred to as SQLHOST-NB.

▼ **To specify the private name of the client**

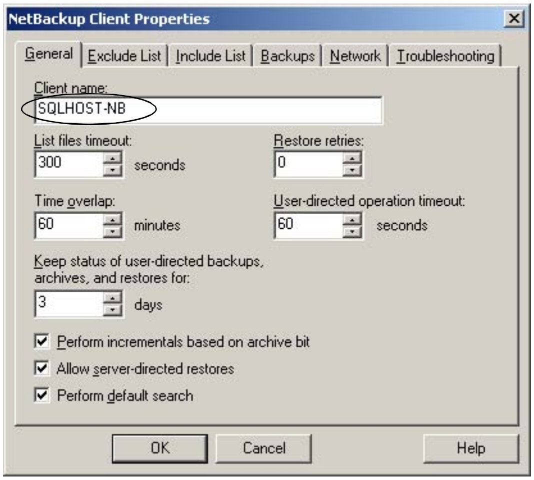
1. Open the Backup, Archive, and Restore interface.

2. Choose **File** > **NetBackup Client Properties**.

3. Click on the **General** tab.

4. In the **Client name** box, specify the private name of the client.

For example, the private name for the machine SQLHOST would be SQLHOST-NB.



NetBackup Client Properties

General | Exclude List | Include List | Backups | Network | Troubleshooting

Client name: SQLHOST-NB

List files timeout: 300 seconds

Restore retries: 0

Time overlap: 60 minutes

User-directed operation timeout: 60 seconds

Keep status of user-directed backups, archives, and restores for: 3 days

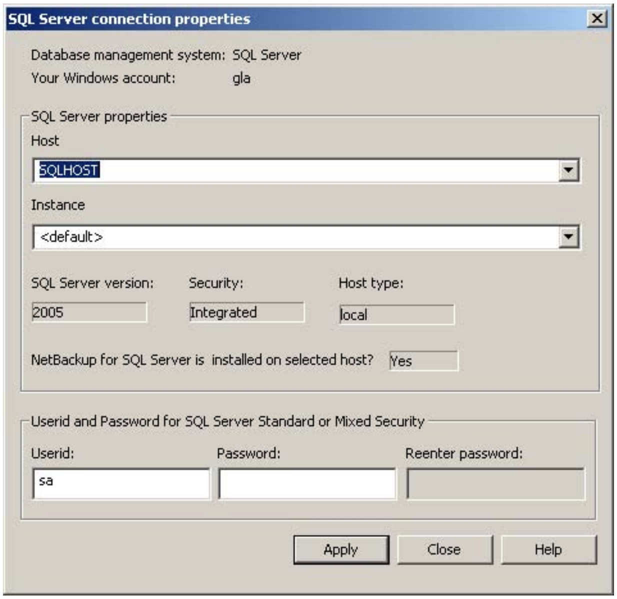
☑ Perform incrementals based on archive bit

☑ Allow server-directed restores

☑ Perform default search

## ▼ To specify the public name of the client

1. Open the NetBackup for SQL Server interface.
2. Choose **File** > **Set SQL Server connection properties**.
3. In the **Host** box, specify the public name of the SQL Server host (SQLHOST).



SQL Server connection properties

Database management system: SQL Server

Your Windows account: gla

SQL Server properties

Host: SQLHOST

Instance: <default>

SQL Server version: 2005

Security: Integrated

Host type: local

NetBackup for SQL Server is installed on selected host? Yes

Userid and Password for SQL Server Standard or Mixed Security

Userid: sa

Password:

Reenter password:

# SQL Server Privileges

When you install SQL Server, you can elect to use either integrated or standard security. Integrated security refers to the usage of Windows authentication in lieu of standard SQL Server-based logins.

**Note** Microsoft recommends that you use integrated security because Windows logins, unlike SQL Server-based logins, can be traced using standard Windows security tools. NetBackup for SQL Server supports both integrated and standard security for any level of SQL Server.

If you choose integrated security, then SQL Server ignores the userid and password that you provide in the SQL Server connection properties dialog of the NetBackup Database Extension GUI. Instead, security validation to perform backup and restore operations and for querying the master database will be based upon the SQL Server privileges that have been granted to the Windows account. The batch file userid and password keywords are also ignored. When integrated security is used SQL Server operations generated by NetBackup for SQL Server will always be validated against the Windows userid.

If you choose standard security, then you must supply NetBackup for SQL Server with a SQL Server-based userid and password. This userid and encrypted password are stored in the registry under this key:

```
HKEY_CURRENT_USER\SOFTWARE\VERITAS\NETBACKUP\NetBackup for SQL
Server\
```

In order to perform an operation with standard SQL Server security, NetBackup for SQL Server looks up the SQL Server userid and password from the registry.

**Note** Since NetBackup for SQL Server does backup and restore commands and issues select statements against the master database, you should review the SQL Server user documentation in order to determine exactly what privileges are required by the user accounts that you establish for NetBackup for SQL Server.

## Setting the SQL Server Login for Scheduled Operations

When you use the NetBackup Database Extension GUI, SQL Server security is satisfied based upon either your current Windows account, if you are using integrated security, or a SQL Server-based userid and password. For information on integrated and standard security, see the previous section “SQL Server Privileges”.

When the NetBackup scheduler is used, backup operations are launched from a daemon process called the NetBackup Client Service. You must ensure that the Client Service has SQL Server privileges to perform backup and restore operations. If you are using integrated security, then you must ensure that the NetBackup Client Service specifies a Windows account that has been granted SQL Server administrative privileges.

**Note** If your site does not place any SQL Server security restrictions on the right to back up databases then the steps described in the next two sections will be unnecessary.

## Authorizing Scheduled Operations (for Sites with SQL Server Security Restrictions)

If you will be using NetBackup to perform automatic SQL Server backups and if your site places SQL Server security restrictions on the right to back up databases, then an account must be set up which the NetBackup Client Service can use to login to SQL Server. The account must have sufficient SQL Server privileges to perform a backups.

### ▼ To set up the NetBackup Client Service to login to SQL Server

1. If you are using standard SQL Server security, first map a Windows account to a standard SQL Server userid. (See “How to Use Standard SQL Server Security” on page 33.)

2. Open the Windows Services application.

3. Double-click on the **NetBackup Client Service** entry.

4. Click on the **Log On** tab.

5. Select **This account**.

6. Type the name of the account, or browse for and select an account.

7. Type the password and the password confirmation.

**Note** If you later change the password, then you must also indicate the new password for the NetBackup Client Service, then restart the NetBackup Client Service. If you do not do this, NetBackup scheduled backup operations will fail.

8. Log on to Windows with the Windows account selected in step 6. Then perform the procedure described in "Selecting the SQL Host and Instance" on page 57. The database login parameters that you choose will be selected for SQL Server backups launched from the NetBackup scheduler.

9. Stop and restart the NetBackup Client Service from the service dialog box.

## How to Use Standard SQL Server Security

If you are using standard SQL Server security, the NetBackup Client Service must be started with a Windows account associated with a standard SQL Server userid that has sufficient administrative privileges to perform backups and restores.

### ▼ To associate a Windows userid with a SQL Server userid

1. Log on to your Windows host using the Windows account which you intend to use for the NetBackup Client Service.

2. Start the NetBackup Database Extension GUI and go to the DBMS login parameters window. Type the standard SQL Server login name and password into the edit boxes provided. The values that you supply will be encrypted in a registry entry that is associated with the current Windows account.

3. Follow the instructions in previous section "Authorizing Scheduled Operations (for Sites with SQL Server Security Restrictions)" to set the login account for the NetBackup Client Service.

# Performance Factors

Backup performance can be influenced by your hardware environment as well as settings in SQL Server, the NetBackup master server, and NetBackup for SQL Server. This section discusses the most prominent factors in SQL Server and NetBackup for SQL Server. The availability of buffer space has a definite influence on how fast backups run.

## Buffer Space Parameters

There are three parameters you can use to increase buffer space in SQL Server.

### MAXTRANSFERSIZE

The MAXTRANSFERSIZE parameter can be set for each backup or restore operation. It ranges in size from 64 kilobytes to 4 M bytes. The default is 64 kilobytes. MAXTRANSFERSIZE is the buffer size used by SQL Server for reading and writing backup images. Generally, you can get better SQL Server performance by using a larger value.

Set the MAXTRANSFERSIZE for your NetBackup for SQL Server session by choosing **File** > **Set NetBackup client properties** or add this parameter to batch file. For more details on the MAXTRANSFERSIZE parameter, see "Overview of Batch Files" on page 100.

### BLOCKSIZE

The BLOCKSIZE parameter can be set for each backup operation. For restores, NetBackup for SQL Server automatically chooses the same size that you backed up with. BLOCKSIZE ranges from 512 bytes to 64 kilobytes. The default is 512 bytes. BLOCKSIZE is the incremental size that SQL Server uses for reading and writing backup images.

Set the BLOCKSIZE for your NetBackup for SQL Server session using the NetBackup client properties dialog. Otherwise, you can use the BLOCKSIZE parameter to set this value when you create a batch file manually. For more details on the BLOCKSIZE parameter, see "Overview of Batch Files" on page 100.

### NUMBER OF BUFFERS PER STRIPE

The NUMBER OF BUFFERS PER STRIPE parameter affects buffer space availability. This parameter is set in the NetBackup for SQL Server administration GUI for each operation. NetBackup for SQL Server uses this parameter to decide how many buffers to allocate for reading or writing each data stream during a backup or restore operation. By allocating a greater number of buffers, you can affect how quickly NetBackup for SQL Server can send data to the NetBackup master server.

The default value for the NUMBER OF BUFFERS PER STRIPE is 1. For operations that you start from the NetBackup Database Extension GUI, you can change this value through the NetBackup client properties dialog. See "NetBackup client properties" on page 205. Otherwise, you can use the BUFFERS parameter to set this value when you create a batch file manually. For more details on the BUFFERS parameter, see "Overview of Batch Files" on page 100.

## Backup Stripes

You can divide your backup operation into multiple concurrent streams in any situation in which SQL Server dumps data faster than your tape drive is capable of writing. Although NetBackup supports multiplexing many stripes to the same drive, you generally will realize performance gains only if you use the same number of tape drives as streams.

To divide a backup operation into multiple stripes, use the **How Many MS SQL Server Stripes?** edit box, which appears in the Backup Microsoft SQL Server Objects dialog of the NetBackup Database Extension GUI.

## Shared Memory Usage

If NetBackup Server is installed on the same host as NetBackup for SQL Server, then you will obtain optimal performance by using shared memory for data transfer instead of sockets. Shared memory is the default for this configuration and will be used unless you create a `install_path\NetBackup\NOSHM` file.

## Alternate Buffer Method

NetBackup for SQL Server supports an alternate buffer method which optimizes CPU usage by allowing NetBackup and SQL Server to share the same memory buffers without transferring data between them.

The alternate buffer method for backup and restore typically does not improve data transfer rate, only CPU utilization. A situation may occur in which the transfer rate is significantly degraded when alternate buffer method is in use. To improve the transfer rate set the MAXTRANSFERSIZE for the backup to the maximum allowed, which is 4 megabytes. Set this parameter by choosing **File** > **Set NetBackup client properties** or add this parameter to a batch file. For more details on the MAXTRANSFERSIZE parameter, see "Overview of Batch Files" on page 100.

### Alternate Buffer Method with Backup Operations

This method is chosen automatically for backups if all of the following conditions apply:

- NetBackup shared memory is in use.
- The backup is non-multiplexed.
- The NetBackup buffer size equals the SQL Server blocksize.

  The default NetBackup buffer size is 64 kilobytes, but this may be overridden by the value contained in:

  `install_path\Netbackup\db\config\SIZE_DATA_BUFFERS` (for tape backups), or,

  `install_path\Netbackup\db\config\SIZE_DATA_BUFFERS_DISK` (for disk backups)

  You can set the SQL Server blocksize either in the NetBackup client properties dialog of the NetBackup Database Extension GUI; or you can adjust it directly using the `BLOCKSIZE` option in your batch file.

- NetBackup for SQL Server must be started with the same account as the NetBackup Client service.

  Backups initiated from an automatic backup policy are started with the NetBackup Client service so the same account is already in use. However, if you start an SQL Server backup through NetBackup for SQL Server or through `dbbackex`, then your logon account must be the same as the NetBackup Client service account in order for your backups to be candidates for the alternate buffer method.

### Alternate Buffer Method with Restore Operations

In addition to the above conditions for backups, the alternate buffer method for restore also requires that the following conditions be met:

- The version of SQL Server being restored must be SQL Server 2000 or SQL Server 2005.
- The backup must have been made with alternate buffer method.

You can verify that the alternate buffer method is being used by looking for the words `Using alternate buffer method`, which appear in the `dbclient` log and the progress report.

## Page Verification

With SQL 2005, you can choose to perform page verification before performing a backup or restore. Enabling this option will impose a performance penalty on a backup or restore operation.

## Instant Data File Initialization

When you restore a database, filegroup, or database file, SQL Server zeroes the file space before beginning the restore operation. This can slow the total recovery time by as much as a factor of 2. To eliminate file initialization, run the MSSQLSERVER service under a Windows account that has been assigned the SE_MANAGE_VOLUME_NAME. For more information about eliminating file initialization, see the SQL Server 2005 and Windows 2003 documentation.

## Using Read-Write and Read-Only Filegroups

You can significantly reduce backup time and storage media needed if you periodically back up only read-write filegroups and keep a single backup of read-only filegroups, which is retained infinitely.

See “Adding Schedules” on page 15 for information on setting the retention in a schedule contained in a NetBackup for SQL Server Policy.

# Backing Up SQL Server in an Environment with Log Shipping

Log shipping is a SQL Server feature that may be employed to enhance the overall availability of your installation. It uses a primary server, which contains the active database, a monitor, and one or more secondary servers. Under log shipping, copies of the transaction log are supplied to the secondary servers on an on-going basis to the secondary servers. This allows each secondary server to be in a standby state in case the primary goes offline.

Many sites also use the secondary server to off-load certain activities from the primary in order to minimize its load. *However, a backup must not be performed on a secondary (or standby) server*. Databases must always be backed up on the primary server and restored on the primary server. This requirement is based on Microsoft SQL Server restriction that is outlined in Microsoft Knowledge Base Article 311115. If an attempt is made to perform a backup on the secondary server, the result cannot be predicted. However, you may see a message in the dbclient log similar to the one below:

```
16:33:26 [1208,2348] <16> CODBCaccess::LogODBCerr: DBMS MSG - ODBC
message. ODBC return code <-1>, SQL State <37000>, Message Text
<[Microsoft][ODBC SQL Server Driver][SQL Server]Database 'Mumbo'
is  in warm-standby state (set by executing RESTORE WITH STANDBY)
and cannot be backed up until the entire load sequence is
completed.>
```

# Backup and Recovery Concepts 4

This chapter describes basic concepts of backup and recovery of SQL Server and how NetBackup for SQL Server performs these operations.

# Overview

A SQL Server instance is created on a Windows host by installing Microsoft SQL Server. You can install multiple instances on a single host including a default instance and multiple named instances. You can select the database instance that you wish to browse and backup by selecting **File** > **Set SQL Server connection properties**.

## SQL Server System Database Types

A SQL Server instance cannot be backed up as a single entity. The largest granularity of a SQL Server backup is the database. SQL Server has system and user databases. The system databases are:

- *Master.* This is the "brains" of your installation, containing a great deal of the metadata describing your instance. Be sure to retain an up-to-date backup of the master database. In particular, back up the master any time you have made changes to your SQL Server installation, including when you have created or modified other databases. Note that you can only do full database backups on the master. You cannot back up its component files, perform differentials, or backup up its transaction log. Recovery of the master database requires special considerations and is described in "Disaster Recovery" on page 115.
- *msdb*. The msdb is used by the Microsoft SQL Agent for scheduling, alerts, and for recording the backup history. It is amenable to all types of backup providing that it has the full or bulk-load recovery option set.
- *Model*. The model database serves as a template for new databases when the Create Database statement is executed. It is also amenable to all types of backup, providing that it has the full or bulk-load recovery option set.
- *tempdb*. The tempdb is for temporary databases used by applications. It cannot be backed up and does not appear in the NetBackup for SQL Server backup browser.

## Database Backups

Three types of backup operations can be performed on databases:

- *Full.* The database, including all of its component files are backed up as a single image. The log file is included in a full database backup.

**Note** The transaction log is not automatically truncated following a full backup. Thus a common practice to preserve disk space is to manually truncate the transaction log following a successful full backup.

- *Differential*. All of the changes since the last full are backed up to a single image.

- *Transaction log*. Transaction log backups are only available for the full and bulk-load recovery options. In this operation, the inactive portion of the transaction log is backed up. Four options are available when you select transaction log backup:

  - Backup and truncate the transaction log.
  - Backup but don't truncate the transaction log.
  - Truncate the transaction log, but back it up.
  - Backup and restore the tail end of the log. (SQL Server 2005, only)

  The third option is not a backup at all. It just provides a convenient way to truncate the transaction log for you. It is typically used after a full database backup. The last option is a backup but it does not create a permanent backup image. A backup and a restore of the tail end of the log is typically used following a page-level restore when the database is recovered, but all of the filegroups have not been brought back on line. By backing up and recovering the tail end of the log, SQL Server is able to bring the database back to a usable state.

## Filegroup Backups

In addition to database backups, it is also possible to back up database components. The first-level component of a database is the filegroup. Each filegroup, is made up of one or database files. Database files can also be backed up to individual backup images.

Note that the log file stands out as an exception to the normal decomposition in that the log file does not belong to a filegroup. The log file also cannot be directly backed up through SQL Server a distinct image.

There are three ways in which filegroups can be backed up:

- *Filegroup Backups*. This is a backup created from a single filegroup. Scripts for filegroup backups are created when you select individual filegroups in the object browser of the backup database dialog box.
- *Read-Write Filegroups Backups (SQL 2005 only)*. This is a backup that contains only the read-write filegroups in a database. If all of the filegroups in a database are read-write, then the read-write filegroup backup will have the same content as a full database backup. Scripts for read-write filegroup backups are created when you select individual databases in the object browser of the backup database dialog box and choose the type of backup as 'read/write filegroups'.
- *Partial Database Backup (SQL 2005 only)*. This is a database backup that contains an improvised selection of filegroups selected by the user. NetBackup for SQL Server creates a template for partial database backups when you select individual databases in the object browser of the backup dialog and chose the type of backup as 'Create a

partial database template'. The template is created with all of the filegroups commented-out. You can chose the filegroups to be included in the partial backup by selectively uncommenting the filegroups.

**Caution** Since the contents of a partial database backup are user-defined,NetBackup for SQL Server does not use them for staging recovered backups. So if you are relying on NetBackup to stage database recovery for you, the partial backup may not be a good choice.

## Differential Backups

Differential backups can be created on the full database as well as all three types of filegroup backup units as described in the previous section. A differential backup contains the changes to the contents of the object since the last time that it was captured in a full backup.

# Protecting Files and Filegroups

If you plan to back up files and filegroups as a part of your plan for protecting SQL Server, then the database must use the full or bulk-logged recovery model. In addition, you must maintain the unbroken sequence of transaction log backups. You will be responsible for creating the files and filegroups for your databases and for placing individual database components into them. NetBackup places a restriction on the layout of your database in order to successfully perform backup and restores of database files and filegroups.

Before attempting a file and filegroup backup, you must ensure no table is placed into a filegroup which is different than any one of its indices.

For example, the layout as indicated by the following Transact SQL statements should not be used:

```
use master
CREATE DATABASE MultiFileDB
ON
PRIMARY ( NAME = FileX,
          FILENAME = 'd:\mssql\data\FileX.mdf'),
FILEGROUP AltGroup
( NAME = AltGroupFil,
  FILENAME = 'd:\mssql\data\AltGroupFil.ndf')
GO
use MultiFileDB
CREATE TABLE Table1 (col1 char(10),col2 char(10), col3 char(10)) on AltGroup
go
create unique clustered index index4 on Table1 (col2)
go
```

Notice in this example, Table1 has been placed in filegroup AltGroup but its index is placed (by default) in the primary filegroup.

If you do place a table into a different filegroup than one of its indices and use NetBackup for SQL Server to back it up, you may fail with the following SQL Server error message:

```
Database file <file name> is subject to logical recovery and must be among
the files to be backed up as part of the file or filegroup backup.
```

# Recovery Considerations for Files and Filegroups

To ensure that you will be able to successfully restore a database from file and filegroup backups, it is important to always have backups of a full set of files and filegroups that constitute the entire database, as well as transaction log backups that span the entire period of time over which the backups were taken. To maintain an unbroken sequence of transaction log backups, it is essential to perform a transaction log backup following every file or filegroup backup. (If you back up several files or filegroups at once, then you only need to back up the transaction log after the last such backup.) If the transaction log is not backed up, SQL Server will not allow you to restore a file or filegroup.

SQL Server does not keep a record in the transaction log of new files or filegroups that are created. Therefore, after you add either a file or a filegroup to the database, you must immediately back it up. Then perform a backup of all the filegroups in the database so NetBackup will select the correct recovery set when subsequent backups are performed. Similarly, after creating a database file, you should back up all of the files in the filegroup to which it belongs.

# Reducing Backup Size and Time By Using Read-Only Filegroups (SQL Server 2005)

Many applications contain a substantial amount of data that does not change. For example, under time-based partitioning, historical data may be preserved indefinitely while only a fraction of the database is subject to change at any given time. With SQL Server 2005, static filegroups can be classified as read-only. NetBackup uses the filegroup read-only designation to optimize the total backup volume speed of recovery.

For backups, the advantage in using read-only filegroups is that you can reduce total media usage by backing up the read-only filegroups once and retaining the backup image indefinitely. This reduces the total time spent on performing backups because only read-write data is backed up periodically.

For recovery, the advantage is that you can bring your database on-line more quickly because read-only filegroups do not need to be restored from backup media unless they are corrupted due to disk error or other hardware failure.

Backup Strategy for Read-Only and Read-Write Filegroups

| Task | Notes | Refer to |
|---|---|---|
| **1.** Create a policy for read-only filegroups. | Create an Application Backup schedule with Retention set to infinite. | "Adding Schedules" on page 15 |
| **2.** Create a policy for read-write filegroups. | Create an Automatic Backup schedule with the desired Retention period and add the read-write filegroups batch file to the Backup Selections list. | "Adding Schedules" on page 15 |
| **3.** Back up read-only filegroups | In the NetBackup for SQL Server interface, select the policy you created in step 1. Perform this backup once.<br>All read-only filegroups must be included in some combination of full, partial, or individual filegroup and file backups.<br>**Note** Immediately back up any filegroup when changing it from read-write to read-only. | "Backing Up Read-Only Filegroups (SQL 2005)" on page 63 |
| **4.** Back up read-write filegroups | This will be done automatically through he schedule you created in step 2. Or you can manually back up the read-write filegroups. | "Backing Up Read-Write Filegroups (SQL 2005)" on page 64 |

Backup Strategy for Read-Only and Read-Write Filegroups

| Task | Notes | Refer to |
|---|---|---|
| **5.** View the read-only backup set. | Do this if needed to confirm all read-only groups have been backed up. | "Viewing Read-Only Backup Sets" on page 63 |

# Recovery Factors

This section is provided to assist you in defining a recovery plan which is suitable to your application environment. Much of this information is based on Microsoft's *SQL Server Books Online*. Refer to that resource for a more inclusive discussion.

## Transaction Logs

SQL Server maintains a write-ahead transaction log for each database. This log helps to maintain database updates in cache memory to ensure that data is not written to disk before it has been committed. Database writes occur as a part of the checkpoint procedure. Checkpoint frequency is determined by SQL Server based upon the "recovery interval" which is a configuration parameter indicating the maximum time interval that can be tolerated during a system restart. When checkpoint occurs the portion of the transaction log that is no longer needed for system restart becomes inactive and is optionally truncated, depending upon the recovery strategy in place, as described in the next section.

If the transaction log is not truncated by the checkpoint procedure, then it can be backed up and used for point-in-time recovery, failure from disk crash, or move and copy operations.

## Recovery Strategies

SQL Server 2000 and 2005 provide three basic levels for database recovery which have different implications for both backup performance and for the granularity of recovery. These levels are:

- *Simple*. With this method the inactive portion of the transaction log is not retained beyond the database checkpoint, this method provides for minimal usage of log space. However, the database can only be restored to the last full backup. Transaction log restores, including point in time recovery and named transaction recovery are not supported. In addition, maximum performance is provided for bulk operations, such as (Create Index, Select Into, and Bulk Copy) because they are not logged.
- *Full*. Using this method, the inactive portion of the transaction log is retained until it is truncated, which normally occurs when it is backed up. The transaction log can then be used to stage a recovery either to a point in time or to a named transaction. The Full Recovery model provides maximum recoverability but it uses the most log space and does not provide maximum performance for bulk operations.
- *Bulk-Logged*. This method is identical to the Full Recovery model except that bulk operations are not logged and thus cannot be recovered.

SQL Server 7.0 provides similar recovery except that functionality is defined by two attributes, these are:

- *Truncate Log on Checkpoint*. Enables “simple” logging, that is, the inactive portion of the transaction log is truncated on checkpoint.
- *Select Into/Bulk Copy*. Bulk operations are not logged.

## Backing up the Transaction Log

Transaction logs can be backed up in SQL Server 2000 or SQL Server 2005 for databases which are set in either full or bulk-logged mode. In SQL Server 7.0, transaction logs can be backed up if “Truncate Log on Checkpoint” has *not* been selected. By default a transaction log is truncated after it has been backed up. However, it is not truncated following a full database or differential backup.

The main factors in deciding how frequently to backup a transaction log would be

- Conservation of log space.
- How close to the failure point you must be able to recover in case of a disk crash.

During peak periods in a high transaction environment, it may not be unusual to back up the transaction log on an hourly basis.

## Differential Backups

Unlike the transaction log backup, the differential backup is a backup of the database. The differential includes all of the changes made since the last full backup. If you have made several differential backups since the last full backup, then to accomplish a full restore to the end of the last differential backup, you would only need to restore the last full database, followed by the last differential. You would not need to restore any of the intermediate differentials.

There are four types of differential backups:

- Databases differential
- Individual filegroup differentials
- Read-write filegroup differentials, i.e., backups that include differentials on all the read-write filegroups in a database.
- Partial differential filegroups, i.e., backups that include differentials of just the filegroups that the user selects.

**Caution** Microsoft recommends you do not create more than one type of differential backup for the same object.

**Caution** NetBackup does not consider differential images in determining recovery staging strategies if more than one type of differential is found for the same object. (See "Staging Recovery" on page 50).

A typical backup procedure may use full database, differential, and transaction log backups in ascending order of frequency. For example the full database backup may be taken bi-weekly, the differential may be taken nightly, and the transaction log backup may be made as frequently as every hour for either mission critical or high volume applications.

## File and Filegroup Backups

In addition to full, database differential, and transaction log backups of a database, SQL Server also supports backing up individual filegroups and files as distinct images. Since a filegroup is composed of one or more database files, backing up the constituent files of a filegroup is logically equivalent to backing up the filegroup itself.

Filegroup and file backups would commonly be used in a tightly architected application in which physical disk locations were mapped to logical objects, such as tables and indexes. There are several factors which may lead to the choice to use file and filegroup backups in this type of environment.

- Some portions of the database, which may be distinguished by such factors as volatility or mission criticality, may be recognized as having to be backed up more frequently than other portions.
- The database may be so large that the time required for a full database backup could not fit in the allocated time window. Thus it may be more viable to do a full backup of one or more files or filegroups on a rotating basis.
- You may wish to optimize on backup volume and recovery speed by placing some of your data into read-only filegroups. See "Reducing Backup Size and Time By Using Read-Only Filegroups (SQL Server 2005)" on page 45.

The filegroup and file backup also carries the advantage on restore that, in case of disk failure, it would be possible to recover just the failed unit without restoring the entire database.

In order to use filegroup and file backups you must maintain backups of the transaction log. For example, to perform a full database restore using filegroups and files, you would be required to restore all of the constituent filegroups and files in addition to all of the transaction log segments starting from the point at which the first component backup was taken up until a point in time following the last component backup.

## Database Recovery

During the restore process, a database goes into “loading mode” until the restore command is executed against the database with the “recovery” option. Prior to placing the database into recovery mode all of the restore commands would be executed using the “Not recovered” option. This way it is possible to continue to stage additional restore statements to bring the database up to the desired state. The database becomes usable again after the last restore statement has been applied the “Recovered” option.

You can choose the desired recovery option, in the Restore Microsoft SQL Server Objects dialog, when performing restores.

## Staging Recovery

NetBackup for SQL Server keeps track of the backups you have performed and when you performed them. You can display the backup history by opening the “Restore Microsoft SQL Server Objects” dialog (see “Restore Microsoft SQL Server Objects” on page 207). This window depicts all of the SQL Server backup images within the parameters that you specify. The images are displayed in tree-form based on the following backup types:

- database
- database differential
- filegroup
- filegroup differential
- file
- read-only filegroups
- read-only filegroups differential
- partial database
- partial database differential
- transaction log

When you select a transaction log for restore, NetBackup for SQL Server automatically searches for a set of images from which to stage a full database recovery. The recovery set consists of the selected transaction log image plus additional images which could reconstitute the database to a recovered state. A full recovery set can include any one of the following:

1. Full database backup set: A full database image, an optional database differential image, and one or more transaction logs.

2. File and filegroup set: A set of filegroup and/or file images which constitute the database, an optional filegroup differential image for any of the filegroup images, and one or more transaction logs.

3. Database and filegroup differential set: A full database image, a filegroup differential image for each filegroup in the database, and one or more transaction logs.

4. Read-write filegroup set: A read-write filegroup image, an optional read-write filegroup differential image, and one or more transaction logs.

If a full recovery set is found, then all of the composite images will be checked. In addition, **Stage full recovery set** is enabled in the Restore dialog. To view the full recovery set, right-click on the transaction log, choose **Properties**, and click on the **Recovery Set** tab.

# Using NetBackup for SQL Server 5

After completing the installation and configuration, you can use the NetBackup Administration Console to start SQL Server backups or use the NetBackup for SQL Server interface to initiate backups and restores.

This chapter describes how to perform user-directed operations for SQL Server on a Windows client. You can initiate NetBackup operations using the command line interface program, `dbbackex`, or the NetBackup for SQL Server interface.

# Using dbbackex

dbbackex can be executed from a Windows MS-DOS command prompt as

```
install_path\NetBackup\bin\dbbackex -f file [-p policy][-u userid][-pw password]
[-s server][-np]
```

where:

*file* is the name of the batch file, (see "Running Batch Files" on page 108) which describes the operations you wish to start.

*policy* is the MS-SQL-Server type policy, used by NetBackup for the operations specified in the batch file.

*userid* is the SQL Server userid for logging into the database management system.

*password* is the SQL Server password for logging into the database management system.

*server* is the name of the host for the NetBackup master server that you wish to backup to or restore from.

`-np` tells `dbbackex` not to create a message box to indicate the operation status when it has completed.

**Note** Any of the options can be delimited with double quotation marks. Use delimiters, for example, if the file name contains spaces.

The *policy* parameter is ignored for restore operations as the NetBackup server can retrieve the dump file based entirely on the image names specified in the batch file for each restore. Policy is used for backing up databases. If *policy* is omitted, then NetBackup server uses the first active SQL Server policy that it finds in its policy list for carrying out all of the backup operations specified in the batch file.

If *server* is omitted, then the client will use the default server according to the Windows NetBackup client configuration. See the *NetBackup Backup, Archive, and Restore Getting Started Guide* for more information.

**Tip** If you are concerned about protecting login passwords for SQL Server, do not use the `-u` or `-pw` parameters. By omitting these parameters, you can force NetBackup for SQL Server to read the default SQL Server login data from an encrypted file as described in "Starting the NetBackup Database Extension Graphical User Interface for the First Time" on page 56.

Unless you specify `-np`, a message is displayed when `dbbackex` completes. This message will tell you how many operations in the batch file were successful and how many failed.

## Using Client-Based Schedulers with dbbackex

dbbackex allows you to employ your choice of client-based schedulers to automatically initiate NetBackup for SQL Server operations. Two examples follow.

- The Windows Scheduler. This scheduler is initiated through the MS-DOS at command. Instructions for using the Windows Scheduler are provided in the Microsoft Windows online documentation.
- The SQL Server Scheduler. This scheduler is closely integrated with SQL Server. It can be accessed through the Microsoft SQL Server Enterprise Manager.

One distinct advantage of the SQL Server Scheduler is that you can create scripts for database maintenance operations which will be initiated as a result of database events that you define. For example, you can create a script that initiates dbbackex, telling it to back up a particular transaction log, and then create an alert which invokes that script when the transaction log for this database becomes full.

**Tip** If you use dbbackex through a client-based scheduler, specify the -np option to ensure that a message box is not generated. However, before using the scheduler with dbbackex, try the dbbackex syntax on the console *without the -np option* in order to test for the successful completion of the batch file that you have created for your operation.

# Using the NetBackup Database Extension Graphical User Interface

The instructions in this section describe how to use the NetBackup Database Extension - Graphical User Interface to perform NetBackup operations on SQL Server.

## Starting the NetBackup Database Extension Graphical User Interface for the First Time

### ▼ To start NetBackup Database Extension Graphical User Interface for the first time

1. From the Windows **Start** menu, choose **Programs** > **VERITAS NetBackup** > **NetBackup Agents** > **NetBackup MS SQL Client**.

   The first time that you open the NetBackup Database Extension - Graphical User Interface on a new user account, you will be prompted to provide the default SQL host and instance you want to access.

   After clicking **OK**, the SQL Server connection properties dialog is displayed.

2. Select the SQL Server host and instance that you want to log into.

3. Click **Apply** and **Close**.

   The NetBackup Database Extension - Graphical User Interface is displayed.

**Note** For subsequent sessions, the NetBackup for SQL Server agent will remember the login parameters you provided for that Windows account.

## Selecting the SQL Host and Instance

Use this procedure to set which SQL Server host and instance that you want the NetBackup MS SQL Client to access when it logs into SQL Server.

1. Choose **File** > **Set SQL Server connection properties**.

   The SQL Server connection properties dialog is displayed.

2. Use the **Host** drop-down list to select the SQL Server host and the **Instance** drop-down list to select the SQL Server instance.

   These drop-down lists can also be manually edited if the host and instance that you want do not appear in the drop-down lists. You can designate the default instance either by setting the Instance box to <default> or to empty (not spaces).

   If you select a remote host and click **Apply**, the **Host type** will be shown as *remote*.

3. If the SQL Server host and instance use standard or mixed security, then you also need to set the userid and password in the **Userid**, **Password**, and **Reenter password** boxes.

4. Click **Apply** to save your changes and **Close** to exit from the dialog.

## Backing Up Databases

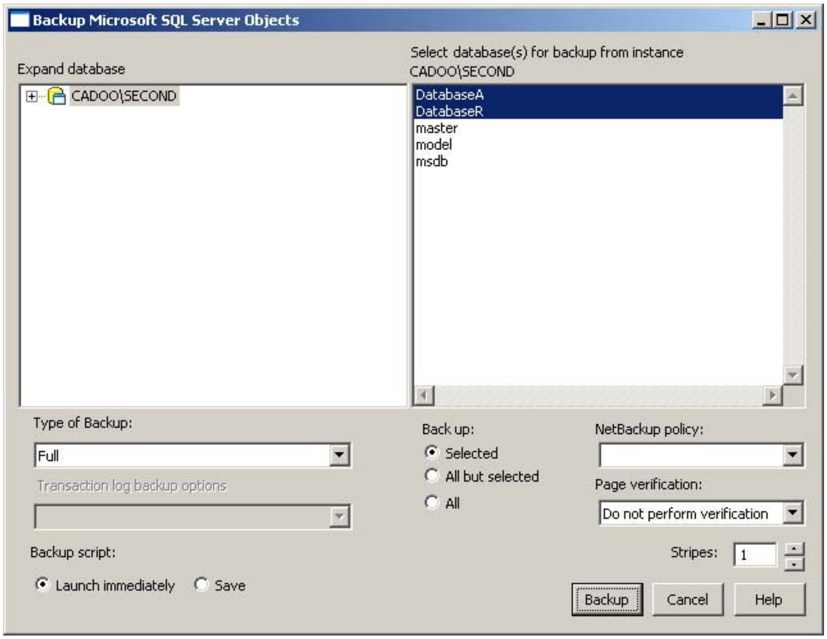
### ▼ To back up a database

1. To change the host and instance you wish to access, refer to "Selecting the SQL Host and Instance" on page 57.

2. Choose **File** > **Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

3. In the right pane, select one or more databases that you want to back up.



Backup Microsoft SQL Server Objects

Expand database

CADOO\SECOND

Select database(s) for backup from instance CADOO\SECOND

DatabaseA
DatabaseR
master
model
msdb

Type of Backup: Full

Transaction log backup options

Backup script: ● Launch immediately ○ Save

Back up: ● Selected ○ All but selected ○ All

NetBackup policy:

Page verification: Do not perform verification

Stripes: 1

4. Choose the Backup type.

- To perform a full backup, choose **Full backup**.
- If you would like to backup the database with the differential option, then select **Perform differential backup**.

5. Select the desired backup options.

See "Backup Microsoft SQL Server Objects" on page 200 for a description of the other available backup options.

6. Click **Backup**.

7. When prompted to start the backup, click **Yes**.

To view the progress of the backup, choose **File** > **View status**.

## Backing Up Transaction Logs

### ▼ To back up a transaction log

1. Before starting a transaction log backup, the database administrator should perform the following actions on the SQL Server.

   ◆ If SQL Server 2000 or later is installed set the *Recovery Model* setting to either *Full* or *Bulk-logged*.

   ◆ If SQL Server 7.0 is installed, set the *Truncate Log on Checkpoint* database option to *Off*. This is an option on the SQL Server interface that applies to the database.

**Caution** You are responsible for ensuring that the entire sequence of transaction logs generated following any database backup are maintained on the same NetBackup server. This means that they should all be backed up to the same facility and that none should be allowed to expire before the others. NetBackup for SQL Server requires that you follow these guidelines in devising your backup strategy in order to provide maximum assistance in restoring your databases.

2. To change the host and instance you wish to access, refer to "Selecting the SQL Host and Instance" on page 57.

3. Choose **File** > **Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

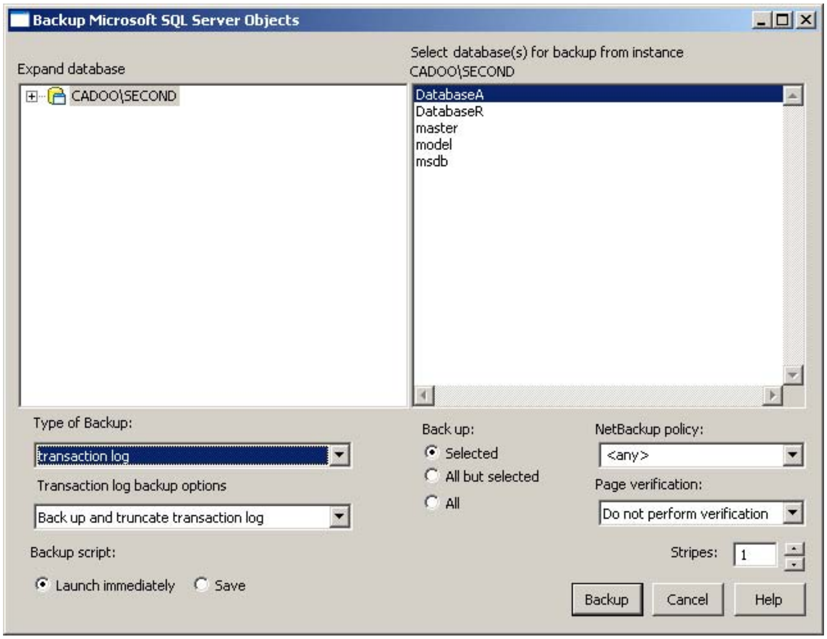
4. In the right pane, select one or more databases whose transaction logs you want to back up.

5. In the Type of Backup list, choose **transaction log**.

6. From the drop-down list, select the desired transaction log option. For more information, refer to the following table.

Transaction log backup options

| | |
|---|---|
| **Back up and truncate transaction log** | Back up the transaction log and remove the inactive part of the transaction log. |
| **Back up transaction log but don’t truncate it** | Back up a transaction log without truncating it. |
| **Truncate transaction log, but don't back it up.** | Truncate the log without performing a backup. |
| **Backup and restore tail log** | Back up and recover the tail log from disk. |



Backup Microsoft SQL Server Objects

Expand database

CADOO\SECOND

Select database(s) for backup from instance CADOO\SECOND

DatabaseA
DatabaseR
master
model
msdb

Type of Backup: transaction log

Transaction log backup options: Back up and truncate transaction log

Backup script: ● Launch immediately ○ Save

Back up: ● Selected ○ All but selected ○ All

NetBackup policy: <any>

Page verification: Do not perform verification

Stripes: 1

7. Select the desired backup options.

   See “Backup Microsoft SQL Server Objects” on page 200 for a description of the other available backup options.

8. Click **Backup**.

   To view the progress of the backup, choose **File** > **View status**.

## Backing Up Database Filegroups

If you are periodically backing up only read-write filegroups as a part of your backup strategy, refer to “Backing Up Read-Write Filegroups (SQL 2005)” on page 64. For information on backing up read-only filegroups, refer to “Backing Up Read-Only Filegroups (SQL 2005)” on page 63

### ▼ To back up a database filegroup

1. To change the host and instance you wish to access, refer to “Selecting the SQL Host and Instance” on page 57.

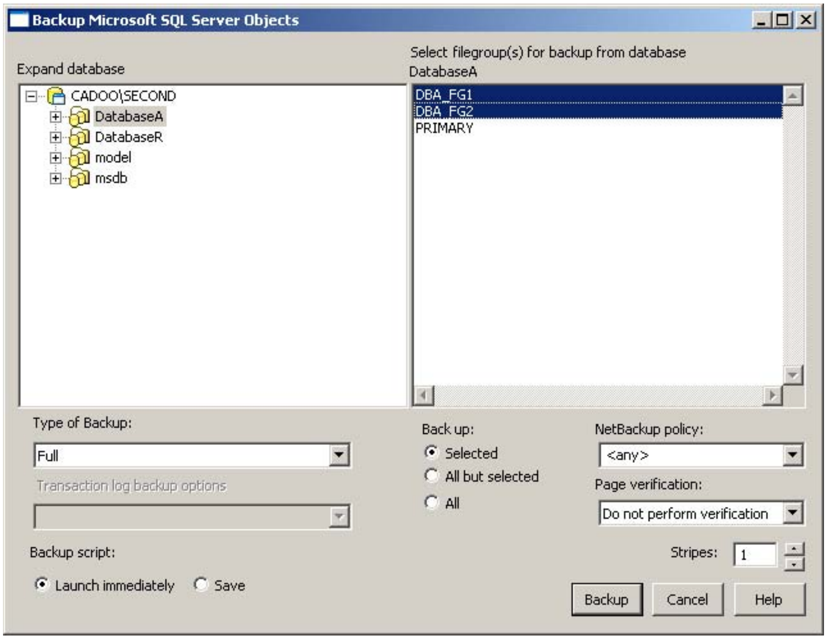
2. Choose **File > Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

3. In the left pane, expand the instance name.

4. Select a database whose filegroups you would like to back up.

5. In the right pane, select one or more filegroups that you want to back up.



Select filegroup(s) for backup from database DatabaseA

DBA_FG1
DBA_FG2
PRIMARY

Type of Backup: Full

Transaction log backup options

Back up: ● Selected ○ All but selected ○ All

NetBackup policy: <any>

Page verification: Do not perform verification

Backup script: ● Launch immediately ○ Save

Stripes: 1

6. Select the desired backup options.

   See "Backup Microsoft SQL Server Objects" on page 200 for a description of the other available backup options.

7. Click **Backup**.

   To view the progress of the backup, choose **File** > **View status**.

## Backing Up Read-Only Filegroups (SQL 2005)

In order to back up read-only filegroups you must first create a separate policy for this type of backup. Refer to "Reducing Backup Size and Time By Using Read-Only Filegroups (SQL Server 2005)" on page 45. You can verify that all read-only filegroups are backed up. Refer to the next section, "Viewing Read-Only Backup Sets."

### Viewing Read-Only Backup Sets

If you have made a decision to do periodic backups only on read-write filegroups, you will want to know whether you are currently retaining backups of the read-only filegroups.

▼ **To view read-only backup sets**

1. Browse for the backup images containing the read-only backup sets. See "Browsing Backup Images" on page 69.

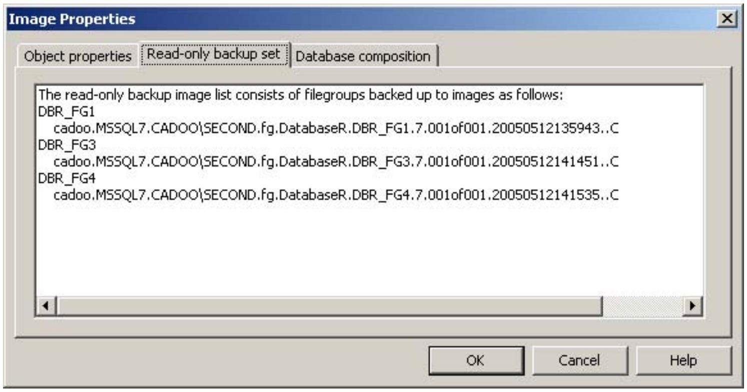
2. In the left pane, expand the instance name.

3. Right-click on the database and choose **Properties**.

4. Click the **Read-only backup set** tab.

   If the database does not contain read-only filegroups, then the message "This database does not contain any read-only filegroups." is shown. If backups do not exist for all of the read-only filegroups, then a list of the filegroups that were not backed up is shown. Finally, if a backup of all of the read-only filegroups is found, then the name of the latest image containing this backup is displayed.

The following image shows an example of a complete recovery set based on several read-only filegroups images.



```
The read-only backup image list consists of filegroups backed up to images as follows:
DBR_FG1
   cadoo.MSSQL7.CADOO\SECOND.fg.DatabaseR.DBR_FG1.7.001of001.20050512135943..C
DBR_FG3
   cadoo.MSSQL7.CADOO\SECOND.fg.DatabaseR.DBR_FG3.7.001of001.20050512141451..C
DBR_FG4
   cadoo.MSSQL7.CADOO\SECOND.fg.DatabaseR.DBR_FG4.7.001of001.20050512141535..C
```

5. If there are read-only filegroups that are not backed up, back them up as soon as possible to ensure you can perform a full recovery.

6. Click **OK**.

## Backing Up Read-Write Filegroups (SQL 2005)

### ▼ To back up read-write filegroups

1. To change the host and instance you wish to access, refer to “Selecting the SQL Host and Instance” on page 57.

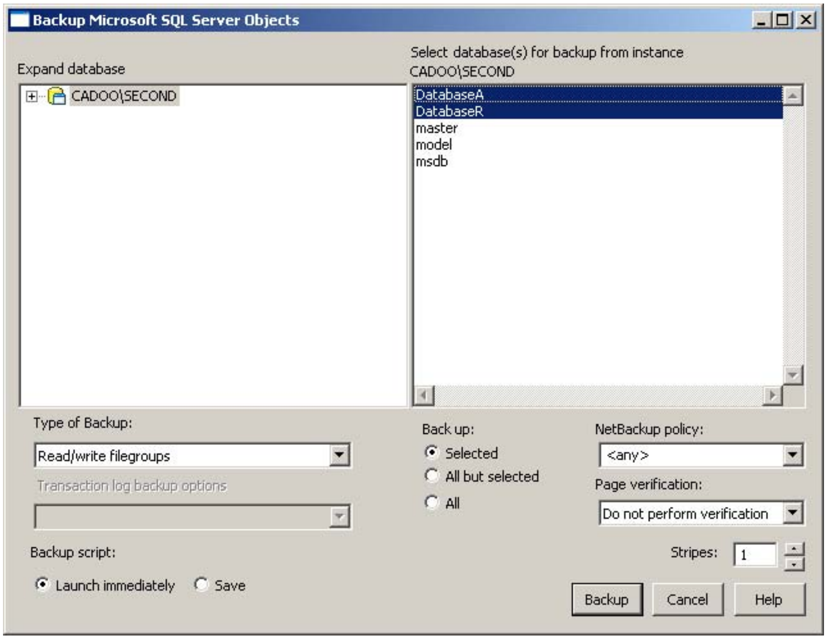
2. Choose **File > Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

3. In the right pane, select one or more databases that you want to back up.

4. Choose the **Backup type**.

    - To perform a full backup of the read-write filegroups, choose **Read/write filegroups**.
    - To perform a differential backup of the read-write filegroups, choose **Differential on read/write filegroups**.



Backup Microsoft SQL Server Objects

Expand database: CADOO\SECOND

Select database(s) for backup from instance CADOO\SECOND: DatabaseA, DatabaseR (selected); master; model; msdb

Type of Backup: Read/write filegroups

Transaction log backup options:

Back up: ● Selected ○ All but selected ○ All

NetBackup policy: <any>

Page verification: Do not perform verification

Backup script: ● Launch immediately ○ Save

Stripes: 1

5. Select the desired backup options.

    See “Backup Microsoft SQL Server Objects” on page 200 for a description of the other available backup options.

6. Click **Backup**.

    To view the progress of the backup, choose **File** > **View status**.

## Backing Up Database Files

### ▼ To back up a database file

1. To change the host and instance you wish to access, refer to “Selecting the SQL Host and Instance” on page 57.

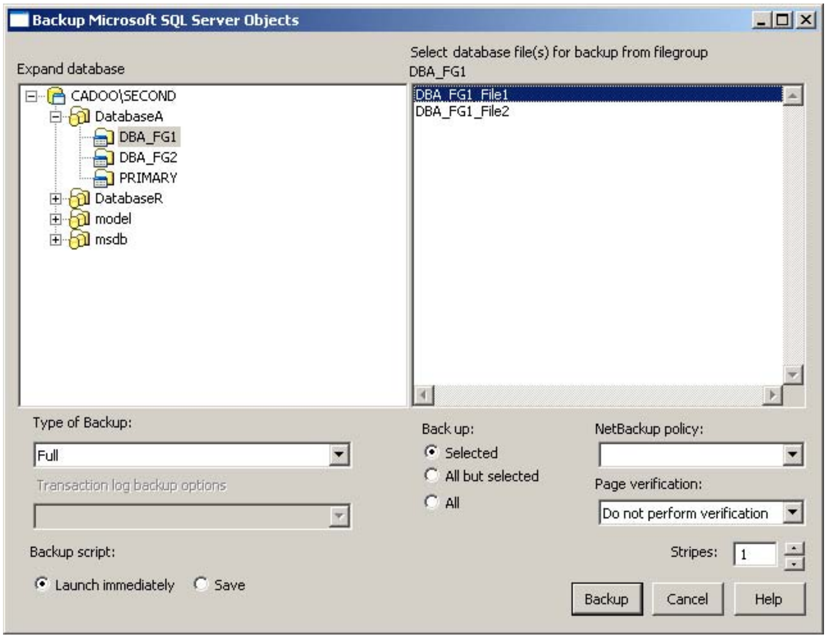
2. Choose **File > Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

3. In the left pane, expand the instance name and database.

4. Select the filegroup that contains the files you want to back up.

5. In the right pane, select one or more files that you want to back up.



Backup Microsoft SQL Server Objects

Expand database

- CADOO\SECOND
  - DatabaseA
    - DBA_FG1
    - DBA_FG2
    - PRIMARY
  - DatabaseR
  - model
  - msdb

Select database file(s) for backup from filegroup DBA_FG1

DBA_FG1_File1
DBA_FG1_File2

Type of Backup: Full

Transaction log backup options

Back up: ● Selected ○ All but selected ○ All

NetBackup policy:

Page verification: Do not perform verification

Stripes: 1

Backup script: ● Launch immediately ○ Save

6. Select the desired backup options.

   See “Backup Microsoft SQL Server Objects” on page 200 for a description of the other available backup options.

7. Click **Backup**.

   To view the progress of the backup, choose **File** > **View status**.

## Performing Partial Backups (SQL 2005)

### ▼ To perform a partial backup

1. To change the host and instance you wish to access, refer to "Selecting the SQL Host and Instance" on page 57.

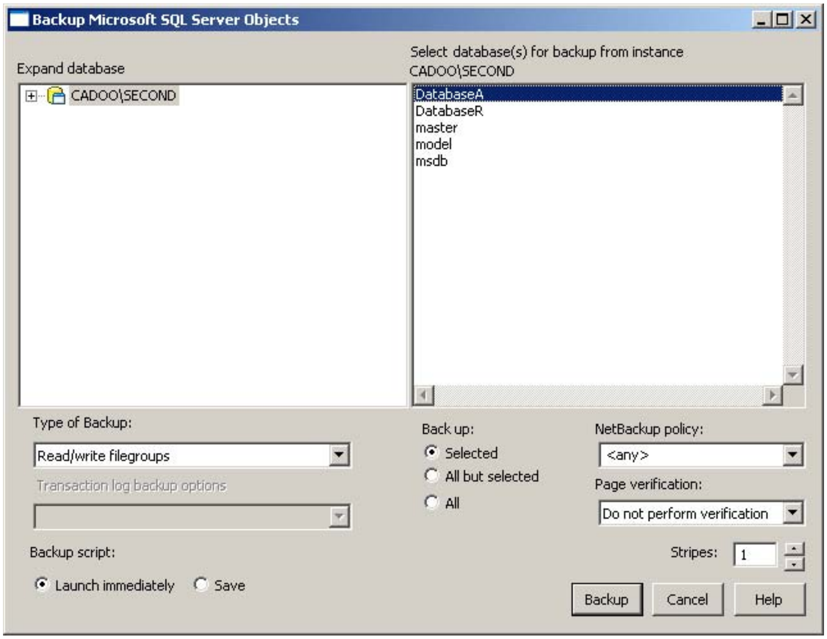
2. Choose **File > Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog is displayed.

3. In the right pane, select a database that you want to back up.

4. Choose the **Backup type**.

   - To perform a full backup, choose **Create a template for partial backup**.
   - To perform a differential backup of the read-write filegroups, choose **Create a template for partial differential backup**.



Backup Microsoft SQL Server Objects

Expand database

⊞ CADOO\SECOND

Select database(s) for backup from instance CADOO\SECOND

DatabaseA
DatabaseR
master
model
msdb

Type of Backup: Read/write filegroups

Transaction log backup options:

Back up: ● Selected ○ All but selected ○ All

NetBackup policy: <any>

Page verification: Do not perform verification

Stripes: 1

Backup script: ● Launch immediately ○ Save

5. Select the desired backup options.

   See "Backup Microsoft SQL Server Objects" on page 200 for a description of the other available backup options.

6. Click **Backup**.

   The Save Script As dialog displays.

7. Specify a file name and click **OK**.

8. When prompted to open the template, click **Yes**.

9. Edit the template by uncommenting the filegroups that you would like to include in the backup. You must  uncomment at least one filegroup.

   For example, replace:

```
#
#  If you wish to include filegroup DBA_FG1 in the partial backup,
#  then remove the hash mark that precedes the following line.
#FILEGROUP DBA_FG1
```

   with:

```
#
#  If you wish to include filegroup DBA_FG1 in the partial backup,
#  then remove the hash mark that precedes the following line.
FILEGROUP DBA_FG1
```

10. When you are finished modifying the template, save it.

11. To run the backup, choose **File** > **Manage script files**, select the script you created, and choose **Start**.

## Browsing Backup Images

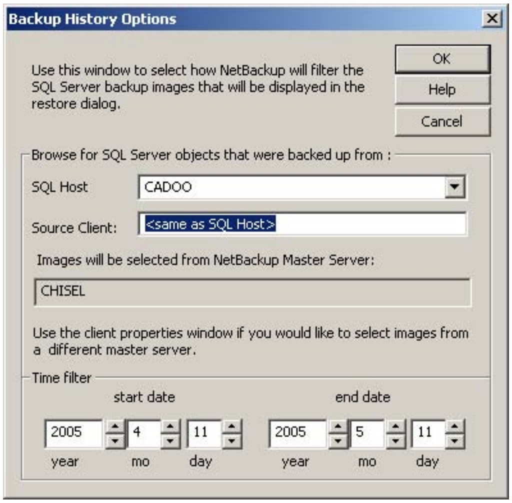
This section describes how to browse for available backup images. Once you have displayed the desired backup images, then follow the instructions for restoring a specific SQL object.

### ▼ To browse for backup images

1. To change the host and instance you wish to access, refer to “Selecting the SQL Host and Instance” on page 57.

2. Choose **File** > **Restore SQL Server Objects**.

   The Backup History Options dialog is displayed.



Backup History Options

Use this window to select how NetBackup will filter the SQL Server backup images that will be displayed in the restore dialog.

Browse for SQL Server objects that were backed up from :

SQL Host: CADOO

Source Client: <same as SQL Host>

Images will be selected from NetBackup Master Server:

CHISEL

Use the client properties window if you would like to select images from a different master server.

Time filter

start date: 2005 year, 4 mo, 11 day

end date: 2005 year, 5 mo, 11 day

3. Select the **SQL Host** whose backup images you would like to browse. (Refer to “Backup History Options” on page 198 for information on the **Source Client** setting.)

   This can also be manually edited if the host that you want does not appear in the drop-down list

4. Choose the date range to search.

5. Click **OK**.

6. Continue with the instructions for restoring the desired object
    - "Restoring a Database Backup" on page 70
    - "Staging a Full Database Recovery" on page 72
    - "Restoring Filegroup Backups" on page 73
    - "Restoring Read-Only Filegroups (SQL Server 2005)" on page 76
    - "Recovering a Database from Read-Write Backups" on page 75
    - "Restoring Database Files" on page 77
    - "Restoring a Transaction Log Image Without Staging a Full Recovery" on page 79
    - "Performing a Database Move" on page 80
    - "Performing Partial Database Restores (SQL Server 2000)" on page 83
    - "Performing Page-Level Restores (SQL 2005)" on page 86

## Restoring a Database Backup

This section describes how to restore a database from a full or differential database backup.

### ▼ To restore a database backup

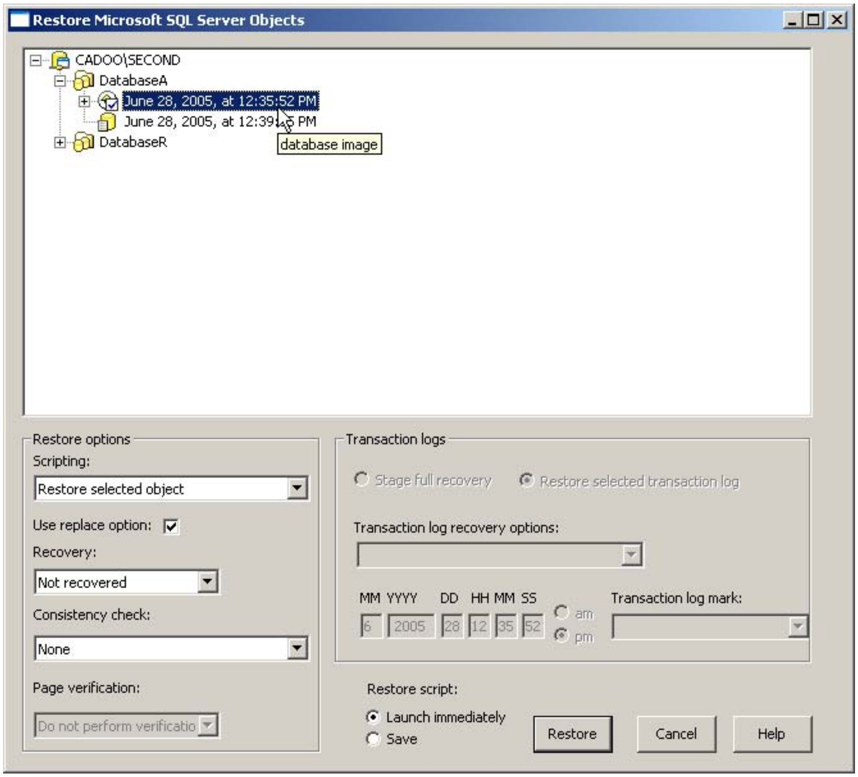
1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 69.

2. In the Restore dialog, expand the desired instance.

3. Expand the database.

4. Select the database image that you wish to restore.
    - To restore a full backup, select the image of the database backup.
    - To restore a full backup and a differential database backup, click the "+" sign and select a differential backup.

The full backup is automatically selected when you select a differential.



Restore Microsoft SQL Server Objects

CADOO\SECOND
DatabaseA
June 28, 2005, at 12:35:52 PM
June 28, 2005, at 12:39:[illegible] PM
DatabaseR
database image

Restore options
Scripting: Restore selected object
Use replace option: ☑
Recovery: Not recovered
Consistency check: None
Page verification: Do not perform verificatio

Transaction logs
○ Stage full recovery ● Restore selected transaction log
Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS |
|---|---|---|---|---|---|
| 6 | 2005 | 28 | 12 | 35 | 52 |

○ am
● pm
Transaction log mark:

Restore script:
● Launch immediately
○ Save

5. Select the desired restore options.

   If you wish to place the database in recovery mode so that it is immediately usable following the restore, then select **Recovered** from the **Recovery** list. However, be aware that after the database has been placed in recovered mode, you will not be able to update it with additional differential or transaction log backups.

   See "Restore Microsoft SQL Server Objects" on page 207 for a description of the other available restore options.

6. Click **Restore**.

   To view the progress of the restore, choose **File** > **View status**.

## Staging a Full Database Recovery

### ▼ To stage a full database recovery

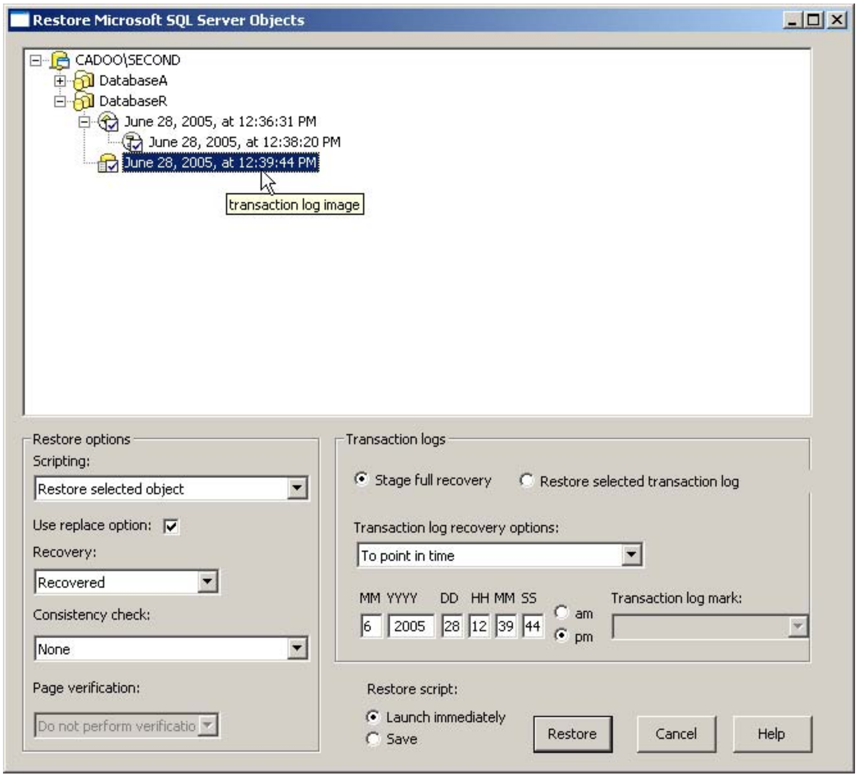
1. Browse for a backup image that contains the point in time to which you would like to recover.

   See "Browsing Backup Images" on page 69.

2. In the Restore dialog, expand the desired instance.

3. Click the "+" next to the database containing the transaction log backup you wish to restore.

4. Select the transaction log image that includes the point in time that you would like to recover to.



Restore Microsoft SQL Server Objects

- CADOO\SECOND
  - DatabaseA
  - DatabaseR
    - June 28, 2005, at 12:36:31 PM
      - June 28, 2005, at 12:38:20 PM
    - June 28, 2005, at 12:39:44 PM

transaction log image

Restore options

Scripting: Restore selected object

Use replace option: ☑

Recovery: Recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

● Stage full recovery ○ Restore selected transaction log

Transaction log recovery options: To point in time

MM 6 YYYY 2005 DD 28 HH 12 MM 39 SS 44 ○ am ● pm

Transaction log mark:

Restore script:

● Launch immediately

○ Save

5. Select **Stage full recovery**.

   **Stage full recovery** is enabled if a set of images exists that includes the transaction log image and that are adequate for staging a full database recovery. Also, when viewing the properties of the transaction log, a **Recovery Set** tab is displayed.

**Note** The recovery set may include any combination of backup images that are sufficient for staging the full recovery. These may include full database, filegroup, differentials, etc.

6. Click **Restore**.

   To view the progress of the restore, choose **File** > **View status**.

## Restoring Filegroup Backups

If you are periodically backing up only read-write filegroups as a part of your backup strategy, refer to "Recovering a Database from Read-Write Backups" on page 75. For information on restoring read-only filegroups, refer to "Restoring Read-Only Filegroups (SQL Server 2005)" on page 76

**Caution** If you attempt to restore a single differential backup without first restoring the preceding database backup file, SQL Server will halt the load process with an error such as 4305 or 4306. If you plan to restore a single differential, then you are responsible for first restoring the database backup file. You can avoid this problem by backing up the entire sequence of transaction logs, the differential backup, and the backup file to the same NetBackup Server and restoring the entire sequence of backup objects as described in "Staging a Full Database Recovery" on page 72.

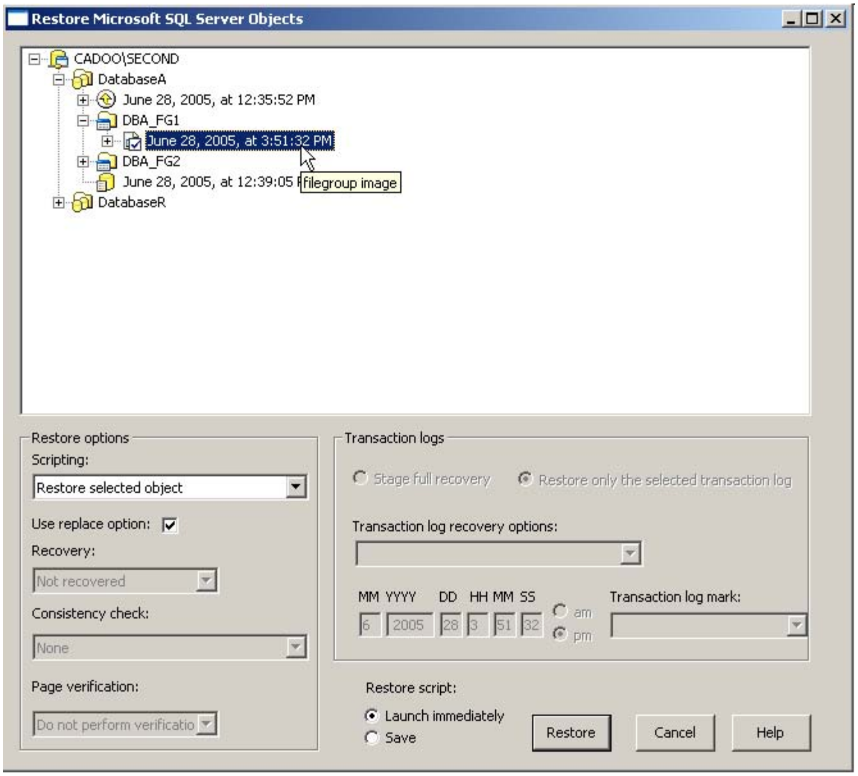
### ▼ To restore a filegroup

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 69.

2. In the Restore dialog, expand the desired instance and database.

3. Expand the filegroup and select a filegroup image to restore.

   - To restore a full backup, select the image of the filegroup backup.
   - To restore a differential filegroup backup, click the "+" next to the full backup and select the differential backup.



CADOO\SECOND
DatabaseA
June 28, 2005, at 12:35:52 PM
DBA_FG1
June 28, 2005, at 3:51:32 PM
DBA_FG2
June 28, 2005, at 12:39:05 P
filegroup image
DatabaseR

Restore options
Scripting: Restore selected object
Use replace option: ☑
Recovery: Not recovered
Consistency check: None
Page verification: Do not perform verificatio

Transaction logs
○ Stage full recovery ● Restore only the selected transaction log
Transaction log recovery options:
MM YYYY DD HH MM SS: 6 2005 28 3 51 32 ○ am ● pm
Transaction log mark:

Restore script:
● Launch immediately
○ Save

4. Select the desired restore options.

   See "Restore Microsoft SQL Server Objects" on page 207 for a description of the available restore options.

5. Click **Restore**.

   To view the progress of the restore, choose **File** > **View status**.

## Recovering a Database from Read-Write Backups

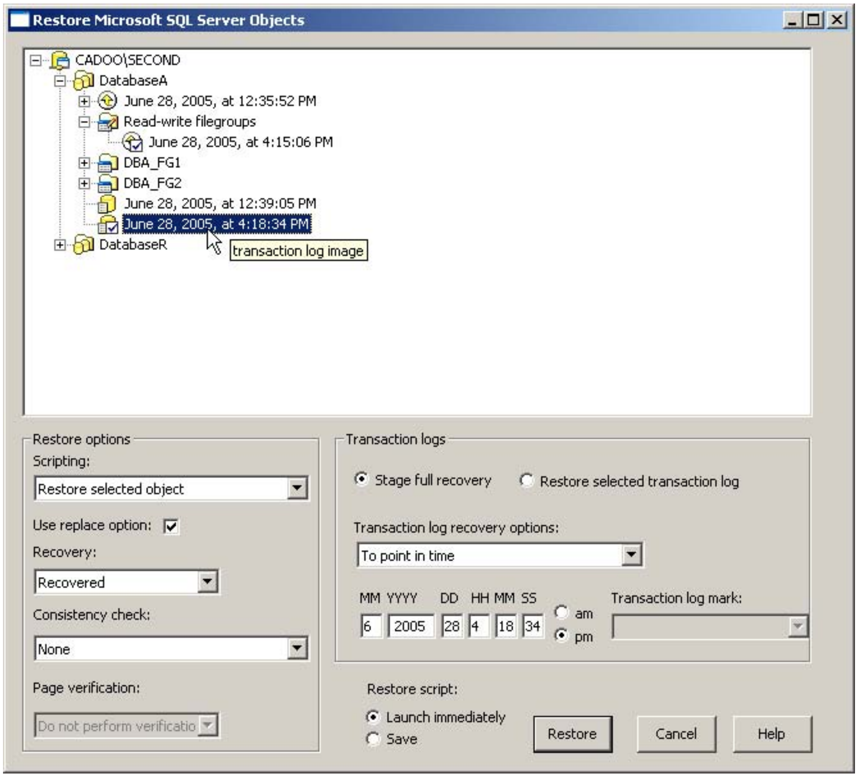
NetBackup for SQL Server automatically generates the most efficient recovery path when you select a transaction log image for restore. The recovery path may be based on read-write filegroups if you use them in your backup strategy. After restoring the read-write filegroups, you can bring the database online, without having to restore the read-only filegroups provided they are not damaged.

### ▼ To restore databases with read-write filegroups

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 69.

2. In the Restore dialog, expand the desired instance.

3. Expand the database containing the read-write filegroups you wish to restore.



Restore Microsoft SQL Server Objects

```
CADOO\SECOND
  DatabaseA
    June 28, 2005, at 12:35:52 PM
    Read-write filegroups
      June 28, 2005, at 4:15:06 PM
    DBA_FG1
    DBA_FG2
    June 28, 2005, at 12:39:05 PM
    June 28, 2005, at 4:18:34 PM
  DatabaseR
```

transaction log image

Restore options
Scripting: Restore selected object
Use replace option: ☑
Recovery: Recovered
Consistency check: None
Page verification: Do not perform verificatio

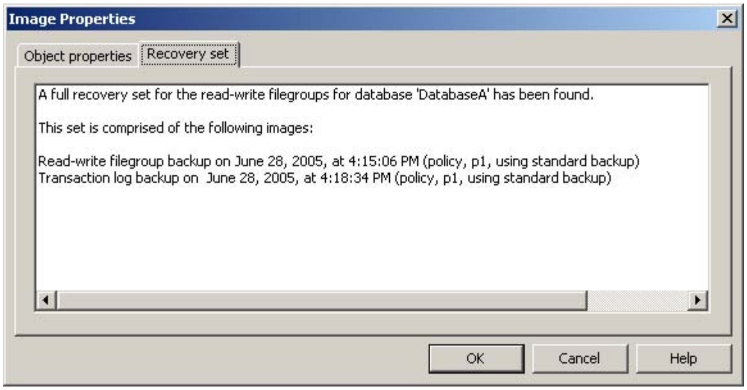
Transaction logs
● Stage full recovery ○ Restore selected transaction log
Transaction log recovery options: To point in time
MM 6 YYYY 2005 DD 28 HH 4 MM 18 SS 34 ○ am ● pm
Transaction log mark:

Restore script:
● Launch immediately
○ Save

4. Right-click on the transaction log backup and choose **Properties**.

5. On the **Recovery set** tab, verify a complete backup set is available.

   The following image shows an example of a recovery set based on a read-write filegroups image and a read-write filegroups differential image.

   A full recovery set for the read-write filegroups for database 'DatabaseA' has been found.

   This set is comprised of the following images:

   Read-write filegroup backup on June 28, 2005, at 4:15:06 PM (policy, p1, using standard backup)
   Transaction log backup on  June 28, 2005, at 4:18:34 PM (policy, p1, using standard backup)

6. Click **OK**.

7. To begin the database restore, click **Restore**.

   After the restore has completed the database will be back online. However, you will not be able to recover the read-only filegroups until they are restored. To restore the read-only filegroups, refer to "Restoring Read-Only Filegroups (SQL Server 2005)" on page 76

## Restoring Read-Only Filegroups (SQL Server 2005)

### ▼ To restore read-only filegroups in a database

1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 69.

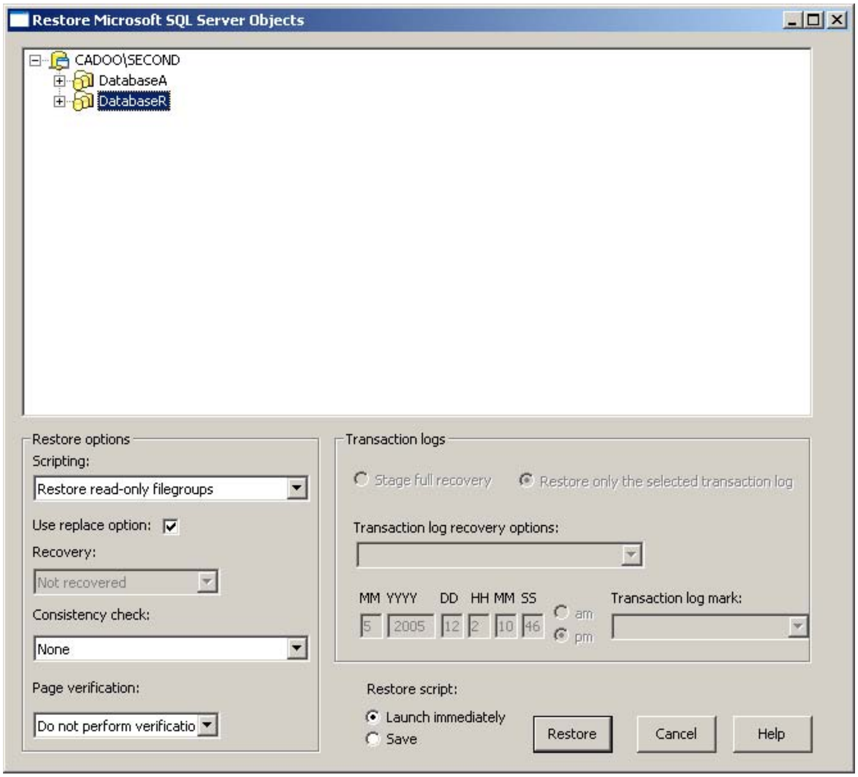
   Be sure that the start date for the Time Filter is early enough to include the timestamp of the earliest backup of the read-only filegroups.

2. In the Restore dialog, expand the desired instance.

3. Select the database containing the read-only filegroups you wish to restore.

   In the **Scripting** list, **Restore read-only filegroups** is selected.

   If the **Restore** button is enabled, then backup images of the full set of read-only filegroups are available.



Restore Microsoft SQL Server Objects

- CADOO\SECOND
  - DatabaseA
  - DatabaseR

Restore options

Scripting: Restore read-only filegroups

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore only the selected transaction log

Transaction log recovery options:

MM YYYY DD HH MM SS: 5 2005 12 2 10 46 ○ am ● pm

Transaction log mark:

Restore script:

● Launch immediately

○ Save

4. Click **Restore**.

   To view the progress of the restore, choose **File** > **View status**.

## Restoring Database Files

### ▼ To restore a database file

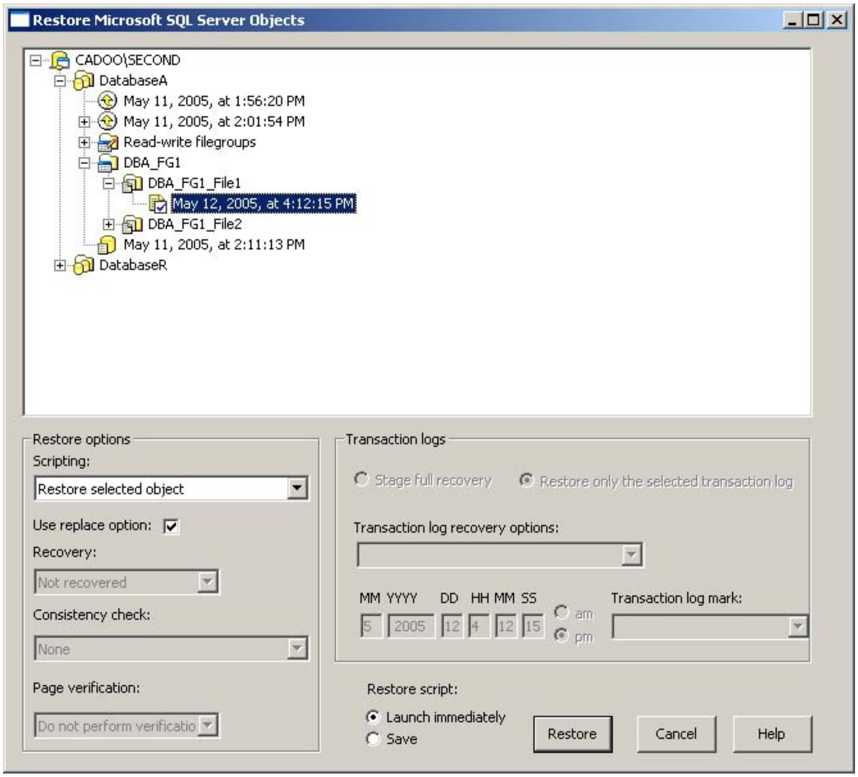
1. Browse for the backup images you wish to restore. See "Browsing Backup Images" on page 69.

2. In the Restore dialog, expand the desired instance and database.

3. Expand the filegroup which contains the file you would like to restore.

4. Expand the file.

5. Select the database file image that you want to restore.



Restore Microsoft SQL Server Objects

```
CADOO\SECOND
  DatabaseA
    May 11, 2005, at 1:56:20 PM
    May 11, 2005, at 2:01:54 PM
    Read-write filegroups
    DBA_FG1
      DBA_FG1_File1
        May 12, 2005, at 4:12:15 PM
      DBA_FG1_File2
    May 11, 2005, at 2:11:13 PM
  DatabaseR
```

Restore options

Scripting: Restore selected object

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore only the selected transaction log

Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS |  |
|---|---|---|---|---|---|---|
| 5 | 2005 | 12 | 4 | 12 | 15 | ○ am<br>● pm |

Transaction log mark:

Restore script:

● Launch immediately

○ Save

6. Select the desired restore options.

   See “Restore Microsoft SQL Server Objects” on page 207 for a description of the available options.

7. Click **Restore**.

   To view the progress of the restore, choose **File** > **View status**.

# Restoring a Transaction Log Image Without Staging a Full Recovery

## ▼ To restore a transaction log without staging a full recovery

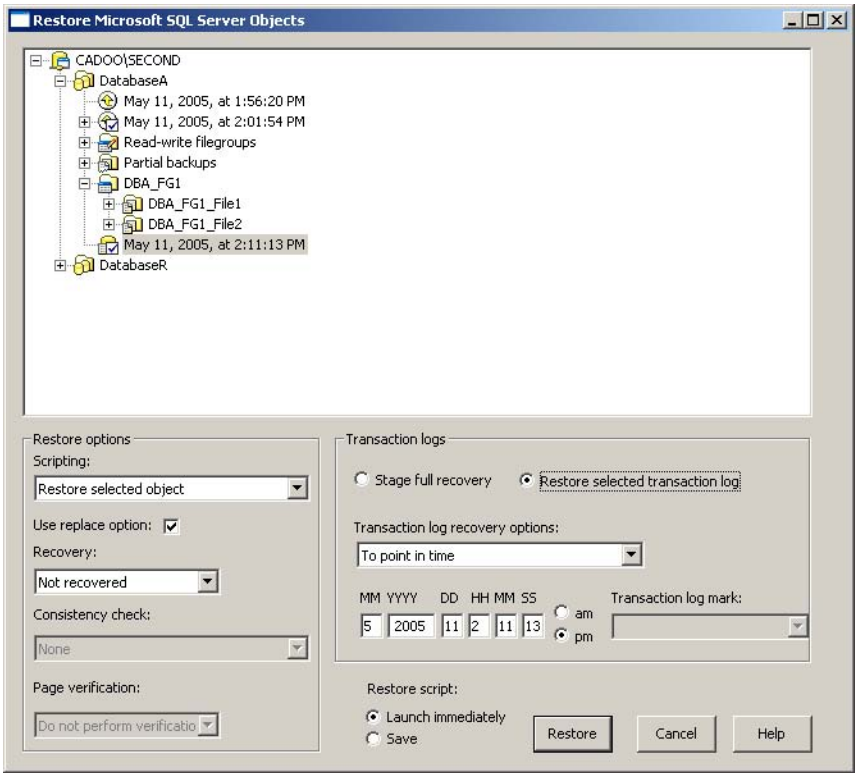
1. Browse for the backup images you wish to restore. See “Browsing Backup Images” on page 69.

2. In the Restore dialog, expand the desired instance.

3. Select the transaction log image that you want to restore.

   If a set of images exists that include the transaction log image and that are sufficient for staging a full database recovery, then **Stage full recovery** is enabled. Also, when viewing the properties of the transaction log, a **Recovery Set** tab is displayed.

4. Select **Restore only the transaction log that you selected**.



Restore Microsoft SQL Server Objects

```
CADOO\SECOND
  DatabaseA
    May 11, 2005, at 1:56:20 PM
    May 11, 2005, at 2:01:54 PM
    Read-write filegroups
    Partial backups
    DBA_FG1
      DBA_FG1_File1
      DBA_FG1_File2
    May 11, 2005, at 2:11:13 PM
  DatabaseR
```

Restore options

Scripting: Restore selected object

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore selected transaction log

Transaction log recovery options: To point in time

| MM | YYYY | DD | HH | MM | SS | | Transaction log mark: |
|---|---|---|---|---|---|---|---|
| 5 | 2005 | 11 | 2 | 11 | 13 | ○ am<br>● pm | |

Restore script:

● Launch immediately

○ Save

5. Click **Restore**.

   To view the progress of the restore, choose **File** > **View status**.

## Performing a Database Move

A database move allows you to use a full set of backup images to copy an existing database to a new database having a different name. Database move operations can only be carried out when your selection includes a database image. This can occur either when you directly select the database backup image, or when NetBackup finds a recovery set which contains a database backup image.

In the following example the move script is created upon selection of a database backup image.

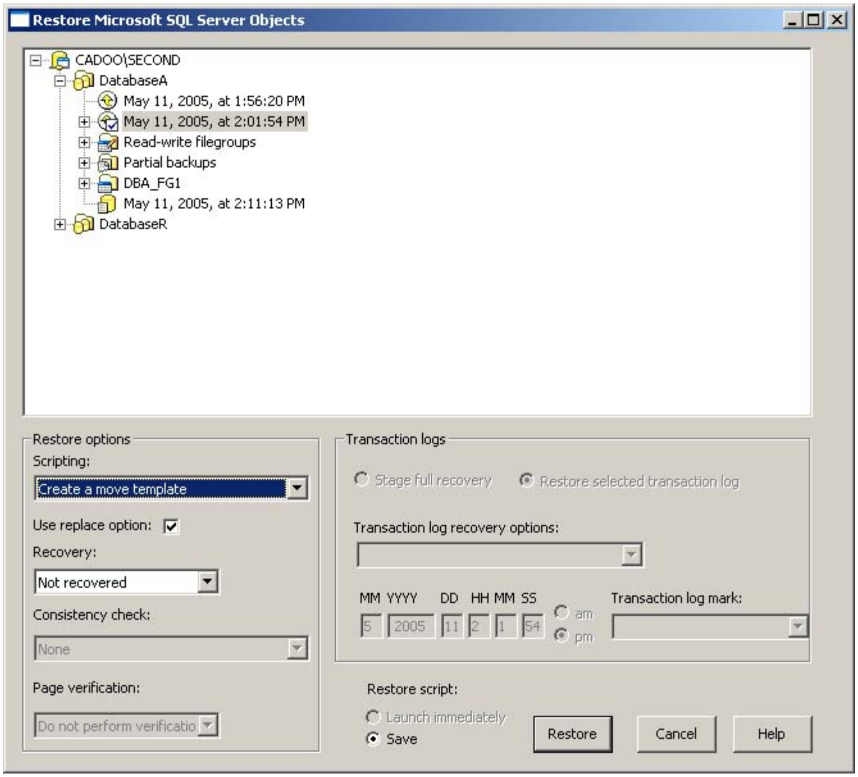
### ▼ To perform a database move

1. Browse for the backup images you wish to restore. See “Browsing Backup Images” on page 69.

2. In the Restore dialog, expand the desired instance.

3. Select the database backup image that you want to restore.

4. From the **Scripting options** list, select **Create a move template**.



Restore Microsoft SQL Server Objects

- CADOO\SECOND
  - DatabaseA
    - May 11, 2005, at 1:56:20 PM
    - May 11, 2005, at 2:01:54 PM
    - Read-write filegroups
    - Partial backups
    - DBA_FG1
    - May 11, 2005, at 2:11:13 PM
  - DatabaseR

Restore options

Scripting: Create a move template

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore selected transaction log

Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS |
|---|---|---|---|---|---|
| 5 | 2005 | 11 | 2 | 1 | 54 |

○ am ● pm

Transaction log mark:

Restore script:

○ Launch immediately
● Save

**Note** When you choose to create a move or partial database restore script, the capability to perform an immediate launch is disabled because you will need to edit the script in order to specify certain destination parameters.

5. Select the desired restore options.

   See "Restore Microsoft SQL Server Objects" on page 207 for a description of the available options.

6. Click **Restore**.

7. Indicate a file name and click **Save**.

   The Save Script As dialog is displayed.

8. Click **Yes** to open the template in Notepad.

9. Change the database name in the template to the name of the database to restore to.

   For example, replace:

```
#  Replace the database name in the following line with the name of the database that you
#  want to move to. Also remove the hash mark <#> which precedes the keyword <DATABASE>.
#
# DATABASE "DatabaseA"
```

   with:

```
#  Replace the database name in the following line with the name of the database that you
#  want to move to. Also remove the hash mark <#> which precedes the keyword <DATABASE>.
#
DATABASE "DatabaseB"
```

10. Change the path for the database files that you would like to restore.

    You must  uncomment at least one file. For example, replace:

```
#  Replace the file path <C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DBA_FG1_File1.ndf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "DBA_FG1_File1".
MOVE  "DBA_FG1_File1"
#TO  "C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DBA_FG1_File1.ndf"
```

    with:

```
#  Replace the file path <C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DBA_FG1_File1.ndf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "DBA_FG1_File1".
MOVE  "DBA_FG1_File1"
TO  "C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DBB_FG1_File1.ndf"
```

11. Change the database file path.

    For example, replace:

```
#  Replace the file path <C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DatabaseA.mdf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "DatabaseA".
MOVE  "DatabaseA"
#TO  "C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DatabaseA.mdf"
```

    with:

```
#  Replace the file path <C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DatabaseA.mdf>
#  with a new file path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "DatabaseA".
MOVE  "DatabaseA"
TO  "C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DatabaseB.mdf"
```

12. Make similar changes to the template for any differential or transaction log backups you wish to move.

13. When you are finished modifying the template, save it.

14. To run the restore, choose **File** > **Manage script files**, select the script you created, and choose **Start**.

15. Click **Yes** to launch the restore.

    To view the progress of the restore, choose **File** > **View status**.

## Performing Partial Database Restores (SQL Server 2000)

The following instructions show you how to perform a partial database restore on a SQL Server 2000 database. A partial database restore lets you select individual filegroup components of a database and restore them to a new database having a different name.

**Tip** When you are preparing your database design, map logically related database components such as tables, indexes, keys, etc., to the same filegroup. This will enable you to use the partial database restore procedure, which follows, to recover these logically related components as a single unit.

### ▼ To perform a partial database restore

1. Browse for the backup images you wish to restore. See “Browsing Backup Images” on page 69.

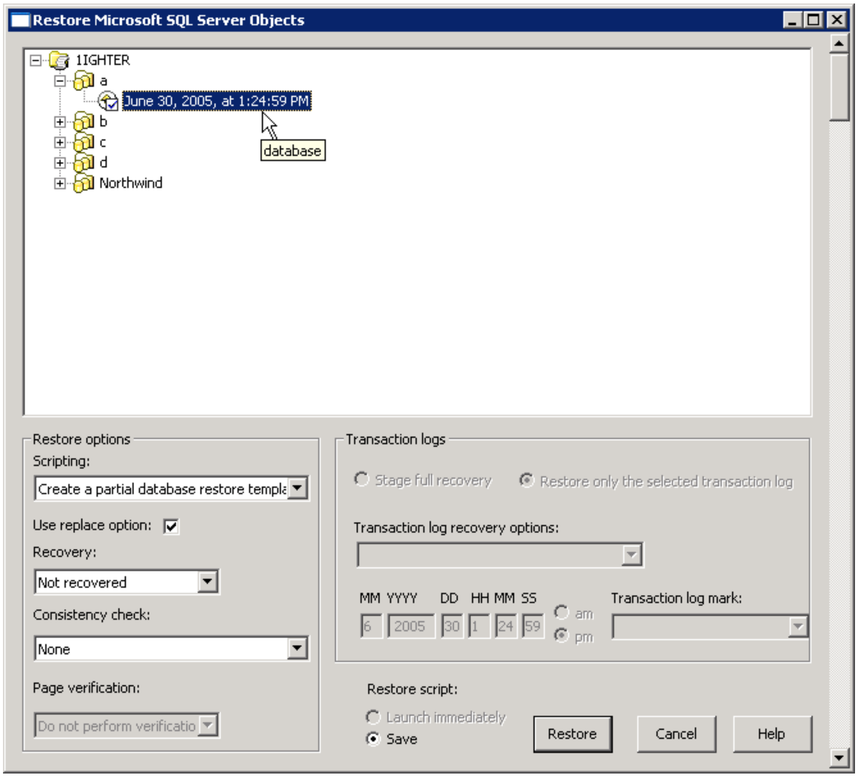
2. In the Restore dialog, expand the desired instance.

3. Expand the database.

4. Select the database image that you want to restore.

5. From the **Scripting options** list, select **Create a partial database restore template**.

**Note** When you choose to create a move or partial database restore script, the capability to perform an immediate launch is disabled because you will need to edit the script in order to specify certain destination parameters.



Restore Microsoft SQL Server Objects

- 1IGHTER
  - a
    - June 30, 2005, at 1:24:59 PM
  - b
  - c
  - d
  - Northwind

Restore options

Scripting: Create a partial database restore templa

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore only the selected transaction log

Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS |
|---|---|---|---|---|---|
| 6 | 2005 | 30 | 1 | 24 | 59 |

○ am ● pm

Transaction log mark:

Restore script:

○ Launch immediately

● Save

**6.** Click **Restore**.

The Save Script As dialog is displayed.

**7.** Type the name of a file to which you want NetBackup to write the template and click **Save**.

**8.** Click **Yes** to open the template in Notepad.

9. Replace the database name in the template with the name of the target database of the restore.

   For example, replace:

```
#  Replace the database name in the following line with the name of the database that you
#  want as the target of the partial restore. Also remove the hash mark <#> which precedes
#  the keyword <DATABASE>.
#DATABASE "DatabaseA"
```

   with:

```
#  Replace the database name in the following line with the name of the database that you
#  want as the target of the partial restore. Also remove the hash mark <#> which precedes
#  the keyword <DATABASE>.
DATABASE "DatabaseB"
```

10. To restore a file, indicate a new file path and uncomment the MOVE and TO lines.

    For example, replace:

```
#  If you wish to restore file <C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.mdf>,
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
#MOVE  "DBA_FG1_File1"
#TO  "C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DBA_FG1_File1.ndf"
```

    with:

```
#  If you wish to restore file <C:\Program Files\Microsoft SQL Server\MSSQL\data\Sample2.mdf>,
#  then replace this path with a new file path. Also remove the hash marks <#>
#  which precede the keywords <MOVE> and <TO>.
MOVE  "DBA_FG1_File1"
TO  "C:\Microsoft SQL Server\MSSQL.3\MSSQL\DATA\DBB_FG1_File1.ndf"
```

11. Make similar changes to the template for any differential or transaction log backups you wish to restore.

12. When you are finished modifying the template, save it.

13. To run the restore, choose **File** > **Manage script files**, select the script you created, and choose **Start**.

## Performing Page-Level Restores (SQL 2005)

If a portion of a SQL Server 2005 database has been corrupted due to hardware failure, you may be able to use page-level restore to recover only the pages that were corrupted. Page-level restore can reduce the total downtime, if you only need to restore a relatively small number of pages. If many pages are corrupt, then a full database recovery may be faster.

When you choose the page restore option, NetBackup for SQL Server creates a page restore template. This template includes the following parts:

- A page restore operation that you will modify by inserting the IDs of the pages that you wish to restore.
- A series of transaction log images, if available, for recovering the database to the current point in time.
- A tail-log backup and recovery operation, which is required in order to bring the database on-line.

### Requirements and Limitations

- Pages can be restored from the following backup types: database, filegroup, file, read-write filegroups, partial database.
- Your SQL Server 2005 database must use either the full or bulk-logged recovery model in order for page-level restore to be available to you.
- SQL Server sometimes cannot recover the specific pages that you request if they contain critical information about the definition of the database itself. For example, you cannot use page-level restore for the first page in a database file. When you detect that page-level restore will not work, then you will need to resort to using full database recovery.
- A maximum of 1000 pages can be recovered from a backup image through a page-level restore.

### ▼ To perform a page-level restore

1. Obtain a list of corrupt pages in the database.

   *SQL Server 2005 Books Online* suggests several methods for obtaining a list of corrupt pages. One of these methods is to run the command `DBCC checkdb` from the SQL Server 2005 Management Studio.

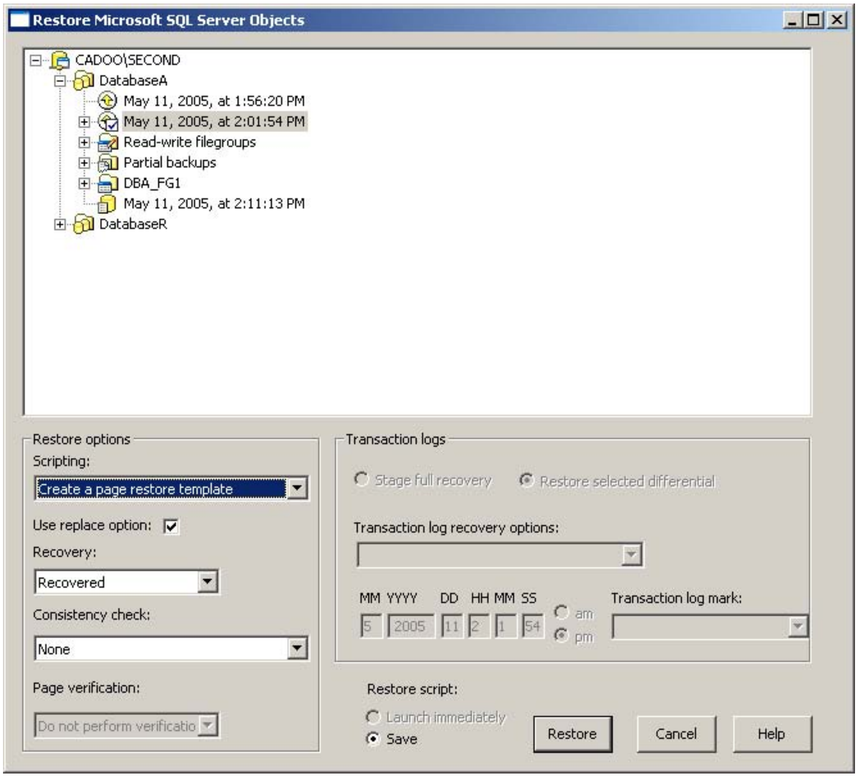
2. Browse for the backup images you wish to restore. See “Browsing Backup Images” on page 69.

3. In the Restore dialog, expand the desired instance.

4. Expand the database.

5. Select the database backup image that contains pages that you want to restore.

6. From the **Scripting options** list, select **Create a page restore template**.



Restore Microsoft SQL Server Objects

- CADOO\SECOND
  - DatabaseA
    - May 11, 2005, at 1:56:20 PM
    - May 11, 2005, at 2:01:54 PM
    - Read-write filegroups
    - Partial backups
    - DBA_FG1
    - May 11, 2005, at 2:11:13 PM
  - DatabaseR

Restore options

Scripting: Create a page restore template

Use replace option: ☑

Recovery: Recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore selected differential

Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS | |
|---|---|---|---|---|---|---|
| 5 | 2005 | 11 | 2 | 1 | 54 | ○ am<br>● pm |

Transaction log mark:

Restore script:

○ Launch immediately

● Save

7. Click **Restore**.

   The Save Script As dialog is displayed.

8. Type a file name for the page restore script and click **Save**.

9. Click **Yes** to open the template in Notepad.

10. Edit the page first operation the page ids that you wish to replace.

    For example, replace:

```
#
# Create one or more page restore requests. These use the following format
#PAGE file-id:page-id
```

    with

```
#
# Create one or more page restore requests. These use the following format
PAGE 1:14
PAGE 1:20
```

11. When you are finished modifying the template, save it.

12. To run the restore, choose **File** > **Manage script files**, select the script you created, and choose **Start**.

13. Click **Yes** to start the restore.

## Redirecting a Restore to a Different Host

Multiple NetBackup database clients may be configured to use a single NetBackup master server for backing up SQL Server databases. With this configuration, you can back up SQL Server database objects from one client and restore them to another. In the following procedure, *HostA* is the client that was originally backed up and *HostB* is the other client to which the restore will be redirected.

### ▼ To redirect a restore to a different host

1. Establish permission settings on the master server. Do one of the following:

    ◆ Create a file called `install_path\NetBackup\db\altnames\No.Restrictions`, to allow unrestricted redirected restore privileges

    *OR*

    ◆ Create a file called `install_path\NetBackup\db\altnames\HostB`, to allow HostB to restore HostA's data. Note that this allows the client named HostB to access HostA's data on the master as well as any other client's data that was backed up on the master.

2. Choose **File** > **Set SQL Server connection properties**.

3. Log onto the target instance on HostB.

4. Choose **File** > **Set NetBackup client properties** and select the current master server to be a master server which is common to both HostA and HostB.

5. Choose **File** > **Restore SQL Server Objects**.

6. From the Backup History Options dialog, select HostA as the SQL host from which the target objects were backed up.

   When the restore window is displayed it will contain a history of objects that were backed up from HostA. You can then select those objects and restore them to HostB.

## Redirecting a Database to a Different Location on a Different Host

The database move redirects the restore of a database to a different location. The new location may be a different instance on the same host, a different host, or a different file path. The move operation also allows you to restore the database under a different name than the original one.

**Note** The destination host\instance of a move or restore operation is the one that you log into as indicated by the SQL Server connection properties dialog. The source (or browse) host\instance for move or restore operations is designated using the Backup History Options dialog, which comes up automatically when you choose **File** > **Restore SQL Server Objects**.

### ▼ To redirect a database to another location on a different host

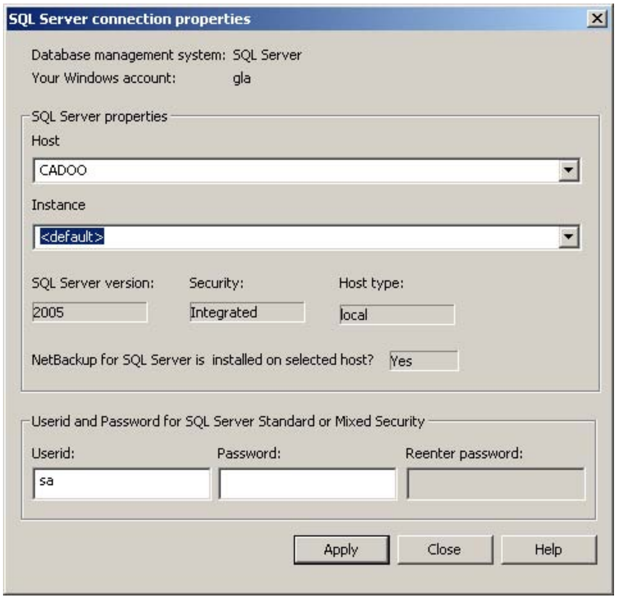
1. The server which backed up the database you wish to restore must appear in the server list of the destination host. If it does not, see “Selecting a Server” on page 94.

2. Choose **File > Set SQL Server connection properties**.

3. From the **Host** list, select the host you wish to restore to.

4. From the **Instance** list, select the desired instance.

To choose the default instance, either select **<default>** or leave the field empty.



SQL Server connection properties

Database management system: SQL Server
Your Windows account: gla

SQL Server properties

Host: CADOO

Instance: <default>

SQL Server version: 2005
Security: Integrated
Host type: local

NetBackup for SQL Server is installed on selected host? Yes

Userid and Password for SQL Server Standard or Mixed Security

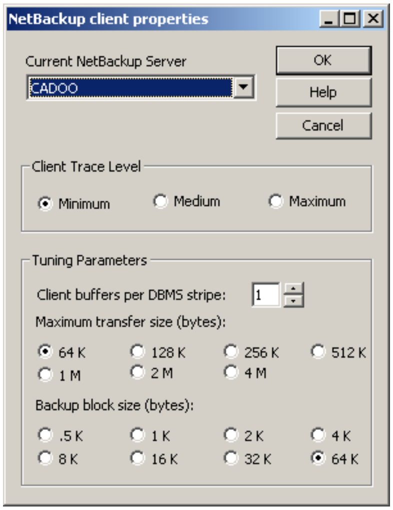
Userid: sa
Password:
Reenter password:

5. Click **Apply** and then **Close**.

6. Choose **File** > **Set NetBackup client properties**.

   The NetBackup client properties dialog is displayed.

7. From the **Current Netbackup Server** list, select the NetBackup master server which contains the SQL Server backup images that you want to restore on the destination host.



Current NetBackup Server: CADOO

Client Trace Level

● Minimum ○ Medium ○ Maximum

Tuning Parameters

Client buffers per DBMS stripe: 1

Maximum transfer size (bytes):

● 64 K ○ 128 K ○ 256 K ○ 512 K
○ 1 M ○ 2 M ○ 4 M

Backup block size (bytes):

○ .5 K ○ 1 K ○ 2 K ○ 4 K
○ 8 K ○ 16 K ○ 32 K ● 64 K

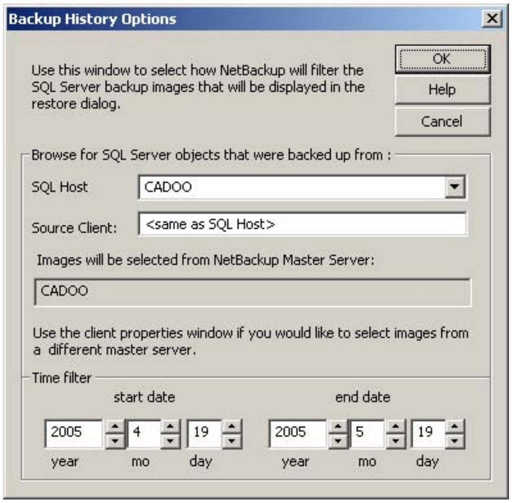
8. Click **OK**.

9. Choose **File** > **Restore Backup SQL Server Objects**.

The Backup History Options dialog is displayed.

10. In the **SQL Host** list, select the host which has the database you wish to restore.

    (Refer to "Backup History Options" on page 198 for information on the **Source Client** setting, if necessary.)



Backup History Options

Use this window to select how NetBackup will filter the SQL Server backup images that will be displayed in the restore dialog.

Browse for SQL Server objects that were backed up from :

SQL Host: CADOO

Source Client: <same as SQL Host>

Images will be selected from NetBackup Master Server: CADOO

Use the client properties window if you would like to select images from a different master server.

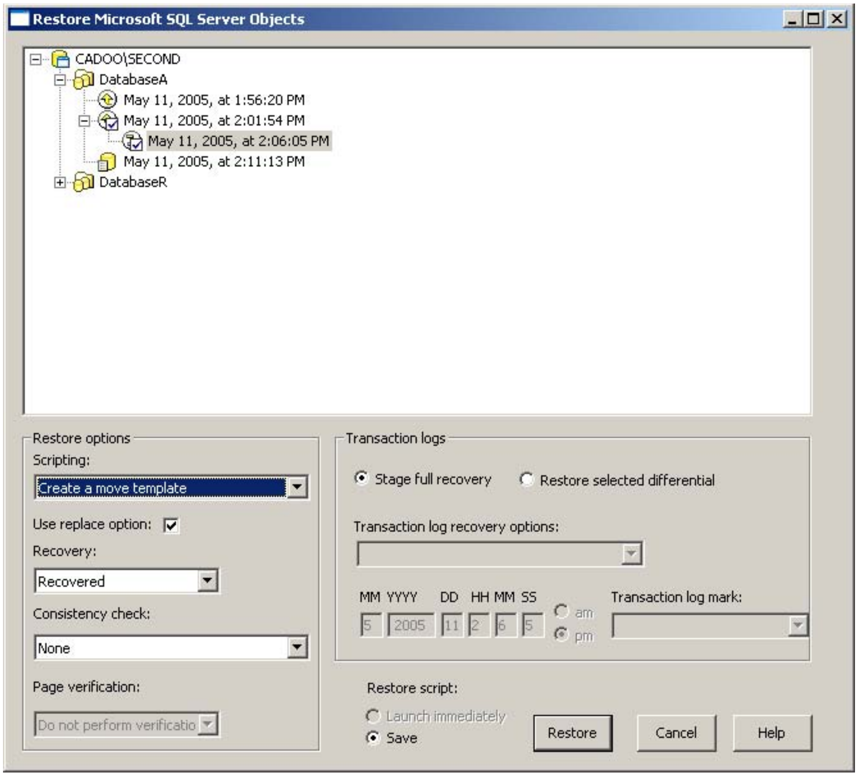
Time filter

| | start date | | | end date | |
|---|---|---|---|---|---|
| 2005 | 4 | 19 | 2005 | 5 | 19 |
| year | mo | day | year | mo | day |

11. Click **OK**.

12. Browse for the database that you want to move.

13. From the **Scripting options** list, select **Create a move template**.



CADOO\SECOND
DatabaseA
May 11, 2005, at 1:56:20 PM
May 11, 2005, at 2:01:54 PM
May 11, 2005, at 2:06:05 PM
May 11, 2005, at 2:11:13 PM
DatabaseR

Restore options

Scripting: Create a move template

Use replace option: ☑

Recovery: Recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

● Stage full recovery ○ Restore selected differential

Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS |
|---|---|---|---|---|---|
| 5 | 2005 | 11 | 2 | 6 | 5 |

○ am ● pm

Transaction log mark:

Restore script:

○ Launch immediately
● Save

14. Click **Restore**.

    A Save As dialog is displayed.

15. Enter a file name, and click **Save**.

16. Click **Yes** to open the template.

17. Edit this template to designate the name that you would like to use for the destination database as well as the file paths that you would like to use for each of the database files.

18. After you have edited the template, you can launch it by choosing **File** > **Manage script files**.

## Selecting a Server

When you perform a move, the backup images must be available on the host machine which is acting as the NetBackup master server for the destination host. If this server is contained in the server list of the destination host, then you can select the current master server by choosing **File** > **Set NetBackup client properties**.

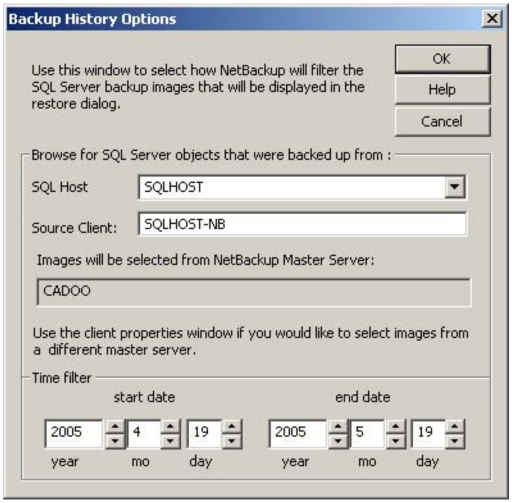
If the server is not in the server list of the destination host—perhaps, because the server machine is remote or has access limitations—then you must duplicate the images onto removable media (with a unique id), transport that media to the master server used by the destination host, and import the images to that server. After the images have been imported, continue with the instructions for performing a move.

# Performing Restores with a Multi-Nic Connection

To perform restores in multi-NIC environment, you need to specify the public name of the SQL Server host in the SQL Server connection properties dialog. You also need to provide both the SQL Server host name and the private interface name in the Browse History Options dialog.

### ▼ To perform restores

1. Open the NetBackup for SQL Server interface.
2. Choose **File** > **Set SQL Server connection properties**.
3. In the **Host** box, specify the public name of the SQL Server host (SQLHOST).
4. Click **OK**.
5. Choose **File** > **Restore Backup SQL Server Objects**.
6. In the **SQL Host** box, specify the primary (or NetBIOS) name.
7. In the **Source Client** box, specify the private interface name.



Backup History Options

Use this window to select how NetBackup will filter the SQL Server backup images that will be displayed in the restore dialog.

Browse for SQL Server objects that were backed up from :

SQL Host: SQLHOST

Source Client: SQLHOST-NB

Images will be selected from NetBackup Master Server: CADOO

Use the client properties window if you would like to select images from a different master server.

Time filter

start date: 2005 year, 4 mo, 19 day

end date: 2005 year, 5 mo, 19 day

8. Click **OK**.

   A dialog opens that shows the SQL Server backups of the SQL Host made on the private network interface.

9. Continue with the restore as normal.

# Backing up and Restoring Databases that contain Full-text Search Catalogs (SQL Server 2005)

If your SQL Server 2005 databases contain full-text search catalogs, then they will automatically be backed up when you back up these databases and restored as a part of the recovery process. Nevertheless, some basic information about full-text catalogs is provided in this section to help you to understand how the backup and restore processes work.

## Full-text Catalog Directory Structure

A full-text catalog consists of a root directory, subdirectories, and dozens of files. The root full-text directory is named by SQL Server based on the directory name given by the end-user. Typically, the directory name is the same as the index name, but SQL Server may alter it in case there are name conflicts. For example, if there is already a full-text catalog named ‘cats’ for database ONE and a user creates a second database named ‘cats’ for database TWO, then SQL Server will create a root full-text directory named ‘cats0000’ for the second full-text catalog.

SQL Server also creates a logical file to contain full-text catalogs. The logical file is named as `sysft_`*`full-text catalog name`*. For example, in the previous example, the ‘cats’ catalog would be contained by `sysft_cats`. This name has import for the database move operation.

SQL Server places the full-text catalog directory into the following path by default: *`SQL-instance-install-path`*`\FTData\`*`full-text-catalog-root-directory`*. For example, the cats directory for database TWO mentioned above would be placed into a path such as:

```
C:\Program Files\Microsoft SQL Server\MSSQL.1\FTData\cats0000\
```

However, when performing a database move command, you can establish the path at the location of your choosing.

One additional characteristic of the full-text catalog is its association with a filegroup. By default, the association is made with the primary filegroup; however, it is also possible to select a different filegroup for which to make the association.

## Backing Up and Restoring Databases Containing Full-text Catalogs

When you back up a database containing a full-text catalog, NetBackup writes the catalog name and directory path to NetBackup catalog. This allows NetBackup for SQL Server to display the full-text catalog information in the backup history properties windows when you select the database backup image. In addition, if you choose to move the database, then the saved full-text catalog information allows NetBackup for SQL Server to generate specifications for the move template allowing you to specify a path for the catalog directory.

NetBackup for SQL Server also extends the snapshot backup to include all of the files in any full-text catalog associated with the object being backed up. This impacts backups database, filegroup, and file backups as follows:

- Database backup: The full-text catalog files are backed up with the full database snapshot backup.
- Filegroup backup: The full-text catalog files are backed up for any catalog that is associated with the filegroup.
- File backup: NetBackup for SQL Server will back up the full-text catalog files along with a snapshot file backup based on these conditions:
    - The file belongs to a filegroup that is associated with a full-text catalog
    - The file has the lowest sort-order of any file belonging to the filegroup.

Backups and restores based on snapshots are impacted based on the extra catalog files that are included in the snapshot.

## Moving a database containing full-text catalogs

NetBackup for SQL Server extends the move template when a database contains a full-text catalog by including the following text:

```
#
#  Replace the full text catalog directory path
#  <C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\FTData\HisCatalog>
#  with a new directory path. Also remove the hash mark <#> which precedes the keyword <TO>.
#  The target of the MOVE keyword must be "sysft_HisCatalog".
MOVE  "sysft_HisCatalog"
#TO  "C:\Program Files\Microsoft SQL Server\MSSQL.1\MSSQL\FTData\HisCatalog"
#
```

As indicated by these instructions, the template should be modified by changing the target directory path.

The name `sysft_HisCatalog` is the logical name of the file containing the full-text directory named `HisCatalog` and is obtained by prepending the catalog name with "`sysft_`".

# Using Batch Files

NetBackup for SQL Server uses scripts called batch files for initiating backup and restore operations. A batch file uses the `.bch` extension and is typically executed from the *`install_path`*`\DbExt\MsSql\` directory. Refer to the Appendix "Sample Batch Files" on page 219 for example batch files.

## Overview of Batch Files

You will have to create a batch file if you start up operations with:

- the Manage Scripts dialog in NetBackup for SQL Server
- the `dbbackex` command line
- the NetBackup scheduler

You can easily create a fully functional batch file by selecting the **Save** option in the backup or restore dialog and then clicking **OK**. The following sections, however, provide guidelines to help you to understand the contents of a batch file and how to modify or create one on your own.

## Text Format Used in Batch Files

Batch files created by the SQL Server Agent interface are Unicode text. This permits inclusion of object names and image names that include non-English characters. If you create a batch file manually, use notepad or some other tool that generates plain text. If your batch file contains any non-English characters save it as Unicode.

## Guidelines for Creating and Using Batch Files

- Give the file a unique name with the extension `.bch`.
- Place the batch file in the *`install_path`*`\NetBackup\DbExt\MsSql\` folder.
- Batch files must reside on the host from which they executed. If performing actions on a remote host, the batch file must reside on that remote host.

## Notes

- A batch file consists of a series of operations (backups and restores) which by default are run in sequence.
- Each operation consists of a series of <*keyword value*> pairs, which completely define the total operation.

- The keyword is not case sensitive but the value is. Generally, you will be safe if you code both the keyword and value in uppercase, with the exception that, if you use the NBIMAGE keyword option, then the value must be specified exactly as it is stored by NetBackup Server.
- Operations are not nested.
- With the exception of the BATCHSIZE and GROUPSIZE parameters, <*keyword value*> pairs are not global. If you use BATCHSIZE or GROUPSIZE, then it must appear only once in your batch file and it must appear in the first operation.
- Within an operation, the <*keyword value*> pairs may appear in any order except that each operation must be terminated by `ENDOPER TRUE`.
- You can include comment lines in your batch file by placing a hash mark (‘#’) in the first column.

## Keywords and Values Used in Batch Files

This section describes the keywords and values used in a batch file.

Keywords and Values Used in Batch Files

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| ALTCLIENT (Same as BROWSECLIENT) | string | no | none | Restores images from a host other than the local host. |
| BACKUPMODEL | BACKUPMODEL_ CONVENTIONAL, BACKUPMODEL_ SNAPSHOT | no | BACKUPMODEL_ CONVENTIONAL | Valid only for restore. Indicates whether the backup was originated from a snapshot method |
| BATCHSIZE | integer | no | 1 | Number of operations to start up simultaneously. Applies to all of the operations in the batch file. Must appear before the end of the first operation. Range is 1-10. |
| BLOCKSIZE | integer | no | 0 | Applicable for backup operations only. Block size is calculated as 512 bytes * $2^{BLOCKSIZE.}$ Range is 1-7. |
| BROWSECLIENT (Same as ALTCLIENT) | string | no | none | Restores images from a host other than the local host. |
| BUFFERS | integer | no | 1 | Number of buffers per stripe. Range is 1-32. |
| CLOAKEDBACKUP | TRUE or FALSE | no | FALSE | Applies only to Advanced Client database backups. If TRUE, the backup will not be recorded by SQL Server and therefore cannot be the basis for any subsequent differential backup. See "Using Copy-Only or Cloaked Snapshot Backups to Impact How Differentials are Based (SQL 2000 and SQL 2005)" on page 143. |

Keywords and Values Used in Batch Files

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| CONSISTENCYCHECK | FULLINCLUDINGINDICES,<br>FULLEXCLUDINGINDICES,<br>PHYSICALCHECKONLY,<br>CHECKCATALOG | no | none | Performs the specified consistency check after the restore has been completed. |
| COPYONLY | TRUE or<br>FALSE | no | See description | If TRUE, an out-of-band backup will be created by SQL Server so that it does not interfere with the normal backup sequence. The default value is FALSE except for full database persistent frozen image backups. COPYONLY is only available for SQL Server 2005 database, differential, and transaction log backups. |
| DATABASE | string | yes | none | Name of database. For backup operations, specify value $ALL to designate all databases (except for tempdb.) |
| DBMS | MSSQL | no | MSSQL | You can specify MSSQL only. |
| DSN | string | no | saved from GUI user session | ODBC data source name. Deprecated. |
| DUMPOPTION | INCREMENTAL | no | none | Specifies INCREMENTAL restoring from an incremental backup. |
| ENDOPER | TRUE | yes | none | Terminates each operation specified in the batch file. |
| EXCLUDE | string | no | none | Name of a database to exclude when DATABASE $ALL is specified in a batch operation |

**Note** EXCLUDE can be used in a batch file only if DATABASE $ALL is used.

| | | | | |
|---|---|---|---|---|
| GROUPSIZE | integer between 1 and 32 | no | none | The number of databases that will be snapped as a single SQL Server backup image. See “Using NetBackup for SQL Server with Advanced Client” on page 133. |

Keywords and Values Used in Batch Files

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| INHIBITALTBUFFER METHOD | TRUE, FALSE | no | FALSE | Tells NetBackup whether to consider the candidacy of alternate buffer method. |
| MAXTRANSFERSIZE | integer | no | 0 | Maximum transfer size is calculated as 64 kilobytes bytes * $2^{\text{MAXTRANSFERSIZE.}}$ Range is 1-6. |
| MOVE | file group | no | none | Specifies a filegroup name. Used for restore types PARTIAL and MOVE. |
| NBIMAGE | string | yes* | none | Specifies a NetBackup image for the restore operations. See note below.<br>* Required for restore operations. |
| NBSCHED | string | no | none | If the NetBackup policy has several *Application Backup Policy* schedules, use NBSCHED to select amongst them. |
| NBSERVER | string | no | none | Specifies which master server to use for the backup or restore operation. Also see following note. |

**Note** If NBSERVER is not specified in a batch file operation, the master server defaults to the name specified at HKEY_CURRENT_USER\Software\VERITAS\NetBackup\NetBackup for Microsoft SQL Server\DEFAULT_SQL_NB_MASTER_SERVER.

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| OBJECTNAME | string | yes* | none | Specifies a file or filegroup name for file/filegroup backups and restores,<br>* If OBJECTTYPE= FILE or FILEGROUP. |
| OBJECTTYPE | DATABASE, TRXLOG, FILEGROUP, FILE | no | DATABASE | Specifies whether you are backing up or restoring a database, transaction log, filegroup, or file. |
| OPERATION | BACKUP, RESTORE | no | BACKUP | Type of operation, either backup or restore. |
| PAGE | Page ID | no | none | SQL Server 2005 only. Specifies a page ID for a page restore operation. |

Keywords and Values Used in Batch Files

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| PARTIAL | TRUE, FALSE | no | FALSE | SQL Server 2005 only. Specifies that a partial backup or restore will be performed. |
| PASSWORD | string | no | null | Password for logging into SQL Server. |
| RECOVERED STATE | RECOVERED, STANDBY, NOTRECOVERED, TRUE, FALSE | no | RECOVERED | See the following note. |

**Note** RECOVERED means that the database should be restored to the recovered state. NOTRECOVERED means that it should remain in the loading state following the restore. STANDBY means that the database should be restored to standby state. If STANDBY is used, then the STANDBYPATH keyword is also required. TRUE and FALSE, when used as values for RECOVEREDSTATE, are synonyms for RECOVERED and NOTRECOVERED.

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| RESTOREBEFOREMARK[1] | string | no | none | SQL Server 2000 or later restore option. Specify transaction log mark. |
| RESTOREBEFOREMARK AFTERTIME[1] | string | no | none | SQL Server 2000 or later restore option. Specify transaction log mark. |
| RESTOREOPTION | REPLACE | no | none | Tells NetBackup to use the SQL Server replace option on a restore. |
| RESTOREPAGES | TRUE, FALSE | no | FALSE | SQL Server 2005 only. Specifies that a page restore operation will be performed. |
| RESTORETOMARK[1] | string | no | none | SQL Server 2000 or later restore option. Specify transaction log mark. |
| RESTORETOMARK AFTERTIME[1] | string | no | none | SQL Server 2000 or later restore option. Specify transaction log mark. |
| RESTORETYPE | FULL, PARTIAL, MOVE | no | FULL | Full = Full database restore, Partial = Partial database restore, Move = Database move |

**Note** RESTORETYPE is applicable only to RESTORE database operations. PARTIAL requires SQL Server 2000 or later. If MOVE is used, then the batch file should contain a series of one or more <MOVE><filegroup> and <TO><file path> sequences. If PARTIAL is used, the sequence for PARTIAL must specify all of the filegroups in the database whose backup image is referenced by the NBIMAGE keyword.

Keywords and Values Used in Batch Files

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| ROLLBACKVOLUME | TRUE,FALSE | no | FALSE | Tells NetBackup to do the recovery of a persistent frozen image backup using the volume rollback method. |
| SQLHOST | string | no | See note | Name of SQL Server host. |
| SQLINSTANCE | string | no | See note | Name of SQL Server instance |

**Note** If SQLHOST is not specified in a batch file operation, then the SQL Server host is obtained from `HKEY_CURRENT_USER\Software\VERITAS\NetBackup\NetBackup for Microsoft SQL Server\DEFAULT_SQL_HOST`. If `SQLINSTANCE` is missing, then the default SQL Server instance is assumed for the SQL Host.

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| STANDBYPATH | string | no | none | Specify a fully- qualified file path to use for the standby redo log. |
| STOPAFTER[1] | datetime string | no | none | (SQL Server 2000 or later) Specifies datetime for **Restore to mark** options. The datetime string is formatted as YYYY/MMDDHH:MM:SS. |
| STOPAT[1] | datetime string | no | none | Specifies point-in-time recovery of a transaction log. The datetime string is formatted as YYYY/MMDDHH:MM:SS. |
| STORAGEIMAGE | string | no | none | Used for restoring a database that was backed up using a grouped Advanced Client snapshot. STORAGEIMAGE identifies the image with which the physical files are associated. |
| STRIPES | integer | no | 1 | Number of stripes. Range is 1-32. |
| TO | file path | no | none | Specifies a filegroup destination path. Required for each MOVE keyword. Also must sequentially follow each MOVE entry. The value may be delimited with single quotes. |

Keywords and Values Used in Batch Files

| Keyword | Values | Required? | Default | Description |
|---|---|---|---|---|
| TRACELEVEL | MIN, MID, MAX | no | MIN | Trace level. |
| TRXOPTION | NOTRUNC, TRUNCONLY, TAILLOG | no | none | SQL Server transaction log backup options.<br>If neither NOTRUNC nor TRUNCONLY is selected, then the transaction log will be backed up and truncated. If TAILLOG is selected, the tail log will be backed up and restored. |
| USERID | string | no | sa | Userid for logging into SQL Server. |
| VDITIMEOUTSECONDS | integer | no | 300 | Timeout interval for SQL Server Virtual Device Interface |
| VERIFYONLY | TRUE,FALSE | no | FALSE | (SQL 2005) Tells SQL Server to verify a backup image but not to restore it. |
| VERIFYOPTION | NONE, STOPONERROR CONTINUEAFTERERROR | no | NONE | (SQL 2005) This option is only valid for databases that have an active page. STOPONERROR will perform verification and stop if a verification error occurs. CONTINUEAFTERERROR will perform verification but continue if a verification error occurs. |

[1] STOPAT, RESTORETOMARK, RESTORETOMARKAFTERTIME, RESTOREBEFOREMARK, and RESTOREBEFOREMARKAFTERTIME are mutually exclusive restore parameters. If either RESTORETOMARKAFTERTIME or RESTOREBEFOREMARKAFTERTIME are used, then the batch file must also specify a datetime string with the keyword STOPAFTER.

## Creating a Batch File

You can use any of the backup or restore dialogs to create a batch file containing a NetBackup for SQL Server script that can be executed at a later time from the Manage Scripts dialog, from the `dbbackex` command line program, or by the NetBackup scheduler. To see examples, refer to "Sample Batch Files" on page 219.

### ▼ To create a batch file

1. Choose **File** > **Backup SQL Server Objects** or **File** > **Restore SQL Server Objects**.

2. Select the object you wish to back up or restore.

3. Select the desired options.

   See "Backup Microsoft SQL Server Objects" on page 200 or "Restore Microsoft SQL Server Objects" on page 207 for a description of the available options.

4. In the **Backup script** group, choose **Save**.

5. Click **Backup** or **Restore**.

6. Specify a filename and click **Save**.

**Note** Alternatively, in step 6 you can highlight the name of an existing file, and NetBackup for SQL Server will append the new script to it.

7. Click **Yes** to open and edit the batch file.

## Running Batch Files

Once you have created a batch file, you manually run it from the NetBackup for SQL Server interface.

### ▼ To run a batch file

1. Log in to the host and instance you wish to access. Refer to "Selecting the SQL Host and Instance" on page 57.

2. Choose **File** > **Manage script files**.

3. Double-click on the desired batch file.

4. Click **Start**.

5. To monitor the operation, choose **File** > **View status**.

# Performing Backups and Restores of Remote SQL Server Installations

You can use NetBackup for SQL Server to backup and restore databases on a remote host. Generated batch files must be saved on the remote host. You can launch the operation from the local installation of NetBackup for SQL Server, from an automatic backup policy, or from a manual backup.

### ▼ To perform backups and restores of remote SQL Server installations

1. Select the host and instance you wish to access. Refer to "Selecting the SQL Host and Instance" on page 57.

2. Create the backup or restore script.

   - To create a backup script, choose **File** > **Backup SQL Server Objects**.
   - To create a restore script, choose **File** > **Restore Backup SQL Server Objects**.

3. Select the desired options for the operation.

   See "Backup Microsoft SQL Server Objects" on page 200 for a description of the available backup options.

   See "Restore Microsoft SQL Server Objects" on page 207 for a description of the available restore options.

   **Save** is enabled in the backup and restore dialogs. **Launch immediately** is disabled because the generated script must be executed on the remote host that you are logged on to.

4. Click **Backup** or **Restore**.

5. In the Save Script As dialog, navigate to the `*install_path*\NetBackup\DbExt\MsSql\` folder on the remote host, and save the batch file there.

6. To launch the backup or restore operation:

   - Run the operation from the local installation of NetBackup for SQL Server, see "Running Batch Files" on page 108.

     OR

   - Create a new policy and add the batch file to the Backup Selections list in the policy. For more information, see "Configuring Backup Policies" on page 13.

# Restoring Multi-Streamed Backups

Restoring backups from multiple stripes is automatic from the Restore Microsoft SQL Server Objects dialog. Select the object you wish to restore and NetBackup for SQL Server will find all of the related backups and restore them. Upon restore, all of the streams must also be available at the same time.

## Conventional Backups Using Multiple Streams

If you specified multiple stripes for a conventional or stream-based backup, then the number of backup streams that you specified was created. NetBackup for SQL Server names these streams, for example:

```
juneberry.MSSQL7.COLE.db.pubs.~.7.001of003.20050908200234..C
juneberry.MSSQL7.COLE.db.pubs.~.7.002of003.20050908200234..C
juneberry.MSSQL7.COLE.db.pubs.~.7.003of003.20050908200234..C
```

If you wish to create your own batch file to restore a striped object, you would specify only the first stripe name with the NBIMAGE keyword. NetBackup for SQL Server finds the remaining ones automatically. See "Using bplist to Retrieve SQL Server Backups" on page 112 for more information about backup names used for SQL Server objects.

## Advanced Backup Methods Using Multiple Streams

If you specified multiple stripes for any Advanced Client backup, which streams the frozen image to tape, then NetBackup divides the number of component files equally among the number of stripes. If the number of files is less than the specified number of stripes, then the agent will perform the backup using only as many stripes as there are files.

**Note** Since Persistent Frozen Image (PFI) policies do not stream the component files to tape, PFI backups ignore the multiple-stream directive.

With SQL Server backups performed with Advanced Client, all of the backup streams are identified by the same name, such as:

```
juneberry.MSSQL7.COLE.db.Northwind.~.7.001of003.20051012131132..C
```

and are differentiated by NetBackup by their backup ids.

# Using bplist to Retrieve SQL Server Backups

If you plan to manually create a restore script, rather than through the NetBackup for SQL Server interface, you can use the `bplist` command to obtain restore images. See the *NetBackup Commands* guides for complete information about `bplist`.

To extract all of the NetBackup for SQL Server backups from a specific server for a specific client, execute the following command from the MS-DOS prompt.

```
install_path\NetBackup\bin\bplist -C <client> -t 15 -S <server> -R \
```

`where` *<client>* is the host machine on which NetBackup for SQL Server resides and *<server>* is the host machine of NetBackup server.

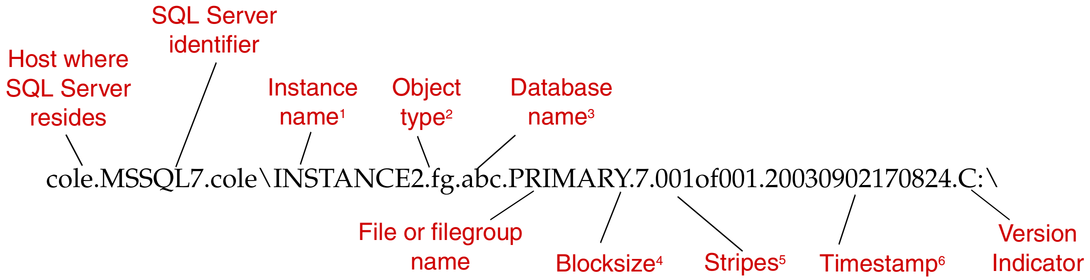
The following example shows how to obtain the list of SQL Server backups backed up from client juneberry to server cole:

```
C:\Program Files\NetBackup\bin\bplist -C juneberry -t 15 -S cole -R \
juneberry.MSSQL7.JUNEBERRY.db.pubs.~.7.001of003.20050920101716..C:\
juneberry.MSSQL7.JUNEBERRY.db.pubs.~.7.002of003.20050920101716..C:\
juneberry.MSSQL7.JUNEBERRY.db.pubs.~.7.003of003.20050920101716..C:\
juneberry.MSSQL7.JUNEBERRY.fil.pubs.pubsnew.7.001of001.20050919175149..C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.trx.abc.~.7.001of001.20050902170920..C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.fg.abc.PRIMARY.7.001of001.20050902170824.C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.db.Howard's Barbeque.~.7.001of001.20050901085255..C:\
juneberry.MSSQL7.JUNEBERRY\NEWINSTANCE.inc.Howard's Barbeque.~.7.001of001.20050903108552..C:\
juneberry.MSSQL7.COLE.db.pubs.~.7.001of001.20050907100101..C:\
juneberry.MSSQL7.COLE.db.pubs.~.7.001of001.20050908200234..C:\
```

**Note** The colon and backslash, which terminate each line, are not part of the backup name.

## Backup Names

The backup name is a string consisting of the following components separated by a delimiter specified by the character preceding the "C" at the end of the backup image name.



1. Named instances are formatted as *host\instance-name*. The default instance is simply the name of the host machine.

2. The object types are `db`, for database; `inc`, for database differential; `trx`, for transaction log; `fg`, for filegroup; `fdg`, for filegroup differential; `fil`, for file.

3. The name of the file or filegroup if the object type is a file or filegroup; otherwise the symbol ~ is used.

4. The blocksize is specified as 256 kilobytes bytes * 2 $^{\text{blocksize}}$.

5. Stripes are specified as <*stripe number*>of<*total stripes*>. Non-striped backups are always `001of001`. For striped backups, <*total stripes*> is the total number of stripes for the backup. <*stripe number*> is the count number of the backup for that backup, starting with 001.

6. The format of the timestamp is *YYYYMMDDHHMMSS*.

7. The character immediately preceding the version indicator is the delimiter, which by default is a period. However, if a period is used in any of the fields, the delimiter may be another character. "C" is for NetBackup 3.4.1 and higher.

# Disaster Recovery 6

This chapter contains information and instructions on restoring the SQL Server in a disaster recovery scenario.

# Preparing for Disaster Recovery of SQL Server

When you are developing your SQL Server disaster recovery plan you will need to plan for how to recover from corruption of the master database as well as loss of your host machine. If the master database has been corrupted, then SQL Server will not start. When this happens you will need to use the SQL Server rebuild master (`rebuildm.exe`) utility to start the SQL Server service. This utility, however, does not recreate the schema information of your application databases. To recover your database schema use the NetBackup SQL Server agent to restore your latest backup of the master database.

As a starting point, disaster recovery of SQL Server assumes that you have already put in place a strategy to recovery from other sorts of data loss such as disk, software, and human error. To prepare for disaster recovery you will need to make frequent backups of the master database, minimally, after you have added or dropped databases or carried out other operations that may result in schema definitions.

# Disaster Recovery of SQL Server

SQL Server corrects itself automatically from temporary or minor problems. However, most disasters are beyond the scope of the automatic recovery feature. For example, if a database becomes severely corrupted, or there is a catastrophic failure, recovery must be initiated by the system administrator.

User-initiated recovery could entail either restoring the entire server, including the SQL Server databases, from full system backups, or restoring only the SQL Server databases to a newly-installed or other available SQL Server.

Restoring the entire server has the added benefit of recovering other applications and data which may have resided on the server at the time of failure, and can be accomplished using one of the following methods:

- Manual recovery of the server. This method involves manually restoring the server from full system backups (see "Preparing for Disaster Recovery of SQL Server" on page 116).
- NetBackup Bare Metal Restore. BMR automates system recovery by restoring the operating system, system configuration, and all system and data files. Refer to *NetBackup Bare Metal Recovery System Administrator's Guide* for more information.

Alternatively, the SQL Server databases can be restored to a newly-installed or other available server. This server should be running the same version of Windows on the same hardware platform, the same Service Pack level, and the same version of SQL server with the same service pack as the original server.

For the purposes of disaster recovery, you should only be restoring to a new installation of SQL Server. If you want to restore to an existing installation of SQL Server with other active databases, refer to "Disaster Recovery of SQL Server Databases."

After recovery of the server is complete, or after the new server installation is available, recovery of the SQL Server databases can begin.

# Disaster Recovery of SQL Server Databases

If you are restoring to a new SQL Server installation, skip the steps for rebuilding the master database. If you are running SQL Server in a cluster or you are using SQL Server 7.0, you will need to start SQL Server in single-user mode before restoring the databases.

To recover the SQL Server databases, you need to perform the following tasks:

- rebuild the master database (if restoring to an existing SQL Server)
- start SQL Server in single-user mode (SQL Server 7.0)
- restore the SQL databases

### ▼ To rebuild the master database

1. Run the Rebuild Master utility (`\`*`SQL Server installation directory`*`\bin\Rebuildm.exe`).

**Note** See the Microsoft SQL Server official documentation for information on how to use `rebuildm.exe`.

2. When the rebuild is complete, restart the SQL Server services if necessary.

### ▼ To start SQL Server in single-user mode

**Note** SQL agent will automatically put the database in single-user mode for restores of non-clustered SQL 2000 or higher.You need to start SQL Server in single-user mode only if SQL Server is configured in a cluster or if you have a version of SQL Server 7.0.

1. In the Windows Control Panel, open **Services**.
2. Select the MSSQLServer service
3. Type the following in the **Startup Parameters** box:

   **`/m`**

4. Click **Start**.

### ▼ To restore the SQL databases

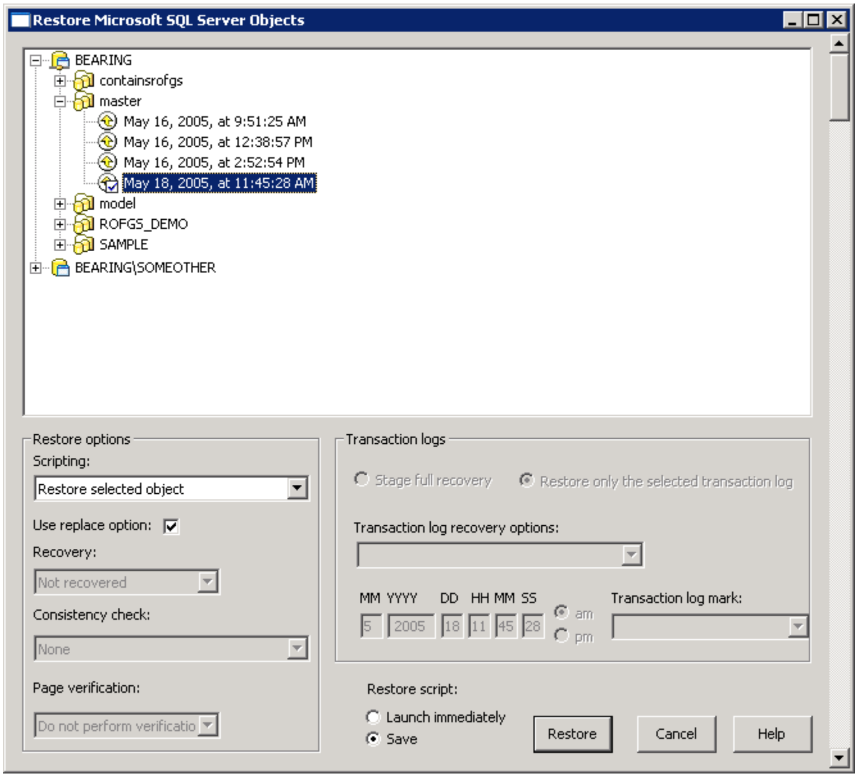
1. If you have SQL 7.0, you must start SQL Server in single-user mode before restoring the SQL databases. See the previous procedure for details.

2. Open the NetBackup for SQL Client interface.

3. Locate all the media required to perform the restore operations. Choose **File** > **Restore Backup SQL Server Objects**.

4. Select the backup image that contains the copy of the master database to be restored.

   Select only the master database at this time.



Restore Microsoft SQL Server Objects

- BEARING
  - containsrofgs
  - master
    - May 16, 2005, at 9:51:25 AM
    - May 16, 2005, at 12:38:57 PM
    - May 16, 2005, at 2:52:54 PM
    - ☑ May 18, 2005, at 11:45:28 AM
  - model
  - ROFGS_DEMO
  - SAMPLE
- BEARING\SOMEOTHER

Restore options

Scripting: Restore selected object

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ● Restore only the selected transaction log

Transaction log recovery options:

| MM | YYYY | DD | HH | MM | SS | | Transaction log mark: |
|---|---|---|---|---|---|---|---|
| 5 | 2005 | 18 | 11 | 45 | 28 | ● am<br>○ pm | |

Restore script:

○ Launch immediately
● Save

5. Click **Restore**.

6. If you manually placed SQL Server in single-user mode, restart the SQL Server service after the restore has completed.

7. Continue restoring the remaining SQL Server Databases.

   Follow the instructions for restoring SQL databases, differentials, transaction logs, files and filegroups from the "Using the NetBackup Database Extension Graphical User Interface" on page 56.

When all of the restore operations have completed successfully, then the recovery of the SQL Server databases is complete.

After the recovery has been completed, it is strongly recommended that a full database backup be performed as soon as possible.

# Using NetBackup for SQL Server with Clustering Solutions 7

NetBackup for SQL Server can be used for backing up and restoring installations which are clustered either with Microsoft Cluster Server (MSCS) or VERITAS Cluster Server (VCS).

# Support for Microsoft Cluster Server Clusters

NetBackup supports MSCS on 32-bit systems. On SQL Server 7.0, a maximum of 2 failover nodes is supported, on SQL Server 2000, 4 nodes, on SQL Server 2005, 8 nodes.

Unique SQL Server instances in an MSCS cluster are distinguished by the virtual server name. An operation performed with an instance of SQL Server which is clustered with MSCS is very similar to a non-clustered operation except that the client or host name is the SQL virtual name and the designated instance is <*default*>.

If SQL Server shares a cluster with a virtual NetBackup server, then SQL Server and the NetBackup server may be placed in the same group; but they must have different IP and network names.

# Support for VERITAS Cluster Server Clusters

NetBackup for SQL Server supports VCS clusters with active SQL Server instances. The VirtualName attribute under the VERITAS Cluster Server resource type, Lanman, is the name of the virtual SQL Server. An operation performed with an instance of SQL Server that is clustered with VCS is very similar to a non-clustered operation except that the client or host name is the VirtualName and the designated instance is <*default*>.

# Installing NetBackup for SQL Server in a Cluster

## Installation Prerequisites

To use NetBackup for SQL Server in a cluster, the following prerequisites must be met:

- SQL Server installed on each node to which NetBackup can failover in the cluster.
- NetBackup server software installed on the server.

  Refer to the *NetBackup Installation Guide for Windows* for details on installing the NetBackup server.
- NetBackup for Windows server or client installed on each node to which NetBackup can failover in the cluster. The NetBackup for SQL Server software is installed along with the server or the client software.

  See the *NetBackup Installation Guide for Windows* for installation instructions on Windows clients.

## Installation Instructions

NetBackup for SQL Server is installed with the server and client software. To use this agent in a clustered environment, you need to register a valid license key for it on each node on which NetBackup server is installed.

# Configuring the NetBackup Server to Be Aware of Clustered SQL Server Instances

Perform the following configuration steps after the installation of a virtual SQL Server (VIRTUALSERVER) has been created and, if applicable, the installation of a virtual NetBackup media server. The following actions must be performed on the master server or on a NetBackup remote client console acting for the master server.

▼ **To configure NetBackup Server to be aware of clustered SQL Server instances**

1. Using the NetBackup Administration Console, create a MS-SQL-Server policy (for example, VIRTSQLPOLICY), to specify the storage attributes of the backup. Set up the following attributes:

   a. Policy-storage unit: Specify a storage unit belonging to the storage group intended for backup. If you are using a virtual media server, then specify a storage unit belonging to the virtual media server.

   b. Create a backup policy schedule for VIRTSQLPOLICY.

   c. Add the virtual SQL Server name (VIRTUALSERVER) to the client list.

2. To incorporate SQL Server backup scheduling, create an automatic schedule in the VIRTSQLPOLICY. Create the automatic schedule as follows:

   a. Add an automatic backup schedule to VIRTSQLPOLICY.

   b. Add one or more script names (batch files) to the file list.

3. Create a standard backup policy (say, STDPOLICY). Add all physical names in the cluster to the client list of STDPOLICY.

4. Establish the permissions settings as follows for a redirected restore to a different client: On the master server, either:

   - Create a file called *install_path*`\NetBackup\db\altnames\No.Restrictions`

     or

   - Create each of the files, *install_path*`\NetBackup\db\altnames\NODEA`, *install_path*`\NetBackup\db\altnames\NODEB`.

**Note** Creating the `No.Restrictions` file allows all clients to perform redirected restores to different clients. This parameter may need to be added and removed according to site policies.

## Performing a Backup on a Virtual SQL Server Instance

Although NetBackup for SQL Server supports browsing for databases and images on any of the physical nodes, a user backup or restore will not be successful unless it is launched from the NetBackup client that is active. After NetBackup has been configured as described in the previous sections, backing up a SQL Server database from a virtual instance is similar to backing up a database from a non-virtual one.

▼ **To perform a backup on a virtual SQL Server instance**

1. Open the NetBackup for SQL Server interface on the active platform node.

2. Choose **File** > **Backup SQL Server Objects**.

3. Select one or more databases.

4. In the **NetBackup Policy** field, enter the name of the MS-SQL Server policy (VIRTSQLPOLICY) that was created for specifying storage attributes for the virtual SQL Server backup.

5. Click **OK**.

# Performing a Restore on a Virtual SQL Server Instance

### ▼ To perform a restore on a virtual SQL Server instance

1. Open the NetBackup for SQL Server interface on the active platform node.

2. Choose **File** > **Restore Backup SQL Server Objects**.

   The Backup History Options dialog is displayed.

3. From the **SQL Host** list, select the Virtual Server name (VIRTUALSERVER) as the SQL host.

4. From the **SQL Server Instance** drop-down list, choose **<default>**.

5. Click **OK**.

   The Restore Microsoft SQL Server Objects dialog is displayed.

6. Select a backup image or staged image list.

7. Click **OK**.

# Backing Up SQL Server in a Cluster with a Multi-Interface Network Connection

This section covers information that is required for combining SQL Server clustering with the usage of public-private interfaces to perform backups. Many administrators desire to reserve a separate network interface for their SQL Server host machines that is used for routing backup traffic. This type of configuration requires special consideration in terms of configuring both the NetBackup master server and the NetBackup client which backs up SQL Server. In addition, a special consideration is required in terms of how to browse for SQL Server backup images.

The following distinct network resources can be identified in a dual-interface SQL Server cluster.

- the public name of each SQL Server host (for example, SQLHOST1 and SQLHOST2),
- the private interface name used for backing up each of the SQL Server hosts (for example, SQLHOST1-NB and SQLHOST2-NB),
- the public virtual name of the SQL Server (for example, VIRTSQL), and
- the private virtual name of the SQL Server (for example, VIRTSQL-NB).

## Master Server Configuration

Two configuration changes must be made on the master server to allow for backups and restores over a private interface: the backup policies must include the private interface name in the Clients list of the policy and permissions must be added to allow for browsing of backups across the private interface.

### Adding Clients to the Policy

The private name of the client must be added to the Clients list of the policy. The NetBIOS or public name of the client should not be used.

▼ **To add clients to the policy**

1. Open the NetBackup Administration Console.

2. Create a new policy or open an existing policy.

   In the Clients list for the policy, add a new client. Specify the Client name as the private SQL virtual instance name, for example, VIRTSQL-NB.

### Adding Permissions That Allow for Browsing of Backups Across the Private Interface

The administrator can allow all clients or allow single clients to browse and restore backups performed over the multi-nic connection.

▼ **To allow all clients to browse for backups and perform restores**

❖ Add the empty file `NB_INSTALL\db\altnames\No.Restriction` on the master server.

  This option allows any NetBackup client to browse backups made by the SQL Server host machine.

▼ **To allow a single client to browse for backups and perform restores**

❖ Add the empty file, `NB_INSTALL\db\altnames\`*`SQLHOST`* on the master server.

  This only allows the SQL Server host machine, SQLHOST, to access backups managed by the master server.

## Client Configuration

The private interface name of the SQL Server host machine must be specified in the Backup, Archive, and Restore interface on the client.

The Client name can be set either during the installation of the NetBackup client, or it can be changed in the NetBackup Client Properties dialog in the Backup, Archive, and Restore interface.

▼ **To specify the private name of the client**

1. Open the Backup, Archive, and Restore interface.

2. Choose **File** > **NetBackup Client Properties**.

3. Click on the **General** tab.

4. In the **Client name** box, specify the private name of the client.

   For example, the client name used for the SQLHOST1 machine would be SQLHOST1-NB and the client name used for the SQLHOST2 machine would be SQLHOST2-NB.

## Performing Backups in a Cluster with a Multi-Nic Connection

To perform backups of SQL Server using a multi-nic connection, you need to specify the public name of the *virtual* SQL Server host in the SQL Server connection properties dialog. You also need to create and edit a batch file for backups so that it includes the private virtual SQL Server name.

### ▼ To perform backups

1. On either SQLHOST1 or SQLHOST2, open the NetBackup for SQL Server interface.

2. Choose **File** > **Set SQL Server connection properties**.

3. In the **Host** box, specify the public name of the virtual SQL Server host (VIRTSQL).

4. Click **Apply** and **Close**.

5. Choose **File** > **Backup SQL Server Objects**.

6. Select the databases to back up.

7. Select the desired backup options.

**Note** Do not attempt to perform an immediate backup from the backup dialog. The generated batch files must be modified before they can be run successfully.

8. From the **Backup script** options, choose **Save**.

9. Click **Backup**.

   A batch file similar to the following will be created:

```
OPERATION BACKUP
DATABASE "ACCOUNTING"
SQLHOST "VIRTSQL"
NBSERVER "THOR"
BROWSECLIENT "VIRTSQL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
ENDOPER TRUE
```

10. Change the line value associated with BROWSECLIENT from the public virtual SQL Server name to the private name.

```
OPERATION BACKUP
DATABASE "ACCOUNTING"
SQLHOST "VIRTSQL"
NBSERVER "THOR"
BROWSECLIENT "VIRTSQL-NB"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
ENDOPER TRUE
```

11. Place the modified batch file on both nodes in the cluster so that it will be available for scheduled backups, regardless of which node is active when a backup is initiated.

## Performing Restores in a Cluster with a Multi-Nic Connection

To perform restores of SQL Server in a multi-NIC, cluster environment, you need to specify the public name of the virtual SQL Server host in the SQL Server connection properties dialog and both the virtual SQL Server host name and the private interface name of the virtual SQL Server host in the Browse History Options dialog. You also need to create a batch file for restores and manually edit it to include the private name of the virtual SQL Server.

### ▼ To perform restores

1. On either SQLHOST1 or SQLHOST2, open the NetBackup for SQL Server interface.

2. Choose **File** > **Set SQL Server connection properties**.

3. In the **Host** box, specify the public name of the virtual SQL Server host (VIRTSQL).

4. Click **Apply** and **Close**.

5. Choose **File** > **Restore Backup SQL Server Objects**.

6. In the Backup History Options dialog, do the following.

   a. In the **SQL Host** box, specify the public name of the virtual SQL Server (VIRTSQL).

   b. In the **Source Client** box, specify the private name of the virtual SQL Server (VIRTSQL-NB).

7. Click **OK**.

8. Select the databases to restore.

**Note** Do not attempt to perform an immediate restore from the restore dialog. The generated batch files must be modified before they can be run successfully.

9. Select the desired restore options.

10. From the **Restore script** options, choose **Save**.

11. Click **Backup**.

    A batch file similar to the following is generated.

    ```
    OPERATION RESTORE
    OBJECTTYPE DATABASE
    DATABASE "ACCOUNTING"
    NBIMAGE
    "SQLHOST1.MSSQL7.VIRTSQL.db.ACCOUNTING.~.7.001of001.20040306111309..C"
    SQLHOST "VIRTSQL"
    NBSERVER "THOR"
    BROWSECLIENT "VIRTSQL"
    MAXTRANSFERSIZE 0
    BLOCKSIZE 7
    RESTOREOPTION REPLACE
    RECOVEREDSTATE RECOVERED
    ENDOPER TRUE
    ```

12. Change the line value associated with BROWSECLIENT from the public virtual SQL Server name to the private name.

    ```
    OPERATION RESTORE
    OBJECTTYPE DATABASE
    DATABASE "ACCOUNTING"
    NBIMAGE
    "SQLHOST1.MSSQL7.VIRTSQL.db.ACCOUNTING.~.7.001of001.20040306111309..C"
    SQLHOST "VIRTSQL"
    NBSERVER "THOR"
    BROWSECLIENT "VIRTSQL-NB"
    MAXTRANSFERSIZE 0
    BLOCKSIZE 7
    RESTOREOPTION REPLACE
    RECOVEREDSTATE RECOVERED
    ENDOPER TRUE
    ```

13. Run the modified batch file by choosing **File** > **Manage script files**.

# Using NetBackup for SQL Server with Advanced Client 8

This chapter explains how to use the NetBackup Advanced Client feature in conjunction with the NetBackup SQL Agent to employ snapshot technology for backing up and restoring SQL Server objects. To use NetBackup Advanced Client with NetBackup for SQL Server, you must register valid license keys for both Advanced Client and Microsoft SQL Server.

# NetBackup Advanced Client for SQL Server Overview

Advanced Client, when used in conjunction with NetBackup for SQL Server, enables you to back up and restore SQL Server objects by taking snapshots of the component files and then backing up the snapshot version to tape or committing them to another form of persistent storage. The snapshot technology uses SQL Server VDI (virtual device interface) quiescence to affect a momentary freeze on database activity while NetBackup takes a snapshot of the identified files.

The term database freeze is interchangeable with database quiescence and refers to the momentary state in which all activity in the database has been halted while a snapshot (or frozen image) copy is created. The state-change in which the freeze is lifted is referred to as database thaw.

The role of NetBackup in managing backup images taken from snapshots is common for both Advanced Client and standard database backups. In so far as NetBackup provides an identical view for cataloguing and browsing images and for accessing them in restore operations, it is transparent whether the images may have been created by standard database backups or by the Advanced Client. This allows for a similar operational experience whether the user is backing up and restoring standard SQL Server database images or images taken from snapshots. However, there are fundamental differences in the technologies.

Advanced Client backups, in contrast to standard backups, are file-based. This means that NetBackup determines the file list constituting the SQL Server object and backs it up asynchronously with respect to SQL Server. On the other hand, standard backups are stream-based, which means that SQL Server provides data to NetBackup buffer-by-buffer constituting a backup stream. The key role of SQL Server in file-based backups is to provide the mechanism for freezing database activity long enough for NetBackup to invoke a so-called snapshot provider that creates volume snapshots of the files.

The snapshot method is the software/hardware component that creates the snapshot image of the files. The method may be a VERITAS product such as Volume Manager, an operating system component such as VSS (Microsoft Virtual Shadow Service), or a third-party product. Snapshot methods use a variety of technologies such as copy-on-write, split-mirror, or RAID. They also provide several competing methodologies that may have different consequences in respect to factors such as recovery speed, impact of the backup on the client processor, and cost of specialized disks or processing units.

## Advanced Client Features

The following NetBackup Advanced Client features are available for use with NetBackup for SQL Server:

| | |
|---|---|
| Snapshot Backup | A snapshot is a disk image of the client's data made almost instantaneously. NetBackup backs up the data from the snapshot image, not directly from the client's primary data. |
| Instant Recovery (formerly Persistent Frozen Image or PFI) | An instant recovery backup is similar to a snapshot backup except that the snapshot is not transferred to tape. The restore may be made directly from snapshot copy. |
| Persistent Frozen Image with Backup to Tape | This feature is similar to the *Instant Recovery* feature except that a backup copy is made to a tape storage unit. If the image is needed for a restore, then a disk copy will be used unless it has been swapped out due to storage contention. In this case, the restore will fall back to the tape copy of the backup. |
| Offhost Alternate Client Backup | An offhost alternate client backup shifts the burden of backup processing to an alternate client, reducing the impact on the client's computing resources ordinarily caused by a local backup. The backup agent reads the data from the shared disk and writes it to storage. |

Although all of these features are provided through Advanced Client support for SQL Server, not all methods are supported.For information on selecting a method, see the *NetBackup Advanced Client System Administrator's Guide*. For a description of snapshot methods available for use with NetBackup for SQL Server, see the supported platforms matrix on the VERITAS support site.

## Supported Platforms

### ▼ To access the supported platforms information

1. Go to the VERITAS support web page: `www.support.veritas.com`

2. From the **Select Product Family** list, click **NetBackup Products**.

3. From the **Select Product** list in the right pane, click **NetBackup Enterprise Server**.

4. Under **Support Resources**, click **Compatibility and Reference**.

5. In the list of documents, click **NetBackup Enterprise Server (tm)/ Server** *x.x* **Advanced Client OS and Database Agent Compatibility (updated** *date***)**.

   For *x.x*, look for the current release. For *date*, look for the most recent date.

6. Click on the link for the PDF document, which is a downloadable file that enables you to view the supported database spreadsheet for this release.

7. Read the document and verify that the software in your environment is compatible with the NetBackup and NetBackup for SQL Server software.

# How SQL Server Operations Use Advanced Client

This section includes information on how NetBackup for SQL Server works with the Advanced Client option, including:

- how a backup method is chosen
- what is backed up in a snapshot backup
- what affects the performance of snapshot backups of SQL Server
- how to perform SQL Server snapshot backups
- how to perform SQL Server snapshot restores
- how multiple databases can be backed up in a single snapshot
- how to use a cloaked database snapshot to impact the basing of differential backups

## Selection of Backup Method

The selection of a backup methodology, whether standard or Advanced Client, is dependent on what policy is used. If a policy configured for Advanced Client is selected, then additional attributes relating to the policy determine the Advanced Client features and specific snapshot methods used. You can specify a policy name in the Backup Microsoft SQL Server Objects dialog (choose **File** > **Backup SQL Server objects**). Or specify <any>, if you want NetBackup to choose the policy.

Due to SQL Server limitations certain objects cannot be backed up via snapshots. These are database differentials, filegroup differentials, and transaction logs. If an Advanced Client policy is selected to back up one of these object types, then NetBackup will perform a stream-based backup using the storage unit provided in the policy configuration. If a storage unit is not provided, then NetBackup will use the default storage unit for the server.

## What is Backed Up

Although the database administrator works exclusively with logical objects, such as databases and filegroups, it is useful to gain some appreciation of the differences between file- and stream-based backups in terms of the data content that is archived. For stream-based backups, NetBackup simply captures the data stream content provided by SQL Server. If the user has specified multiple streams, then SQL Server opens multiple streams that NetBackup catalogs as separate images.

For file-based backups, NetBackup creates a file list consisting of all the physical files that constitute the object and supplies it to the Advanced Client, which is responsible for snapshot creation. If multiple streams have been specified, then NetBackup divides the file list into sub-lists. Each sub-list is backed up separately and constitutes a separate image. Users may notice that if multiple streams are specified for a file-based backup and if the number of streams exceeds the number of component files, then the number of file-based streams will not exceed the number of files. With stream-based SQL Server backups, SQL Server always creates exactly the number of streams that the end user specifies.

The file list used for backing up a SQL Server database consists of the physical files constituting the primary filegroup, any secondary filegroups, and the transaction log. Typically, these can be identified respectively by their name extensions, which are `.mdf`, `.ndf`, and `.ldf`. The file list for a filegroup backup consists of the physical files belonging to the filegroup. And, finally, the file list for a SQL Server file object backup consists of a single physical file, namely, the file that maps to the SQL Server file object.

## Performance Considerations

When a physical file is backed up using the Advanced Client, the backup will consist of the entire extent. This contrasts with stream-based SQL Server backups where only the actual data content of the objects are archived. For this reason, if you intend to use snapshot technology for backing up SQL Server, then it may be advantageous to use the SQL Server dynamic file allocation to reduce the likelihood that any of the component files contain large areas of empty space.

Another consideration for choosing between file- and stream-based backups concerns how SQL Server zeroes the target disk area prior to a stream-based restores. In some cases, this almost equals the total disk-copy time for restore. For Advanced Client restores, however, disk zeroing is not done so the total recovery time can be substantially less.

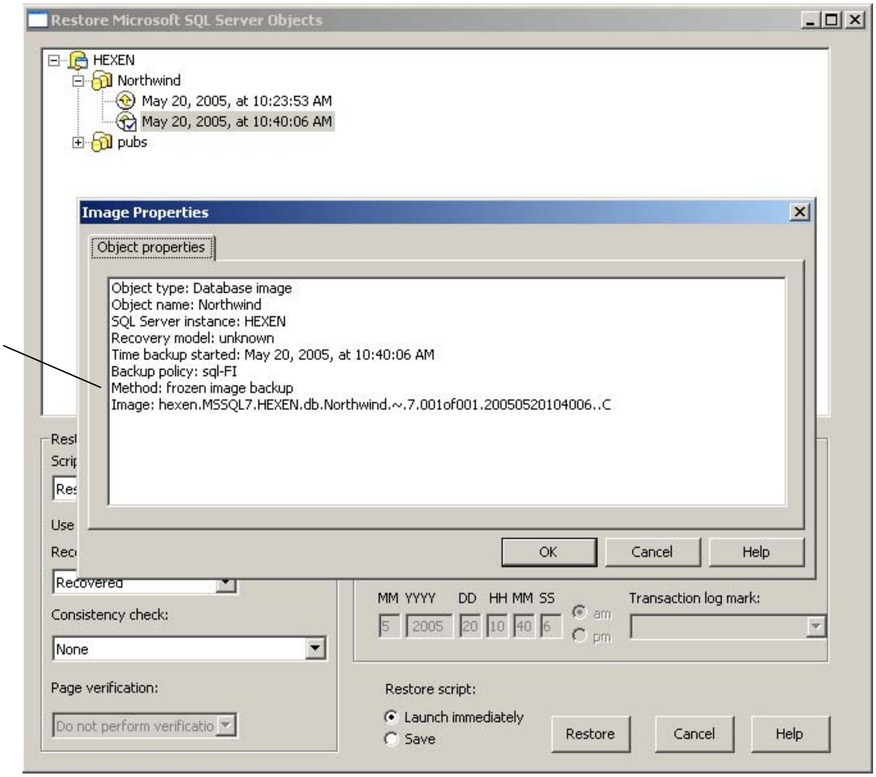
For further considerations on SQL Server disk initialization see “Instant Data File Initialization” on page 37 .

## Performing SQL Server Snapshot Backups

There are no special interfacing considerations for performing Advanced Client backups of SQL Server. A snapshot backup will be performed if the backup object is a database, a filegroup, or a file and a policy is selected which is configured for Advanced Client. If a differential or transaction log backup is attempted with an Advanced Client backup, then the operation will use the selected policy, but a standard database backup will be performed using the configured storage unit.

## Performing SQL Server Snapshot Restores

Backup images that were created from snapshots are displayed on equal footing with standard backup images. That is, all backup items—without regard to method—are displayed in a time-sequenced ordering that respects the composition of the database hierarchy. In addition, no weighting is given in determining an optimal recovery based on the backup method. To determine what backup method and policy were used when a SQL Server backup was created, right-click on the backup image and choose **Properties**.



Object type: Database image
Object name: Northwind
SQL Server instance: HEXEN
Recovery model: unknown
Time backup started: May 20, 2005, at 10:40:06 AM
Backup policy: sql-FI
Method: frozen image backup
Image: hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C

Recovered

Consistency check: None

Page verification: Do not perform verificatio

5 2005 20 10 40 6 ● am ○ pm

Restore script:
● Launch immediately
○ Save

Indicates backup was created with a frozen image (snapshot) method.

## Grouped Backups

The SQL Server agent provides a method in which multiple databases can be quiesced together and split-off to form a single snapshot. This minimizes the usage of system resources if the databases exist on a single volume, because the aggregation of constituent files would use one snapshot volume instead of one per database. The method for aggregating database Advanced Client backups is called backup *grouping*.

When databases are backed up in a group, all of the databases are quiesced simultaneously, and the constituent files of all of the databases are backed up to a single storage image under the same backup id. This means that an import and copy procedure would use only one image in order to export all of the database backups in the group.

### Requirements

The following requirements must be met for a grouped backup to be performed. If any of these requirements are not met, a standard backup will be performed.

- All backup operations must be full backups. Differential backups are not supported.
- The master database can not be included in a grouped backup.
- The same policy must be specified for each backup operation in the group.
- The same NetBackup server must be specified for each backup operation in the group.

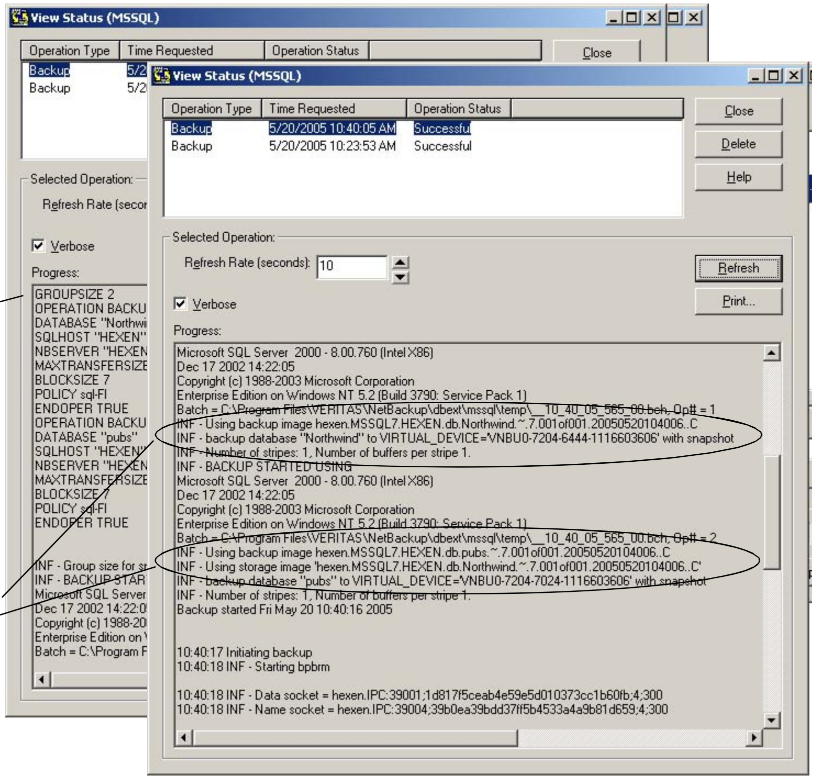
The simplest way to use grouped backup is to select multiple databases using the Backup Microsoft SQL Server Objects dialog box. If the above conditions apply, then the selected databases will be backed up as a group.

### Viewing the Progress of a Grouped Backup"

You can determine that a grouped backup is underway from the progress report.

In the figure "Progress report for a grouped backup operation," notice the keyword GROUPSIZE appears at the beginning of the batch file, indicating NetBackup will attempt to use grouping to back up the selected SQL Server databases. If the appropriate conditions apply, for example, all operations are full database backups, then all of the databases will be snapped and backed up as a group. When this happens, the progress log will display the backup image name as well as the storage image for each database in the group.

Progress report for a grouped backup operation



Indicates that this set of databases is a candidate to be backed up as a group..

Backup image name and storage image name shown in a grouped backup

| Operation Type | Time Requested | Operation Status |
|---|---|---|
| Backup | 5/20/2005 10:40:05 AM | Successful |
| Backup | 5/20/2005 10:23:53 AM | Successful |

Refresh Rate (seconds): 10

```
GROUPSIZE 2
OPERATION BACKU
DATABASE "Northwi
SQLHOST "HEXEN"
NBSERVER "HEXEN
MAXTRANSFERSIZE
BLOCKSIZE 7
POLICY sql-FI
ENDOPER TRUE
OPERATION BACKU
DATABASE "pubs"
SQLHOST "HEXEN"
NBSERVER "HEXEN
MAXTRANSFERSIZE
BLOCKSIZE 7
POLICY sql-FI
ENDOPER TRUE


INF - Group size for s
INF - BACKUP STAR
Microsoft SQL Server
Dec 17 2002 14:22:0
Copyright (c) 1988-20
Enterprise Edition on
Batch = C:\Program F
```

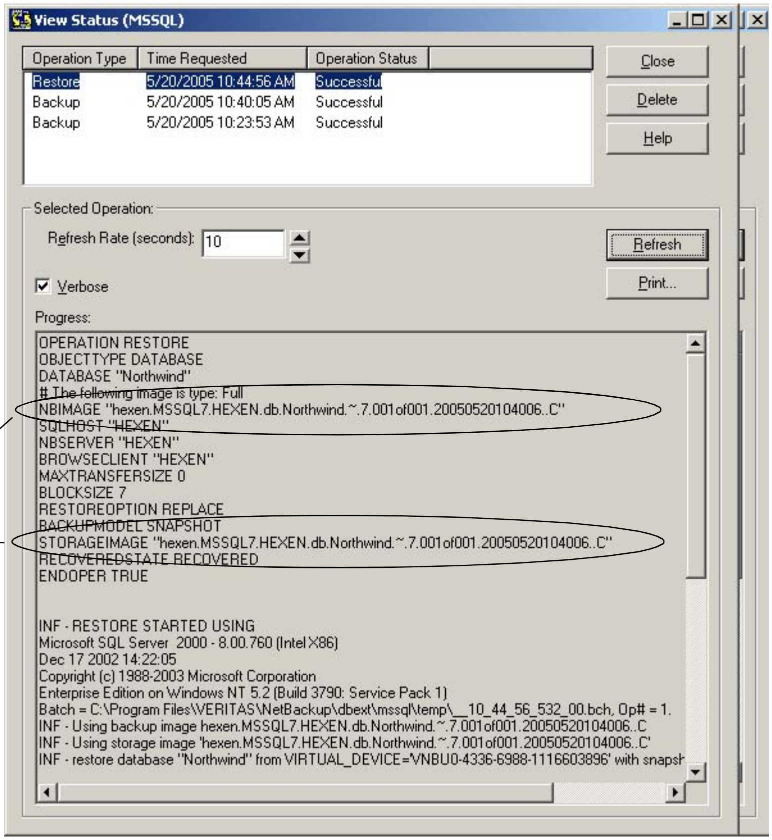
```
Microsoft SQL Server  2000 - 8.00.760 (Intel X86)
Dec 17 2002 14:22:05
Copyright (c) 1988-2003 Microsoft Corporation
Enterprise Edition on Windows NT 5.2 (Build 3790: Service Pack 1)
Batch = C:\Program Files\VERITAS\NetBackup\dbext\mssql\temp\__10_40_05_565_00.bch, Op# = 1
INF - Using backup image hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C
INF - backup database "Northwind" to VIRTUAL_DEVICE='VNBU0-7204-6444-1116603606' with snapshot
INF - Number of stripes: 1, Number of buffers per stripe 1.
INF - BACKUP STARTED USING
Microsoft SQL Server  2000 - 8.00.760 (Intel X86)
Dec 17 2002 14:22:05
Copyright (c) 1988-2003 Microsoft Corporation
Enterprise Edition on Windows NT 5.2 (Build 3790: Service Pack 1)
Batch = C:\Program Files\VERITAS\NetBackup\dbext\mssql\temp\__10_40_05_565_00.bch, Op# = 2
INF - Using backup image hexen.MSSQL7.HEXEN.db.pubs.~.7.001of001.20050520104006..C
INF - Using storage image 'hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C'
INF - backup database "pubs" to VIRTUAL_DEVICE='VNBU0-7204-7024-1116603606' with snapshot
INF - Number of stripes: 1, Number of buffers per stripe 1.
Backup started Fri May 20 10:40:16 2005


10:40:17 Initiating backup
10:40:18 INF - Starting bpbrm

10:40:18 INF - Data socket = hexen.IPC:39001;1d817f5ceab4e59e5d010373cc1b60fb;4;300
10:40:18 INF - Name socket = hexen.IPC:39004;39b0ea39bdd37ff5b4533a4a9b81d659;4;300
```

## Restoring a Database Backed Up in a Group

A database backed up in a group can be restored like any other database. Follow the instructions for “Restoring a Database Backup” on page 70.When you launch the restore operation, notice that the batch file (as displayed below in the progress log monitor) specifies both the name of the storage image and the name of the backup image.



| Operation Type | Time Requested | Operation Status |
|---|---|---|
| Restore | 5/20/2005 10:44:56 AM | Successful |
| Backup | 5/20/2005 10:40:05 AM | Successful |
| Backup | 5/20/2005 10:23:53 AM | Successful |

Selected Operation:

Refresh Rate (seconds): 10

☑ Verbose

Progress:

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE "Northwind"
# The following image is type: Full
NBIMAGE "hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C"
SQLHOST "HEXEN"
NBSERVER "HEXEN"
BROWSECLIENT "HEXEN"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
BACKUPMODEL SNAPSHOT
STORAGEIMAGE "hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C"
RECOVEREDSTATE RECOVERED
ENDOPER TRUE


INF - RESTORE STARTED USING
Microsoft SQL Server  2000 - 8.00.760 (Intel X86)
Dec 17 2002 14:22:05
Copyright (c) 1988-2003 Microsoft Corporation
Enterprise Edition on Windows NT 5.2 (Build 3790: Service Pack 1)
Batch = C:\Program Files\VERITAS\NetBackup\dbext\mssql\temp\__10_44_56_532_00.bch, Op# = 1.
INF - Using backup image hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C
INF - Using storage image 'hexen.MSSQL7.HEXEN.db.Northwind.~.7.001of001.20050520104006..C'
INF - restore database "Northwind" from VIRTUAL_DEVICE='VNBU0-4336-6988-1116603896' with snapsh
```

Storage image name and backup image name are shown when restoring from a grouped backup.

## Using Copy-Only or Cloaked Snapshot Backups to Impact How Differentials are Based (SQL 2000 and SQL 2005)

SQL Server records the history of successful database backups in the MSDB system database. It uses this history in deciding how to base differential backups. In particular, SQL Server creates differential database backups as cumulative with respect to the last full database backup that it has recorded in the MSDB. This allows for a quick recovery in case a failure has been detected after the last full database backup.

For example, assume that full backups are created every day at midnight; differentials are created every day at 6AM, noon, and 6PM; and transaction log backups are created every two hours. If a failure occurs at 7:50PM on Tuesday, then a point in time recovery could be achieved by restoring the full database from Tuesday at 12AM, followed by the differential at 6PM on Tuesday, and finally the transaction log at 8PM (specifying "TO 7:50 PM").

However, in some situations with persistent frozen image backups, it may not be feasible to retain the daily full backup after the next full backup has been created. So, if a point in time restore is required prior to the latest backup, the differentials would effectively be based on backups that no longer exist. Thus the only alternative would be to recover based upon the last full backup that had been retained using a potentially long sequence of transaction log images.

To resolve this issue, NetBackup allows you to create SQL Server Advanced Client backups that are not recorded in the MSDB. To create these backups in SQL Server 2005, NetBackup uses the *copy-only* backup feature, which allows the backups to be created as out-of-band. In SQL Server 2000, since copy-only backups are not available, NetBackup allows these backups to be *cloaked*, effectively creating the backup without telling SQL Server.

In SQL Server 2000, NetBackup uses cloaking to provide SQL Server with an "unsuccessful" status thereby inhibiting SQL Server from recording that the backup had occurred. Due to the unsuccessful status, the job line of the MSSQL progress monitor will show a `-1` status, whereas the NetBackup server job monitor registers the job as successful. NetBackup places information in its catalog to denote when a backup has been cloaked and uses this information in determining full database recovery paths. In SQL Server 2005, cloaking is not needed because SQL Server does not record the backup, but simply declares it as copy-only.

## Sample Backup Schedule Using Cloaked Backups

To understand how recovery staging works with copy-only or cloaked backups, consider a sample backup schedule with the following characteristics:

- The transaction log is backed up frequently, e.g., every two hours
- A full backup is saved to secondary storage, such as tape, once every several days
- Differential database backups are created several times per day
- A persistent frozen image backup is created several times per day and expires when the next one is created. This backup is created as copy-only (SQL 2005) or it is cloaked (SQL 2000).

The following is an excerpt from this schedule:

Sample backup schedule using cloaked backups

| Time | Full backup saved to secondary storage | Differential backup | PFI Copy-Only or Cloaked Backup | Transaction log backup |
|---|---|---|---|---|
| **Day 1** | | | | |
| 12 AM | ✔ | | | ✔ |
| 2 AM | | | | ✔ |
| 4 AM | | ✔ | | ✔ |
| 6 AM | | | ✔ | ✔ |
| 8 AM | | | | ✔ |
| 10 AM | | ✔ | | ✔ |
| 12 PM | | | ✔ | ✔ |
| 2 PM | | | | ✔ |
| 4 PM | | ✔ | | ✔ |
| 6 PM | | | ✔ | ✔ |
| 8 PM | | | | ✔ |
| 10 PM | | ✔ | | ✔ |

Sample backup schedule using cloaked backups

| Time | Full backup saved to secondary storage | Differential backup | PFI Copy-Only or Cloaked Backup | Transaction log backup |
|---|---|---|---|---|
| **Day 2** | | | | |
| 12 AM | | | ✔ | ✔ |
| 2 AM | | | | ✔ |

Under this schedule, full backups are performed every six hours. If a failure occurs, and is detected immediately, then the recovery can be achieved by restoring the last full backup and replaying, on average, three hours of transaction logs. However, if a failure is not detected until after the next full backup, then there would not be any full backups available since 12 AM on day 1. Since the persistent frozen image backups are cloaked, however, the differential backups would each be cumulative with respect to the last non-cloaked full backup.

In this example, suppose that an error occurs at 11:30 PM on day 1 but is not detected until 12:30 AM on day 2, after the 12:AM full backup. Since the 6 PM full backup no longer exists it would be necessary to begin the recovery with the backup taken at 12 AM on day 1. However, since all of the full backups since then were cloaked, the differential backup from 10 PM would be cumulative with respect to that backup. This means that the recovery sequence would be to restore the 12 AM day 1 backup followed by the 10 PM differential backup, followed by 1½ hours of transaction log backups.

If you use copy-only or cloaked backups, then the copy-only or cloaking attribute is displayed in the properties for the snapshot backup image. Differential backups are automatically associated with the correct full backup and are recognized by the SQL agent when it selects the recovery set for the full database restore.

**Caution** Microsoft SQL Server does not recognize the cloaked backup. Therefore, if it is incorporated in your database protection strategy, it is essential that you maintain a comprehensive set of transaction logs that span the time duration back to the last non-cloaked full backup.

## Creating a Cloaked Backup (SQL 2000)

### ▼ To create a batch file for a cloaked backup

1. Open an existing batch file in a text editor.
2. Insert the following:

   ```
   CLOAKEDBACKUP TRUE
   ```

3. Save the batch file.

## Creating a Copy-Only Backup (SQL 2005)

Any backup can be created as copy-only in SQL Server 2005. A persistent frozen image snapshot is automatically created as copy-only. See the previous section, "Creating a Cloaked Backup (SQL 2000)" on page 146 to create a snapshot that is *not* copy-only.

### ▼ To create a copy-only backup

1. Open an existing batch file in a text editor.
2. Insert the following:

   ```
   COPYONLY TRUE
   ```

3. Save the batch file.

## Creating a PFI Snapshot that is Not Copy-Only

If a persistent frozen image snapshot is selected for backup, NetBackup will automatically create the backup image as copy-only. You can choose *not* to have the backup created as as copy-only.

### ▼ To create a PFI snapshot that is not copy-only

1. Open an existing batch file in a text editor.
2. Insert the following:

   ```
   COPYONLY FALSE
   ```

3. Save the batch file.

# Configuring a Snapshot Backup

Before configuring NetBackup for SQL Server for snapshot backups, review the configuration requirements and configuration steps in the following two sections. Instructions for configuring snapshot policies are covered in "Configuring an Advanced Client Policy" on page 148 and "Configuring a Policy for Instant Recovery" on page 151.

## Configuration Requirements

- See the *NetBackup Advanced Client System Administrator's Guide* for details on the hardware and software requirements for the snapshot method that you want to use. See the VERITAS Technical Support Web site for details on the snapshot methods and platforms that are supported for NetBackup for SQL Server (instructions for accessing this information provided in "Supported Platforms" on page 136).
- It is recommended that the volume(s) which contains the SQL Server databases and log files should be dedicated to SQL Server only. Other types of databases (e.g., Exchange) should not reside on the volume(s).
- NetBackup Advanced Client is installed and configured correctly and the license key for this option has been registered. Refer to the *NetBackup Advanced Client System Administrator's Guide* for details.
- SQL Server 2000 or higher

## Configuration Steps

In order to perform an Advanced Client for SQL Server backup, you must perform the following configuration steps:

❑ Create a backup script (`.bch` file) using the NetBackup for SQL Server interface. See "Using Batch Files" on page 100.

❑ Configure an MS-SQL-Server policy that has the Advanced Client attributes selected. See "Configuring Advanced Client Policies for NetBackup for SQL Server" on page 148.

# Configuring Advanced Client Policies for NetBackup for SQL Server

This section only covers what is necessary to configure snapshot backups for a MS-SQL-Server policy. For information on other policy attributes, creating schedules, adding clients, and adding backup selections, see the “Configuration” chapter earlier in this manual. For information on how a snapshot method is selected automatically and details on the types of snapshot methods, refer to the *NetBackup Advanced Client System Administrator’s Guide*.

Only one snapshot method can be configured per policy. If, for instance, you want to select one snapshot method for clients one group of clients and a different method for another group then you would need to create two policies for each group of clients and select one method for each policy.

- a snapshot backup (with the option of performing an offhost backup)
- an instant recovery backup (to disk and to tape, or, to disk only)

## Configuring an Advanced Client Policy

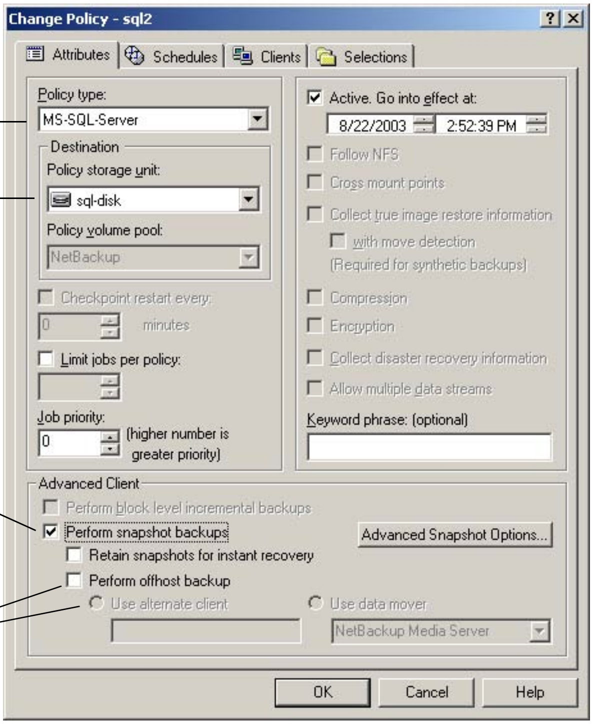
This section describes how to configure an Advanced Client policy. Optionally you can choose to perform an offhost backup.

### ▼ To configure an Advanced Client policy

1. Open the policy you wish to configure.

2. Click on the **Attributes** tab.

3. Select the MS-SQL-Server policy type.



Select the policy type.

Select appropriate storage unit or storage unit group.

Click **Perform snapshot backups**.

(Optional) Click **Perform Offhost backup** and select **Use alternate client**.

Change Policy - sql2

Policy type: MS-SQL-Server

Destination

Policy storage unit: sql-disk

Policy volume pool: NetBackup

□ Checkpoint restart every: 0 minutes

□ Limit jobs per policy:

Job priority: 0 (higher number is greater priority)

☑ Active. Go into effect at: 8/22/2003 2:52:39 PM

□ Follow NFS

□ Cross mount points

□ Collect true image restore information

□ with move detection (Required for synthetic backups)

□ Compression

□ Encryption

□ Collect disaster recovery information

□ Allow multiple data streams

Keyword phrase: (optional)

Advanced Client

□ Perform block level incremental backups

☑ Perform snapshot backups

□ Retain snapshots for instant recovery

□ Perform offhost backup

○ Use alternate client

○ Use data mover

NetBackup Media Server

4. Select a **Policy storage unit**.

If database differentials, filegroup differentials, or transaction logs are included in the Backup Selections list of an Advanced Client policy, then NetBackup will perform a stream-based backup using the selected storage unit. If a storage unit is not provided, then NetBackup will use the default storage unit for the server.

5. Click **Perform snapshot backups**.

6. Choose to have NetBackup select the snapshot method or select the snapshot method manually.

   - By default, NetBackup will choose a snapshot method for you. If you have changed this setting and want NetBackup to choose the method automatically, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose **auto**.

- If you wish to use a specific snapshot method, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose the method you wish to use for this policy.

See the *NetBackup Advanced Client System Administrator's Guide for* details selecting the snapshot method and automatic snapshot selection.

7. To configure schedules, click on the **Schedules** tab.

   Follow the instructions in the "Configuration" chapter earlier in this manual to configure an Application and Automatic schedule.

8. Optional: If you wish to use an alternate client to reduce the processing load on the client, perform the following steps:

   a. The alternate client must be the client that shares the disk array. This option may require additional configuration. Refer to the *NetBackup Advanced Client System Administrator's Guide*.

   b. Select **Perform offhost backup**.

   c. Click **Use alternate client** and enter the name of the alternate client.

**Note** **Use data mover** is not a supported option for NetBackup for SQL Server.

9. Use the **Clients** tab to specify clients to be backed up by this policy.

   Refer to the "Configuration" chapter earlier in this manual for details on adding clients to a policy.

   Note that for NetBackup 5.0, it is no longer possible to configure a snapshot method on a per-policy, per client basis. Snapshot options are applied to all clients controlled by that policy.

10. Use the **Backup Selections** tab to enter the batch files.

11. Configure other attributes, and add the desired schedules and backup selections. Refer to the "Configuration" chapter earlier in this manual for more information.

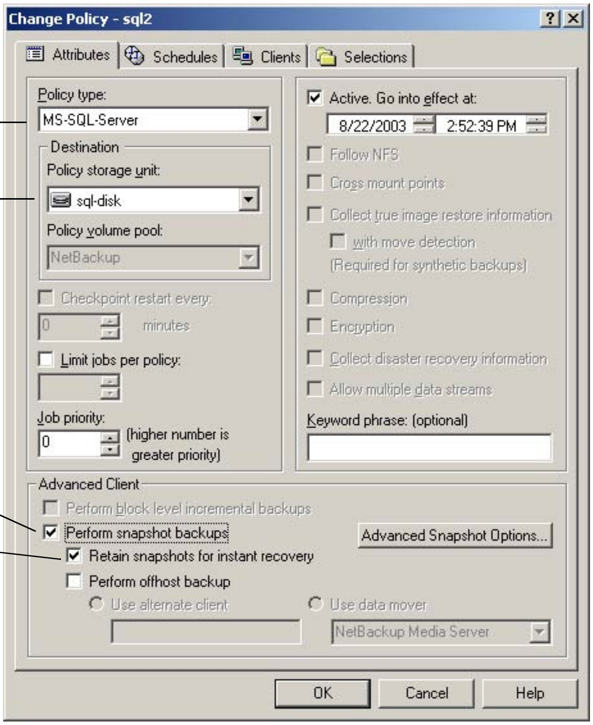
12. Click **OK** to close the dialog.

## Configuring a Policy for Instant Recovery

This section describes how to configure a policy for instant recovery. Optionally you can choose to back up to disk only.

### ▼ To configure a policy for instant recovery

1. Open the policy you wish to configure.
2. Click on the **Attributes** tab.
3. Select the MS-SQL-Server policy type.



Change Policy - sql2

Attributes | Schedules | Clients | Selections

Policy type: MS-SQL-Server

Destination

Policy storage unit: sql-disk

Policy volume pool: NetBackup

□ Checkpoint restart every: 0 minutes

□ Limit jobs per policy:

Job priority: 0 (higher number is greater priority)

☑ Active. Go into effect at: 8/22/2003 2:52:39 PM

□ Follow NFS

□ Cross mount points

□ Collect true image restore information

□ with move detection

(Required for synthetic backups)

□ Compression

□ Encryption

□ Collect disaster recovery information

□ Allow multiple data streams

Keyword phrase: (optional)

Advanced Client

□ Perform block level incremental backups

☑ Perform snapshot backups

Advanced Snapshot Options...

☑ Retain snapshots for instant recovery

□ Perform offhost backup

○ Use alternate client

○ Use data mover

NetBackup Media Server

OK Cancel Help

Select the policy type.

Select appropriate storage unit or storage unit group.

Click **Perform snapshot backups**.

Click here to enable instant recovery.

4. Select a **Policy storage unit**.

   If you select an Instant Recovery option on the **Schedules** tab (see step 8), the storage unit is not used; NetBackup creates only a disk snapshot.

   If database differentials, filegroup differentials, or transaction logs are included in the Backup Selections list of an Advanced Client policy, then NetBackup will perform a stream-based backup using the selected storage unit. If a storage unit is not provided, then NetBackup will use the default storage unit for the server.

5. Click **Perform snapshot backups**.

6. Choose to have NetBackup select the snapshot method or select the snapshot method manually.

   - By default, NetBackup will choose a snapshot method for you. If you have changed this setting and want NetBackup to choose the method automatically, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose **auto**.
   - If you wish to use a specific snapshot method, click **Advanced Snapshot Options** and from the **Snapshot method for this policy** list, choose the method you wish to use for this policy.

   See the *NetBackup Advanced Client System Administrator's Guide for* details selecting the snapshot method and automatic snapshot selection.

7. Select **Retain snapshots for instant recovery**.

   NetBackup retains the snapshot on disk, so that instant recovery can be performed from the snapshot. A normal backup to storage is also performed, if you do not select an Instant Recovery option (see step 8).

8. To configure schedules, click on the **Schedules** tab.

   a. Follow the instructions in the "Configuration" chapter earlier in this manual to configure an Application and Automatic schedule.

   b. Optional: If you wish to create a disk image only, open the Application schedule and select an Instant Recovery option.

      - If **Snapshots and copy snapshots to a storage unit** is selected, NetBackup creates a disk snapshot and backs up the client's data to the storage unit specified for the policy.
      - If **Snapshots only** is selected, the image is not backed up to tape or other storage. NetBackup creates a disk snapshot only. Note that this disk snapshot is not considered a replacement for traditional backup.

**Note** The **Policy storage unit** is ignored when making a disk image only.

9. Use the **Clients** tab to specify clients to be backed up by this policy.

   Refer to the “Configuration” chapter earlier in this manual for details on adding clients to a policy.

   Note that for NetBackup 5.0, it is no longer possible to configure a snapshot method on a per-policy, per client basis. Snapshot options are applied to all clients controlled by that policy.

10. Use the **Backup Selections** tab to enter the batch files.

11. Configure other attributes, and add the desired schedules and backup selections. Refer to the “Configuration” chapter earlier in this manual for more information.

12. Click **OK** to close the dialog.

# Other Applications of NetBackup for SQL Server 9

NetBackup for SQL Server can also be used to perform backups and restores in an SAP environment and to restore backups performed by Backup Exec.

# Performing Backups and Restores in a SAP Environment

With NetBackup you can perform scheduled SAP backups, in accordance with a predefined backup strategy, or manual backups that are not planned and may be necessary in exceptional situations. The practices described here are based on the practices recommended by SAP in SAP/MS SQL Server 2000 DBA in CCMS.

The NetBackup backup and restore procedures for the SAP R/3 database are identical to the NetBackup procedures with any other SQL Server database.

NetBackup for SQL Server allows you to create scripts in order to perform full or differential backups of databases and backups of transaction logs. In addition to the database backups and restores, NetBackup also provides the capabilities to backup the SAP file systems.

### Manual Backups

The administrator on the master server can use the NetBackup Administration Console to manually execute an automatic backup schedule for an “MS-SQL-Server” policy, where the R/3 database is specified in the backup script. For more information, see the section on manual backups in the *Netbackup System Administrator Guide for Windows*.

## Policy Configuration

In order to automatically perform backups of a SAP environment, you need to create backup policies. A backup policy with the “MS-SQL-Server” policy type selected must be created for R/3 database backups. Batch files, which will initiate the backup of the database and transaction logs, must be added to the Backup Selections list in the policy. For information on creating the batch files needed, see “Creating Batch Files to be Used in Automatic Backup Schedules” below. For information on MS-SQL-Server policies, see the instructions in “Configuring Backup Policies” on page 13.

For backups of the executables disk (a file-system backup), a backup policy must be created with the “MS-Windows-NT” policy type selected. For information on MS-Windows-NT policies, see the *Netbackup System Administrator Guide for Windows, Volume 1*.

## Creating Batch Files to be Used in Automatic Backup Schedules

NetBackup for SQL Server uses batch files to initiate database backup and restore operations. A batch file must be created for database backups and for transaction log backups. These batch files must then be added to the Selections list in the backup policies you've created.

### ▼ To create a script for database backups

1. From the Windows **Start** menu, choose **Programs** > **NetBackup** > **NetBackup Agents** > **NetBackup MS SQL Client**.

   The NetBackup for Microsoft SQL Server Graphical User Interface displays.

2. Choose **File** > **Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog box displays.

3. Select the R/3 database in the **Database** list.

4. Choose the desired **Backup type**, **Full** or **Full differential**.

   See "Backup Microsoft SQL Server Objects" on page 200 for a description of the available backup options.

5. Click **OK**.

6. When prompted to start the backup, click **Yes**.

7. Check that each backup operation completed successfully by choosing **File** > **View status**. See "Monitoring Backups" on page 158 for more information.

### ▼ To create a script for transaction log backups

1. Before starting a transaction log backup, the database administrator should set the **Transaction Log Backup Options** database option to off. This is an option on the SQL Server interface that applies to the databases.

   The entire sequence of transaction logs generated following any database dump must be maintained on the same NetBackup server. NetBackup for SQL Server requires that you follow these guidelines in devising your backup strategy in order to ensure success in restoring your database.

2. Choose **File** > **Backup SQL Server Objects**.

   The Backup Microsoft SQL Server Objects dialog appears.

3. Select the R/3 in the **Database** drop-down list.

4. For the **Backup type**, select **Backup transaction log**.

   See "Backup Microsoft SQL Server Objects" on page 200 for a description of the available backup options.

5. Click **OK**.

6. When prompted to start the backup, click **Yes**.

7. Check that each backup operation completed successfully by choosing **File** > **View status**. See "Monitoring Backups" on page 158 for more information.

## Monitoring Backups

Scheduled backups should be checked regularly to ensure they were completed successfully.

Always check that:

- The most recent backup has run successfully. See "Progress Reports Created for NetBackup for SQL Server on the Client" on page 184.
- All the backups in the backup cycle are being executed according to the schedule. Gaps in a backup sequence can have serious consequences in a subsequent attempt to restore the database.

## Restoring the R/3 Database

This section describes how to restore the R/3 database.

- If you have scheduled *differential* backups, review the information in next section "Including Differential Backups in a Restore Operation." .
- If the R/3 database disk system is damaged or the transaction log disk system is damaged, follow the instructions in "Restoring the R/3 Database after a Disk Crash" on page 159.
- To perform a regular restore of the R/3 database, follow the instructions in "Restoring the Database and Transaction Log Backups." .

### Including Differential Backups in a Restore Operation

If you incorporated *differential backups* in the backup strategy, the restore process differs depending on the type of backups available.

- If differential backups were made after the last full database backup, restore the last database backup followed by the most recent differential backup. Then apply all subsequent transaction logs.
- If no differential backups were made since the last full database backup, restore the last full database backup and then apply all subsequent transaction logs.
- If several differential backups are available, but the latest one cannot be read.

  Restore the most recent full database backup and then you restore the latest readable differential backup. Then apply all subsequently created transaction logs.

### Restoring the R/3 Database after a Disk Crash

This section describes how to restore the database when the R/3 database disk system is damaged or the transaction log disk system is damaged. This process is only applicable to a configuration with three disk systems: one system for the R/3 database, one for the R/3 transaction logs and one for all others.

**Note** The R3 database must not be in use when performing a restore operation. Make sure that all SAP services are stopped before attempting a restore using NetBackup.

**▼ To restoring the R/3 database after a disk crash**

1. Back up the current transaction log.

   *If the disk system on which the R/3 database resides is damaged, it is vital to immediately backup the currently active transaction log to prevent loss of data.* Without a backup of the current log, the database can only be restored to the status at the time of the last transaction log backup. If work has been carried out on the R/3 system since then, this work will be lost.

2. Replace damaged disks.

   Replacing damaged disks in a RAID disk system is normally a straightforward procedure. If you are uncertain how to proceed, refer to the documentation of your hardware vendor to find out how to handle the disks. The new disks must be formatted and assigned the same drive letter as the old disks.

3. Restore the database and transaction logs.

   The central phase of a restore operation is the reloading of the database backup and the application of the available transaction logs. When the database backup is reloaded, the database files are automatically recreated and the data is copied from the backup device to the newly created files. Once this has been done, the transaction logs are applied in the same sequence as they were originally made. In a final step, open transactions that were not completed at the time of the database failure are rolled back.

## Restoring the Database and Transaction Log Backups

NetBackup MS-SQL Server Agent GUI provides for automatic staging. By selecting the latest transaction log backup, the GUI automatically restores the previous full database backup along with any optional differential backups and subsequent transaction log backups. There is also the option to specify a point in time to which to restore to.

**Note** The R3 database must not be in use when performing a restore operation. Make sure that all SAP services are stopped before attempting a restore using NetBackup.

**Caution** To restore the R/3 database you first restore the most recent database backup and then the subsequent transaction logs. During the entire procedure, do not execute any transactions and do not shut down the database server. A server shutdown would write a checkpoint to the log and, as a result, you would not be able to restore further transaction logs.

### ▼ To restore the database and transaction log backups

1. Restore the most recent database backup.

2. Restore the latest differential database backup (if available).

3. Restore all succeeding transaction log backups.

4. Restore the latest transaction log backup.

# Using NetBackup to Restore SQL Server from Backup Exec Images

NetBackup can restore SQL Server from Backup Exec images, using the NetBackup Backup, Archive, and Restore (NetBackup Client) interface.

## Requirements for Restoring Backup Exec Images

- In order to successfully restore Backup Exec images, the Backup Exec media needs to be imported into NetBackup by first running `vmphyinv`, and then running `bpimport`, Phase 1 and Phase 2. Refer to the instructions for importing images from Backup Exec media in the *NetBackup System Administrator's Guide for Windows, Volume I*. NetBackup can restore Backup Exec images of SQL 7.0 and above.
- The NetBackup Client Service should be running in a user account that has been granted the System Administrator role on the target SQL instance. See "SQL Server Privileges" on page 31 for details on configuring the logon account for this service.

## Limitations When Restoring Backup Exec Images

- Restoring multiple databases in a single restore job is not supported. It is however possible to restore full, differential and transaction log backups of a same database in a single restore job.
- Table backups cannot be restored, though they are included in the backup image and can be selected for restore. If you attempt to restore a table, the job will fail with the error "unsupported object was selected for restore".

# Restoring from Backup Exec Images

This section describes the restore options available and the procedures for restoring SQL Server from Backup Exec images.

## Restore Options

The following options are available when restoring from SQL backups from Backup Exec images.

Restore options on the Microsoft SQL Server tab

| Item | Description |
|---|---|
| **Recovery completion state** | |
| **Leave database operational. No additional transaction logs can be restored.**<br>(With Recovery) | Select this option when restoring the last database, differential, or log backup in the restore sequence in order to have the restore operation roll back all uncompleted transactions. After the recovery operation, the database is ready for use. If Leave database operational is not performed, the database is left in an intermediate state and is not usable.<br>If Leave database operational is selected when an intermediate backup is being applied, you cannot continue to restore backups.<br>You must restart the restore operation from the beginning. |
| **Leave database nonoperational but able to restore additional transaction logs**<br>(No Recovery) | Select this option during a restore if you have additional differential or transaction log backups to be restored in another restore job. |
| **Leave database read-only and able to restore additional transaction logs**<br>(Standby) | Select this option during transaction log and database restore to create and maintain a standby database. See your SQL documentation for information on standby databases. |
| **Replace databases or filegroups** | Select this checkbox to replace a database or filegroup, even if another database or filegroup with the same name already exists on the server. If Replace Databases or Filegroups is not specified for a restore, SQL performs a safety check to ensure that a different database or filegroup is not accidentally overwritten. Refer to your SQL documentation for more information about the safety check that occurs when the REPLACE option is not selected. |

Restore options on the Microsoft SQL Server tab

| Item | Description |
|---|---|
| **Automate master database restore** | Enable NetBackup to stop SQL so that the master database can be restored. All existing users are logged off, and SQL Server is put into single-user mode.<br>When this option is selected, only the master database can be restored; if this option is selected for any other database, those jobs will fail.<br>If NetBackup does not have access to the SQL registry keys, HKEY_LOCAL_MACHINE\Software\Microsoft\ Microsoft SQL Server, and HKEY_LOCAL_MACHINE\Software\Microsoft\MSSQLServer , then a restore to the default directory may not work, and the option **Automate master database restore** will not work. To ensure that NetBackup has access rights, verify that the user account of the NetBackup client service has administrator rights to the Windows server that the SQL instance is installed on. |
| **Consistency check after restore** | ◆ **Full check, excluding indexes**. Select this to exclude indexes from the consistency check. If indexes are not checked, the consistency check runs significantly faster but is not as thorough. Only the data pages and clustered index pages for each user table are included in the consistency check. The consistency of the nonclustered index pages is not checked.<br>◆ **Full check, including indexes**. Select this to include indexes in the consistency check. Any errors are logged. This option is selected by default.<br>◆ **Physical check only (SQL 2000 only)**. Select this to perform a low overhead check of the physical consistency of the SQL 2000 database. This option only checks the integrity of the physical structure of the page and record headers, and the consistency between the pages' object ID and index ID and the allocation structures.<br>◆ **None**. Select this if you are doing sequential restores. Do not run a consistency check after a restore until all sequential restores have been done. If a consistency check is selected during a restore, the restore will complete but the consistency check will not be done. Check the job log for this information.<br>◆ If you need to recover the database after restores are complete, select one of the following consistency checks when you select the **Leave database operational** option. |

Restore options on the Microsoft SQL Server tab

| Item | Description |
|---|---|
| **Note** Do not select the **Alternate drive** option when restoring filegroups. Filegroups must be restored to the same drive letter and path that they were backed up from. | |
| **Alternate drive for restoring database files** | Use this option to select a drive to which SQL database files can be restored if the drive where one or more of the database files previously resided no longer exists.<br>When a SQL database is backed up, the physical file names (which include the directory path) of the files that make up the database are stored in the backup set by SQL. For example, for the logical file pubs, the physical file name is stored as E:\MSSQL7\DATA\pubs.mdf. If the database must later be restored, SQL uses these same physical file names to target the restore to. During a restore, NetBackup automatically creates any necessary subdirectories that do not exist.<br>However, if the drive where one or more of the database files previously resided no longer exists, NetBackup moves those files to their original directory path, but on the alternate drive specified.<br>Using the same example, if drive C: is specified, then the file with the original directory path of E:\MSSQL7\DATA\pubs.mdf is restored to C:\MSSQL7\DATA\pubs.mdf.<br>If no alternate drive is specified in this situation, the job will fail. |
| Restore to alternate drive | |
| **Only when original drive does not exist** | Select this option to restore all database files to their original directory path on the alternate drive, only if the drive where they originally resided exists. To make this option available, select a drive letter in **Alternate drive for restoring database files** list. |
| **Even when original drive does exist** | Select this option to restore all database files to their original directory path on the alternate drive, even if the drive where they originally resided exists. To make this option available, select a drive letter in **Alternate drive for restoring database files** list. |

Restore options on the Microsoft SQL Server tab

| Item | Description |
|---|---|
| **Note** | Do not select the **Restore all database files to the target instance's data location** option when restoring filegroups. Filegroups must be restored to the same drive letter and path that they were backed up from. |
| **Restore all database files to the target instance's data location** | Select this checkbox to restore files to the default data and log directories of the destination instance. For example, if you are restoring a database to a different instance of SQL, you would select this option to move the database files to the correct location for the new instance.<br>If this option is not selected, then the files are restored to the directory that the master database is in. |
| **Point in time log restore** | Select this checkbox to restore transactions from a transaction log up to and including a point in time in the transaction log. After the point in time, recovery from the transaction log is stopped.<br>Select the part of the date you want to change, and then enter a new date or click the arrow to display a calendar from which you can select a date.<br>Select the part of the time you want to change, and then enter a new time or click the arrows to select a new time. |
| **Restore log up to named transaction (SQL 2000)** | Select this checkbox to restore transactions from a transaction log up to a named transaction (or named mark) in the transaction log; after that, recovery from the transaction log is stopped. The named transactions are case-sensitive. |
| **Include the named transaction** | Select this checkbox to include the named transaction in the restore; otherwise the restore will stop immediately before the named transaction is restored. |
| **Found after** | Select this checkbox to specify a date and time after which the restore operation is to search for the named transaction. For example, if you specify a restore from a log up to the named transaction *AfternoonBreak*, found after 6/02/2003, 12:01 p.m., then the restore operation will not search for *AfternoonBreak* until after that time. |

## Restore Options for Redirected Restores

Restores options on the Microsoft SQL Server Redirection tab

| Item | Description |
|---|---|
| **Redirect Microsoft SQL Server sets** | Select this checkbox to enable redirection of SQL backup sets. |
| **SQL Logon Account User Name** | If SQL Server Authentication is being used, enter the logon account that stores the credentials of the SQL user account. You do not need to provide a user name and password if integrated security is being used. See "SQL Server Privileges" on page 31 for more information on standard and integrated security. |
| **SQL Logon Account Password** | Indicate the password associated with logon account. |
| **SQL Redirection** | To redirect this restore to a named instance, type the name of the server and the instance name, in the format "server_name instance_name". For example, **`aj civolus`**.<br>If you are redirecting the default instance, type the name of the server, in the format "server_name". For example, **aj**. |
| **Restore to Database** | To redirect the restore to a different database on the target server, type the target database name; otherwise, leave the field blank.<br>You can redirect a full database backup to a different server and/or database; however, if the drive configuration is different from when the database backup was created, you must select either **Alternate drive for restoring database files** or select **Restore all database files to the target instance's data location** on the Microsoft SQL Server tab. See "Restore Options" on page 162.<br>If you are restoring a differential or log backup, and the associated database backup was restored to a different server, enter the new database name. |

## Specifying the Server, Client, and Policy Type

In order to browse for backups of SQL databases backed up by Backup Exec, you must first specify the server containing the backup images, the client that performed the backups, and the type of policy associated with the backups.

### ▼ To choose the server, client, and policy type

1. Choose **File** > **Specify NetBackup Machines and Policy Type**.

   The Specify NetBackup Machines and Policy Type dialog is displayed.

2. From the **Server to use for backups and restores** list, select the desired server.

3. From the **Source client for restores** list, select the desired client.

   The source client is the Exchange Server machine name whose backup images you would like to browse.

4. From the **Policy type for restores** list, choose **MS-SQL-Server**.

5. If you wish to redirect a restore to a different client, select the client to which to redirect the restore from the **Destination clients for restores** list.

   See "Redirecting a Restore" on page 181 for additional information on redirecting a Backup Exec restore to a different location.

6. Click **OK**.

   NetBackup browses for SQL Server backup images.

   The NetBackup History pane displays SQL Server backup information. The top split windows show individual image information and the bottom split gives file and folder information and also allows the user to select what files are to be restored.

## Restoring Backup Exec Database Backups

If the database you wish to restore is using the simple recovery model, there are no transaction log backups to restore. You only need to restore the most recent full database backup and if you were running differential database backups, restore the most recent differential database backup.

### ▼ To restore a database backup

1. Log on as Administrator.

2. Open the Backup, Archive, and Restore interface.

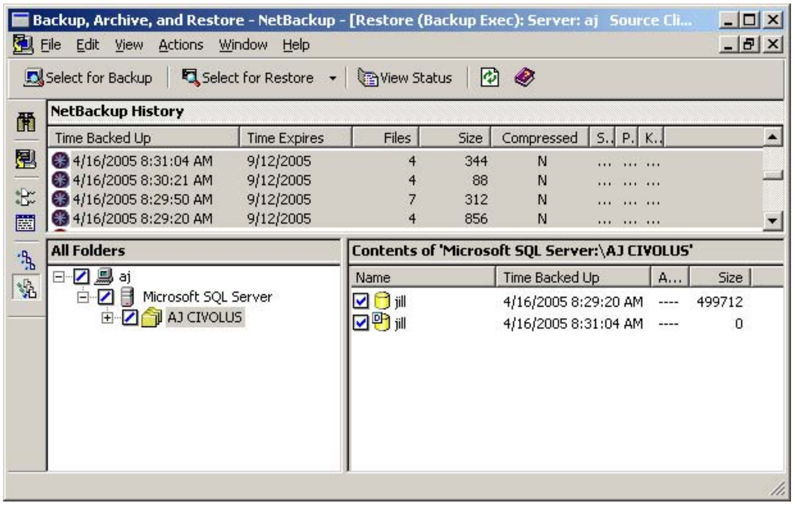
3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 167.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you wish to restore.

6. Select the most recent full database backup, and the most recent differential database backup, if any, to restore.



NetBackup History

| Time Backed Up | Time Expires | Files | Size | Compressed | S.. | P.. | K.. |
|---|---|---|---|---|---|---|---|
| 4/16/2005 8:31:04 AM | 9/12/2005 | 4 | 344 | N | ... | ... | ... |
| 4/16/2005 8:30:21 AM | 9/12/2005 | 4 | 88 | N | ... | ... | ... |
| 4/16/2005 8:29:50 AM | 9/12/2005 | 7 | 312 | N | ... | ... | ... |
| 4/16/2005 8:29:20 AM | 9/12/2005 | 4 | 856 | N | ... | ... | ... |

Contents of 'Microsoft SQL Server:\AJ CIVOLUS'

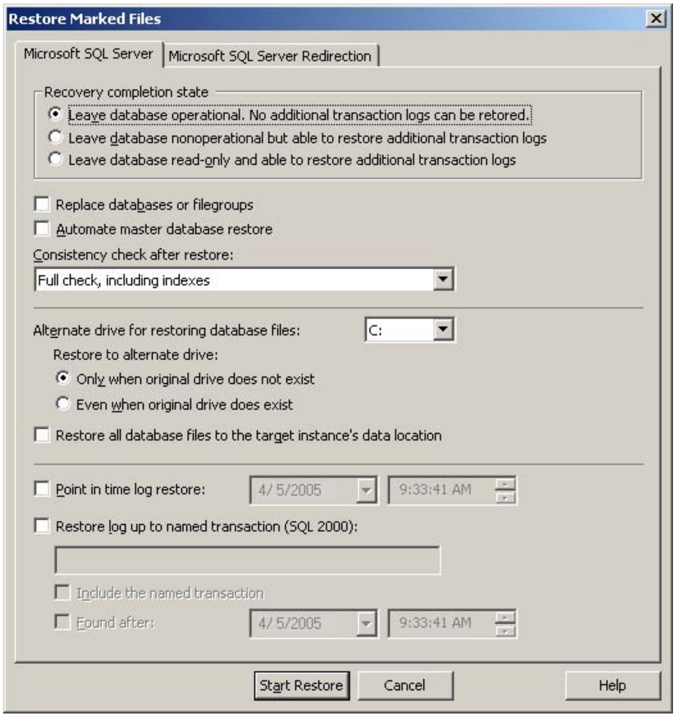
| Name | Time Backed Up | A... | Size |
|---|---|---|---|
| ☑ jill | 4/16/2005 8:29:20 AM | ---- | 499712 |
| ☑ jill | 4/16/2005 8:31:04 AM | ---- | 0 |

7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. From the Recovery completion state group, choose **Leave database operational. No additional transaction logs can be restored.**

   For information on other SQL restore options, refer to “Restore Options” on page 162.



Restore Marked Files

Microsoft SQL Server | Microsoft SQL Server Redirection

Recovery completion state

● Leave database operational. No additional transaction logs can be retored.

○ Leave database nonoperational but able to restore additional transaction logs

○ Leave database read-only and able to restore additional transaction logs

☐ Replace databases or filegroups

☐ Automate master database restore

Consistency check after restore: Full check, including indexes

Alternate drive for restoring database files: C:

Restore to alternate drive:

● Only when original drive does not exist

○ Even when original drive does exist

☐ Restore all database files to the target instance's data location

☐ Point in time log restore: 4/ 5/2005 9:33:41 AM

☐ Restore log up to named transaction (SQL 2000):

☐ Include the named transaction

☐ Found after: 4/ 5/2005 9:33:41 AM

9. Click **Start Restore**.

## Restoring Backup Exec SQL Transaction Logs Backups Up to a Point in Time

You can restore transactions from a transaction log up to and including a point in time in the transaction log. After the point in time is reached, recovery from the transaction log is stopped.

### ▼ To restore SQL transaction logs up to a point in time

1. Log on as Administrator.

2. Open the Backup, Archive, and Restore interface.

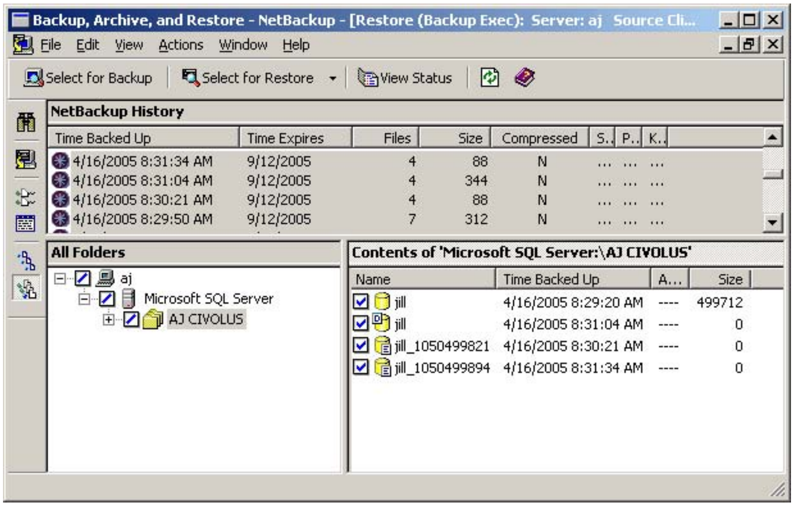
3. Specify the appropriate server, client, and policy type, as described in “Specifying the Server, Client, and Policy Type” on page 167.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you wish to restore.

6. In the All Folders pane, select the most recent full database backup, and the most recent differential database backup, if any, and all the log backups you want to restore.



NetBackup History

| Time Backed Up | Time Expires | Files | Size | Compressed | S.. | P.. | K.. |
|---|---|---|---|---|---|---|---|
| 4/16/2005 8:31:34 AM | 9/12/2005 | 4 | 88 | N | ... | ... | ... |
| 4/16/2005 8:31:04 AM | 9/12/2005 | 4 | 344 | N | ... | ... | ... |
| 4/16/2005 8:30:21 AM | 9/12/2005 | 4 | 88 | N | ... | ... | ... |
| 4/16/2005 8:29:50 AM | 9/12/2005 | 7 | 312 | N | ... | ... | ... |

Contents of 'Microsoft SQL Server:\AJ CIVOLUS'

| Name | Time Backed Up | A... | Size |
|---|---|---|---|
| ☑ jill | 4/16/2005 8:29:20 AM | ---- | 499712 |
| ☑ jill | 4/16/2005 8:31:04 AM | ---- | 0 |
| ☑ jill_1050499821 | 4/16/2005 8:30:21 AM | ---- | 0 |
| ☑ jill_1050499894 | 4/16/2005 8:31:34 AM | ---- | 0 |

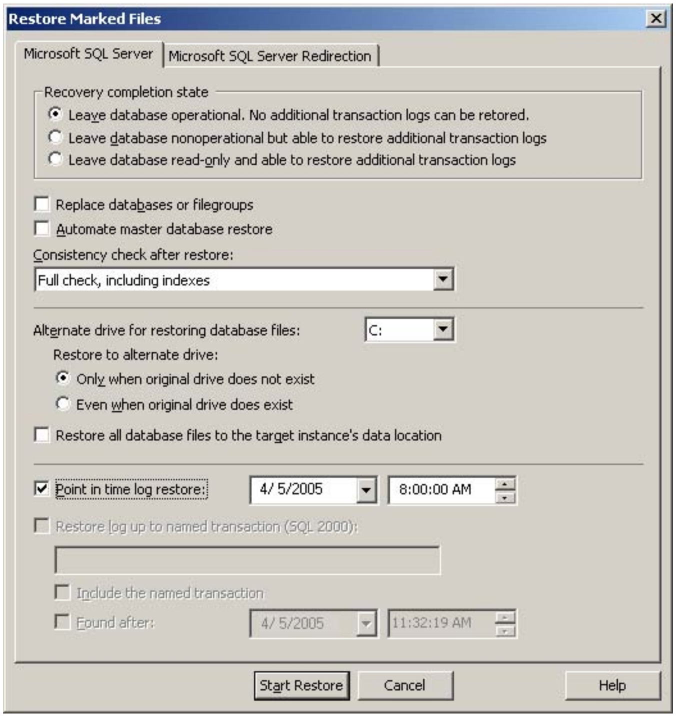
7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. From the Recovery completion state group, choose **Leave database operational. No additional transaction logs can be restored.**

9. Select **Point in time log restore**, and then select a date and time.

   For information on other SQL restore options, refer to “Restore Options” on page 162.



Restore Marked Files

Microsoft SQL Server | Microsoft SQL Server Redirection

Recovery completion state

● Leave database operational. No additional transaction logs can be retored.

○ Leave database nonoperational but able to restore additional transaction logs

○ Leave database read-only and able to restore additional transaction logs

□ Replace databases or filegroups

□ Automate master database restore

Consistency check after restore: Full check, including indexes

Alternate drive for restoring database files: C:

Restore to alternate drive:

● Only when original drive does not exist

○ Even when original drive does exist

□ Restore all database files to the target instance's data location

☑ Point in time log restore: 4/ 5/2005 8:00:00 AM

□ Restore log up to named transaction (SQL 2000):

□ Include the named transaction

□ Found after: 4/ 5/2005 11:32:19 AM

10. Click **Start Restore**.

## Restoring Backup Exec SQL Transaction Logs Backups Up to a Named Transaction

When restoring SQL 2000 transaction logs, you can restore transactions from a transaction log up to and including a named transaction (or mark). After the named transaction is reached, recovery from the transaction log is stopped.

### ▼ To restore a transaction log up to a named transaction

1. Log on as Administrator.

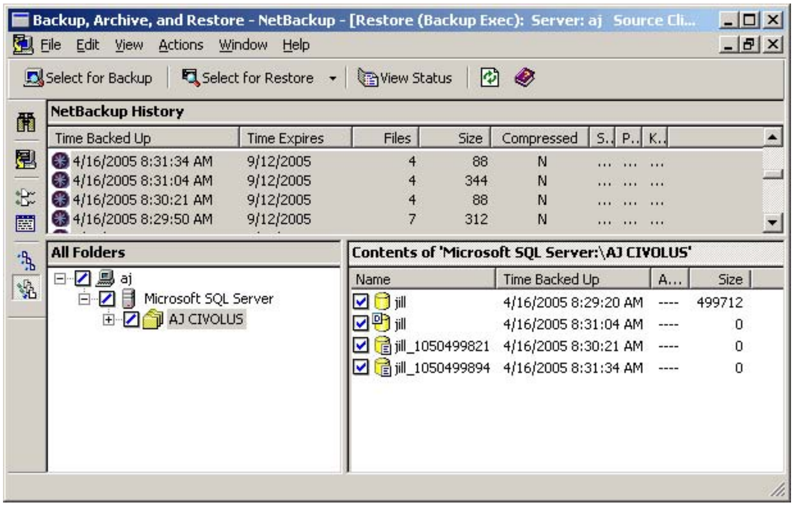
2. Open the Backup, Archive, and Restore interface.

3. Specify the appropriate server, client, and policy type, as described in “Specifying the Server, Client, and Policy Type” on page 167.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you In the All Folders pane, select the most recent full database backup, and the most recent differential database backup, if any, and all the log backups you want to restore.


Backup, Archive, and Restore - NetBackup - [Restore (Backup Exec): Server: aj Source Cli...
File Edit View Actions Window Help
Select for Backup
Select for Restore
View Status
NetBackup History
All Folders
aj
Microsoft SQL Server
AJ CIVOLUS

| Time Backed Up | Time Expires | Files | Size | Compressed | S.. | P.. | K.. |
|---|---|---|---|---|---|---|---|
| 4/16/2005 8:31:34 AM | 9/12/2005 | 4 | 88 | N | ... | ... | ... |
| 4/16/2005 8:31:04 AM | 9/12/2005 | 4 | 344 | N | ... | ... | ... |
| 4/16/2005 8:30:21 AM | 9/12/2005 | 4 | 88 | N | ... | ... | ... |
| 4/16/2005 8:29:50 AM | 9/12/2005 | 7 | 312 | N | ... | ... | ... |

Contents of 'Microsoft SQL Server:\AJ CIVOLUS'

| Name | Time Backed Up | A... | Size |
|---|---|---|---|
| ☑ jill | 4/16/2005 8:29:20 AM | ---- | 499712 |
| ☑ jill | 4/16/2005 8:31:04 AM | ---- | 0 |
| ☑ jill_1050499821 | 4/16/2005 8:30:21 AM | ---- | 0 |
| ☑ jill_1050499894 | 4/16/2005 8:31:34 AM | ---- | 0 |

6. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

7. From the Recovery completion state group, choose **Leave database operational. No additional transaction logs can be restored.**

8. Select **Restore log up to named transaction**, and then enter the name of the transaction.

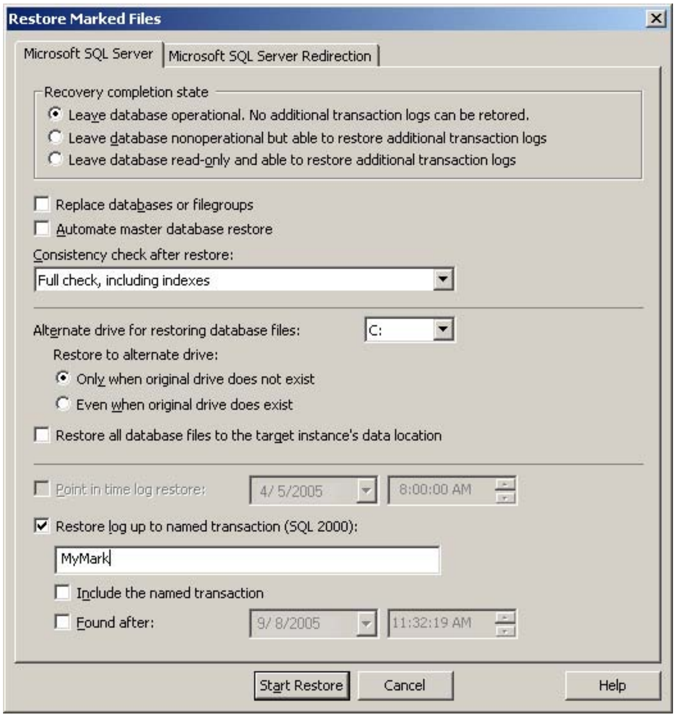
   The names are case-sensitive. Make sure you enter the correct upper- and lower-case characters.

9. To include the named transaction in the restore, select **Include the named transaction**.

10. To specify a particular named transaction in the log, select **Found after** and then select a date and time.

    If a date and time are not entered, recovery from the transaction log is stopped at the first transaction with the specified name.

For information on other SQL restore options, refer to “Restore Options” on page 162.



Restore Marked Files

Microsoft SQL Server | Microsoft SQL Server Redirection

Recovery completion state

● Leave database operational. No additional transaction logs can be retored.

○ Leave database nonoperational but able to restore additional transaction logs

○ Leave database read-only and able to restore additional transaction logs

□ Replace databases or filegroups

□ Automate master database restore

Consistency check after restore: Full check, including indexes

Alternate drive for restoring database files: C:

Restore to alternate drive:

● Only when original drive does not exist

○ Even when original drive does exist

□ Restore all database files to the target instance's data location

□ Point in time log restore: 4/ 5/2005 8:00:00 AM

☑ Restore log up to named transaction (SQL 2000): MyMark

□ Include the named transaction

□ Found after: 9/ 8/2005 11:32:19 AM

**11.** Click **Start Restore**.

## Restoring Backup Exec Filegroup Backups

With filegroup backups, you can restore the entire database, a primary filegroup, a filegroup containing a deleted or changed table, and a nonprimary filegroup.

### ▼ To restore a filegroup

**Note** Use separate restore jobs to restore the primary filegroup, the rest of the filegroup backup sets, and the transaction logs.

**1.** Log on as Administrator.

**2.** Open the Backup, Archive, and Restore interface.

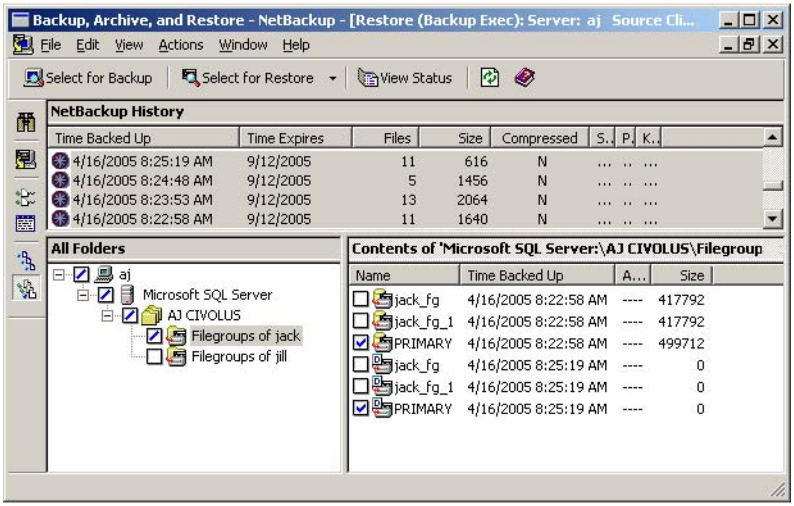
**3.** Specify the appropriate server, client, and policy type, as described in “Specifying the Server, Client, and Policy Type” on page 167.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the objects you wish to restore.

6. In the All Folders pane, select the full backup of the primary filegroup and any differential backups.


Backup, Archive, and Restore - NetBackup - [Restore (Backup Exec): Server: aj Source Cli...
File Edit View Actions Window Help
Select for Backup
Select for Restore
View Status
All Folders
aj
Microsoft SQL Server
AJ CIVOLUS
Filegroups of jack
Filegroups of jill

NetBackup History

| Time Backed Up | Time Expires | Files | Size | Compressed | S.. | P. | K.. |
|---|---|---|---|---|---|---|---|
| 4/16/2005 8:25:19 AM | 9/12/2005 | 11 | 616 | N | ... | .. | ... |
| 4/16/2005 8:24:48 AM | 9/12/2005 | 5 | 1456 | N | ... | .. | ... |
| 4/16/2005 8:23:53 AM | 9/12/2005 | 13 | 2064 | N | ... | .. | ... |
| 4/16/2005 8:22:58 AM | 9/12/2005 | 11 | 1640 | N | ... | .. | ... |

Contents of 'Microsoft SQL Server:\AJ CIVOLUS\Filegroup

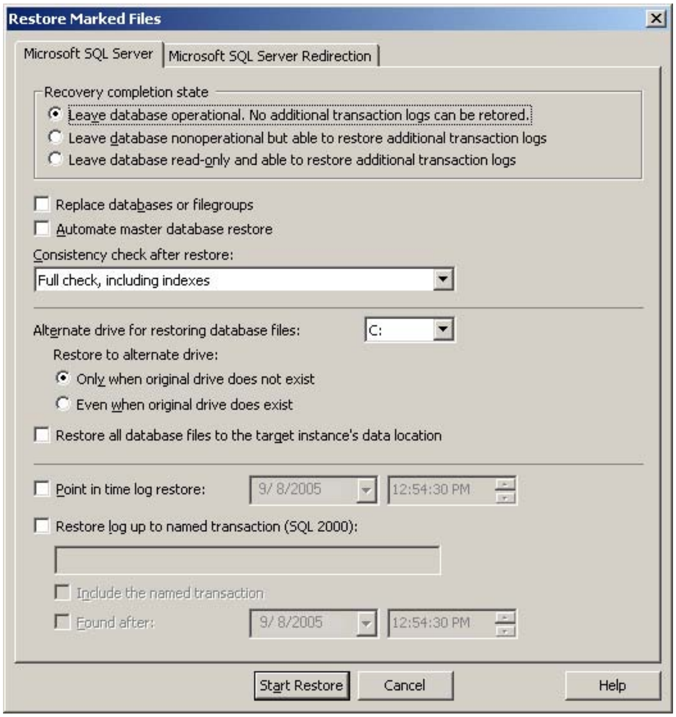
| Name | Time Backed Up | A... | Size |
|---|---|---|---|
| □ jack_fg | 4/16/2005 8:22:58 AM | ---- | 417792 |
| □ jack_fg_1 | 4/16/2005 8:22:58 AM | ---- | 417792 |
| ☑ PRIMARY | 4/16/2005 8:22:58 AM | ---- | 499712 |
| □ jack_fg | 4/16/2005 8:25:19 AM | ---- | 0 |
| □ jack_fg_1 | 4/16/2005 8:25:19 AM | ---- | 0 |
| ☑ PRIMARY | 4/16/2005 8:25:19 AM | ---- | 0 |

7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. From the Recovery completion state group, choose **Leave database nonoperational but able to restore additional transaction logs**.

   For information on other SQL restore options, refer to “Restore Options” on page 162.



Recovery completion state

● Leave database operational. No additional transaction logs can be retored.

○ Leave database nonoperational but able to restore additional transaction logs

○ Leave database read-only and able to restore additional transaction logs

□ Replace databases or filegroups

□ Automate master database restore

Consistency check after restore: Full check, including indexes

Alternate drive for restoring database files: C:

Restore to alternate drive:

● Only when original drive does not exist

○ Even when original drive does exist

□ Restore all database files to the target instance's data location

□ Point in time log restore: 9/ 8/2005 12:54:30 PM

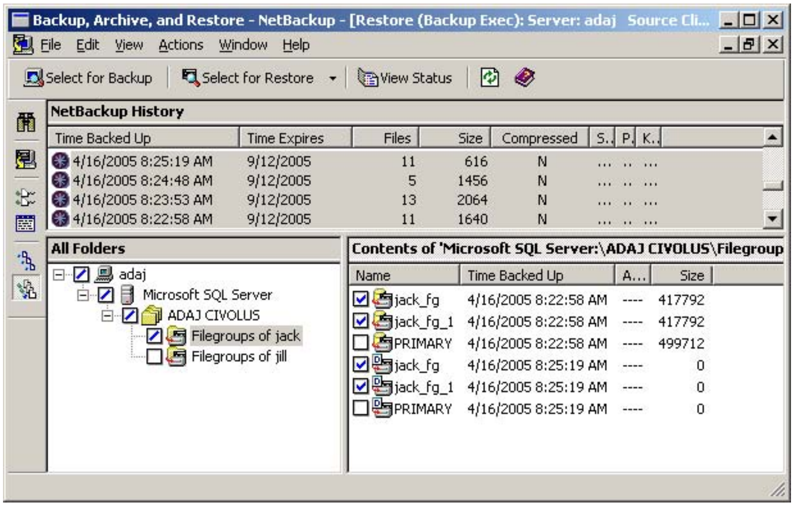
□ Restore log up to named transaction (SQL 2000):

□ Include the named transaction

□ Found after: 9/ 8/2005 12:54:30 PM

9. Click **Start Restore**.

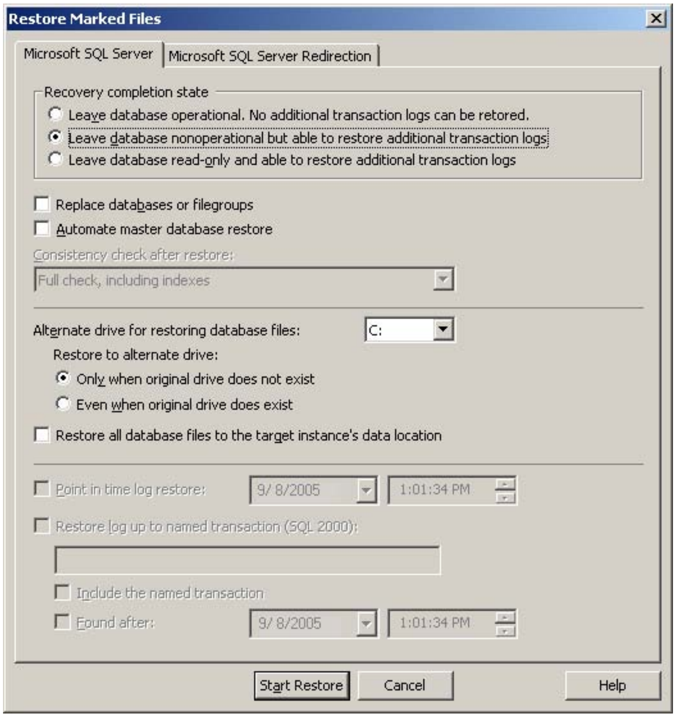
10. After the primary filegroup is restored, select the latest full and differential backups for the other filegroups.


Backup, Archive, and Restore - NetBackup - [Restore (Backup Exec): Server: adaj Source Cli...
File Edit View Actions Window Help
Select for Backup
Select for Restore
View Status
NetBackup History
All Folders
adaj
Microsoft SQL Server
ADAJ CIVOLUS
Filegroups of jack
Filegroups of jill
Contents of 'Microsoft SQL Server:\ADAJ CIVOLUS\Filegroup

| Time Backed Up | Time Expires | Files | Size | Compressed | S.. | P. | K.. |
|---|---|---|---|---|---|---|---|
| 4/16/2005 8:25:19 AM | 9/12/2005 | 11 | 616 | N | ... | .. | ... |
| 4/16/2005 8:24:48 AM | 9/12/2005 | 5 | 1456 | N | ... | .. | ... |
| 4/16/2005 8:23:53 AM | 9/12/2005 | 13 | 2064 | N | ... | .. | ... |
| 4/16/2005 8:22:58 AM | 9/12/2005 | 11 | 1640 | N | ... | .. | ... |

| Name | Time Backed Up | A... | Size |
|---|---|---|---|
| ☑ jack_fg | 4/16/2005 8:22:58 AM | ---- | 417792 |
| ☑ jack_fg_1 | 4/16/2005 8:22:58 AM | ---- | 417792 |
| □ PRIMARY | 4/16/2005 8:22:58 AM | ---- | 499712 |
| ☑ jack_fg | 4/16/2005 8:25:19 AM | ---- | 0 |
| ☑ jack_fg_1 | 4/16/2005 8:25:19 AM | ---- | 0 |
| □ PRIMARY | 4/16/2005 8:25:19 AM | ---- | 0 |

11. On the **Microsoft SQL Server** tab, select the recovery completion state **Leave database nonoperational but able to restore additional transaction logs**, and then start the restore job.



Restore Marked Files

Microsoft SQL Server | Microsoft SQL Server Redirection

Recovery completion state

○ Leave database operational. No additional transaction logs can be retored.
● Leave database nonoperational but able to restore additional transaction logs
○ Leave database read-only and able to restore additional transaction logs

□ Replace databases or filegroups
□ Automate master database restore

Consistency check after restore: Full check, including indexes

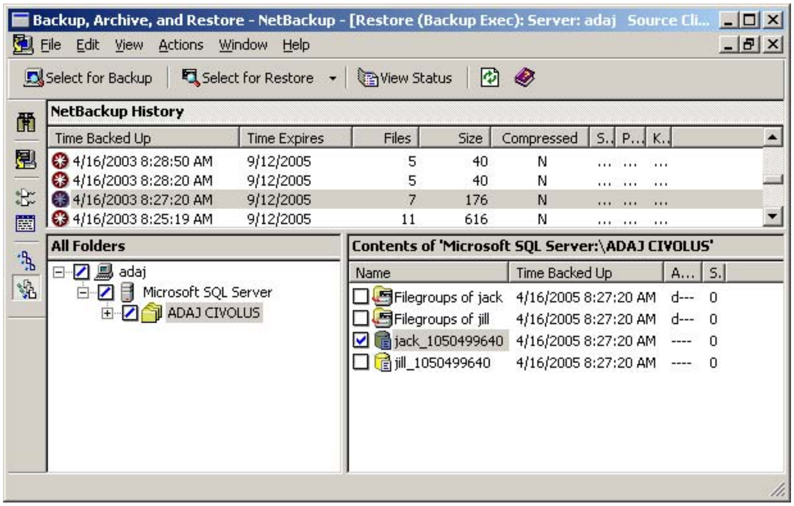
Alternate drive for restoring database files: C:

Restore to alternate drive:
● Only when original drive does not exist
○ Even when original drive does exist

□ Restore all database files to the target instance's data location

□ Point in time log restore: 9/ 8/2005 1:01:34 PM
□ Restore log up to named transaction (SQL 2000):
□ Include the named transaction
□ Found after: 9/ 8/2005 1:01:34 PM

12. When the other filegroups are restored, select the transaction logs.


Backup, Archive, and Restore - NetBackup - [Restore (Backup Exec): Server: adaj Source Cli...
File Edit View Actions Window Help
Select for Backup
Select for Restore
View Status

NetBackup History

| Time Backed Up | Time Expires | Files | Size | Compressed | S.. | P... | K.. |
|---|---|---|---|---|---|---|---|
| 4/16/2003 8:28:50 AM | 9/12/2005 | 5 | 40 | N | ... | ... | ... |
| 4/16/2003 8:28:20 AM | 9/12/2005 | 5 | 40 | N | ... | ... | ... |
| 4/16/2003 8:27:20 AM | 9/12/2005 | 7 | 176 | N | ... | ... | ... |
| 4/16/2003 8:25:19 AM | 9/12/2005 | 11 | 616 | N | ... | ... | ... |

All Folders

- adaj
  - Microsoft SQL Server
    - ADAJ CIVOLUS

Contents of 'Microsoft SQL Server:\ADAJ CIVOLUS'

| Name | Time Backed Up | A... | S. |
|---|---|---|---|
| □ Filegroups of jack | 4/16/2005 8:27:20 AM | d--- | 0 |
| □ Filegroups of jill | 4/16/2005 8:27:20 AM | d--- | 0 |
| ☑ jack_1050499640 | 4/16/2005 8:27:20 AM | ---- | 0 |
| □ jill_1050499640 | 4/16/2005 8:27:20 AM | ---- | 0 |

13. On the Microsoft SQL Server tab, select the recovery completion state **Leave database operational and restore all of the transaction logs**, or select **Point in time log restore** or **Restore log up to named transaction**.

14. Click **Start Restore**.

## Restoring the SQL Master Database from a Backup Exec Image

If the master database is damaged, symptoms may include:

- An inability to start SQL.
- Segmentation faults or input/output errors.
- A report generated by SQL Database Consistency Checker utility (DBCC).

If you can still start SQL, you can restore the latest copy of the master database backup using the **Automate master database restore** option on the **Microsoft SQL Server tab** and then restore any other databases, if needed.

If the master database is critically damaged and SQL cannot be started, rather than running the Rebuild Master utility or reinstalling SQL to be able to restart SQL, you can replace the corrupted or missing databases with the copies of the master and model databases that Backup Exec automatically creates and updates whenever backups of those databases are run. After SQL is running again, you can restore the latest copy of the master database using the **Automate master database restore** option, and then restore any other databases, if needed.

If copies of the master and model databases were not made, then you must use Microsoft's `rebuildm.exe` utility to rebuild the master database and start SQL.

Because all changes made to the master database after the last backup was created are lost when the backup is restored, the changes must be reapplied. If any user databases were created after the master database was backed up, those databases cannot be accessed until the databases are restored from backups or reattached to SQL.

### ▼ To restart SQL using database copies:

1. Verify that the database copies are present.

   The database copies are named master$4idr, mastlog$4idr, model$4idr, and modellog$4idr.

   In a default installation of SQL 2000, the databases are in:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Data\*.*.
   ```

   In a named instance of SQL 2000, the databases are in:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL$Instance_Name\Data\*.*
   ```

   If necessary, restore the master and model database copies from a backup set to the same directory that the original master and model databases are in.

2. Open a command prompt window, and delete the original master and model databases and their transaction logs.

   For example:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Data> del master.mdf
   mastlog.ldf model.mdf modellog.ldf
   ```

3. Rename the copies of the databases back to their original names.

   Type the following:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename
   master$4idr master.mdf

   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename
   mastlog$4idr mastlog.ldf

   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename model$4idr
   model.mdf

   C:\Program Files\Microsoft SQL Server\MSSQL\Data>rename
   modellog$4idr modellog.ldf
   ```

4. Use the SQL Service Control Manager to start SQL Server.

5. Continue with the next procedure to restore the latest changes to the master database.

### ▼ To restore the master database

1. Log on as Administrator.

2. Open the Backup, Archive, and Restore interface.

3. Specify the appropriate server, client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 167.

4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.

5. From the NetBackup History pane, select the backup image containing the last master database backup.

6. In the All Folders pane, select the master database.

7. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.

8. On the **Microsoft SQL Server** tab, select **Automate master database restore**.

   When this option is selected, only the master database can be restored; if this option is selected for any other database, those jobs will fail. When you start the restore operation, all existing users are logged off, and SQL Server is put into single-user mode.

   If NetBackup does not have access to the SQL registry keys HKEY_LOCAL_MACHINE\Software\Microsoft\Microsoft SQL Server and HKEY_LOCAL_MACHINE\Software\Microsoft\MSSQLServer, then a restore to the default directory may not work, and the option **Automate master database restore** will not work. To ensure that NetBackup has access rights, verify that the account that NetBackup uses has administrator rights to the computer that is running SQL.

9. Select a consistency check to be run after the restore.

10. Click **Start Restore**.

    After the restore, SQL is restarted in multi-user mode.

## Redirecting a Restore

You can redirect the following:

- A database backup to a different server, database, or instance.
- Differential and log backups to wherever the associated database is restored.
- One or more filegroups in a backup to a different server or instance. Filegroups can be redirected to a different server, but the database file paths cannot be changed. For example, if the filegroup was backed up from `G:\SQLDATA`, then it must be restored to `G:\SQLDATA`, even if it is redirected to another server. Filegroups must be restored to the same drive letter and path that they were backed up from.

### ▼ To redirect a restore

1. Log on as Administrator.
2. Open the Backup, Archive, and Restore interface.
3. Specify the appropriate server, source client, destination client, and policy type, as described in "Specifying the Server, Client, and Policy Type" on page 167.
4. Choose **File** > **Select Files and Folders to Restore** > **from Backup Exec Backup**.

   The Restore window is displayed.
5. Choose the objects you wish to restore.
6. Choose **Actions** > **Start Restore of Marked Files**.

   The Restore Marked Files dialog is displayed.
7. Click on the **Microsoft SQL Server Redirection** tab.
8. Refer to "Restore Options for Redirected Restores" on page 166
9. Select the other desired restore options on the **Microsoft SQL Server** tab. For more information, refer to "Restore Options" on page 162.
10. Click **OK**.

# Troubleshooting 10

This chapter describes the tools that are available to help you prevent some common, however infrequent, problems encountered with the daily operation of the SQL Server. For more in-depth information, please refer to the Microsoft Support web site.

The following topics cover troubleshooting NetBackup for SQL Server:

- Progress Reports Created for NetBackup for SQL Server on the Client
- Debug Logging
- NetBackup Reports
- Minimizing Timeout Failures on Large Database Restores

# Progress Reports Created for NetBackup for SQL Server on the Client

NetBackup for SQL Server creates a progress report for each operation that has been initiated. This report contains summary information concerning the overall status of your job. The reports are contained in directory `install_path\NetBackup\logs\user_ops\MsSql\logs` and can be viewed by choosing **File** > **View status** in the NetBackup Database Client Graphical User Interface.

The progress report contains the following types of information:

- the batch keywords and values which define the operation. See "Overview of Batch Files" on page 100 for information about the batch file syntax
- summary information about the operation
- information about the operation as it progresses
- any error conditions or warnings that cause the operation to fail
- the final outcome of the operation, whether it succeeded or failed and how long it took

The following is a typical progress report created for a database backup.

| Line | Text |
|---|---|
| 1 | OPERATION BACKUP |
| 2 | DATABASE "DatabaseA" |
| 3 | SQLHOST "JUY" |
| 4 | SQLINSTANCE "NEWINSTANCE" |
| 5 | NBSERVER "JUY" |
| 6 | MAXTRANSFERSIZE 0 |
| 7 | BLOCKSIZE 0 |
| 8 | ENDOPER TRUE |
| 9 | INF - BACKUP STARTED USING |
| 10 | Microsoft SQL Server Yukon - 9.00.852 (Intel X86) |

| Line | Text |
|---|---|
| 11 | Jul 19 2004 22:09:12 |
| 12 | Copyright (c) 1988-2003 Microsoft Corporation |
| 13 | Beta Edition on Windows NT 5.1 (Build 2600: Service Pack 2) |
| 14 | Batch = C:\Program Files\VERITAS\NetBackup\dbext\mssql\temp\__09_42_24_076_00.bch, Op# = 1 |
| 15 | INF - Using backup image juy.MSSQL7.JUY\NEWINSTANCE.db.DatabaseA.~.7.001of001.20050701094227..C |
| 16 | INF - backup database "DatabaseA" to VIRTUAL_DEVICE='VNBU0-5652-1224-1120228947' with stats = 10, blocksize = 65536, maxtransfersize = 65536, buffercount = 1 |
| 17 | INF - Number of stripes: 1, Number of buffers per stripe 1. |
| 18 | 12:41:07 Initiating backup |
| 19 | 12:41:10 INF - Starting bpbrm |
| 20 | 12:41:12 INF - Data socket = juy.domain.com.4146 |
| 21 | 12:41:12 INF - Name socket = juy.domain.com.1527 |
| 22 | 12:41:12 INF - Job id = 143 |
| 23 | 12:41:12 INF - Backup id = juy_1058982070 |
| 24 | 12:41:12 INF - Backup time = 1058982070 |
| 25 | 12:41:12 INF - Policy name = sql |
| 26 | 12:41:12 INF - Snapshot = 0 |
| 27 | 12:41:12 INF - Frozen image = 0 |
| 28 | 12:41:12 INF - Backup copy = 0 |
| 29 | 12:41:12 INF - Master server = juy |
| 30 | 12:41:12 INF - Media server = juy |
| 31 | 12:41:12 INF - Multiplexing = 0 |

| Line | Text |
|---|---|
| 32 | 12:41:12 INF - New data socket = juy.domain.com.1388 |
| 33 | 12:41:12 INF - Use shared memory = 1 |
| 34 | 12:41:12 INF - Compression = 0 |
| 35 | 12:41:12 INF - Encrypt = 0 |
| 36 | 12:41:12 INF - Client read timeout = 300 |
| 37 | 12:41:12 INF - Media mount timeout = 0 |
| 38 | 12:41:16 INF - Data buffer size = 262144 |
| 39 | 12:41:18 INF - Beginning backup on server juy of client juy |
| 40 | INF - Thread has been closed for stripe #0 |
| 41 | 12:41:25 INF - Server status = 0 |
| 42 | 12:41:26 INF - Backup by hao on client juy using policy sql: the requested operation was successfully completed. |

Observe the following:

- Lines 1 to 8 contain the batch syntax which drove this operation. This operation was started from a GUI created batch file.
- Line 9 indicates that this was a backup (“Dump”). The term ‘Load’ would indicate a restore.
- Lines 10-13 provide the version of SQL Server and the Windows operating system.
- Line 14 provides the name of the batch file. Since the backup was launched immediately, it was placed in `*install_path*\NetBackup\DbExt\MsSql\Temp` directory.
- Line 15 provides the actual SQL syntax that was used to launch the backup command.
- Line 17 indicates that this was a single stream backup.
- Lines 18 to 42 provide NetBackup-based client parameters and statuses. Note that the server status on line 40 indicates that the operation completed with status 0, which indicates success.

# Debug Logging

The NetBackup master server and client software offers a comprehensive set of debug logs for troubleshooting problems that may occur during NetBackup operations. Debug logging is also available for SQL backup and restore operations. You can control the amount of information written to debug logs by following the instructions in "Changing the Debug Level". After the cause of the problem has been determined, debug logging can be disabled.

For details on the contents of these debug logs, refer to the *NetBackup Troubleshooting Guide*. For additional NetBackup client logs and NetBackup master server logs, see the online help for the Backup, Archive, and Restore interface and the *NetBackup System Administrator's Guide, Volume I*.

### ▼ To create all debug logs

❖ Run the following batch file:

`install_path\NetBackup\logs\mklogdir.bat`

## Backup Operation Debug Logging

The following logs pertain to backup operations:

`install_path\NetBackup\logs\bphdb` (scheduled backups only)

`install_path\NetBackup\logs\dbclient`

`install_path\NetBackup\logs\bpbkar` (Advanced Client)

`install_path\NetBackup\logs\bpfis` (Advanced Client)

Log names are formatted as *mmddyy*`.log`, where *mm* is the month, *dd* is the day, and *yy* is the year.

## Restore Operation Debug Logging

The following logs pertain to restore operations:

`install_path\NetBackup\logs\dbclient`

`install_path\NetBackup\logs\bpbkar` (Advanced Client)

`install_path\NetBackup\logs\bpfis` (Advanced Client)

`install_path\NetBackup\logs\bppfi` (instant recovery)

Log names are formatted as *mmddyy*`.log`, where *mm* is the month, *dd* is the day, and *yy* is the year.

## Changing the Debug Level

You can control the amount of information written to the debug logs by changing the General debug level. The higher the value, the more information is logged. For most operations, the default value of 0 is sufficient. However, VERITAS technical support may ask you to set the value higher when a problem is being analyzed.

If you wish to set TRACELEVEL parameter to the Maximum (or MAX) level, this must be done in the batch file or in the NetBackup Database Extension GUI. The Maximum level will produce large amounts of output, usually appropriate only for internal debugging.

### ▼ To change the Debug Level

1. Choose **Start** > **Programs** > **VERITAS NetBackup** > **Backup, Archive, and Restore**.

2. Choose **File** > **NetBackup Client Properties**.

3. Click the **Troubleshooting** tab.

   By default, the settings are zero.

4. Set the **General** debug level.

5. Click **OK** to save your changes.

### ▼ To set the maximum trace level in the NetBackup Database Extension GUI

1. Open the NetBackup MS SQL Client.

2. Choose **File** > **Set NetBackup client properties**.

3. In the Client Trace Level group, choose **Maximum**.

### ▼ To set the maximum trace level in the batch file

1. Open the NetBackup MS SQL Client.

2. Choose **File** > **Manage script files**.

3. Select the batch file you wish to change and click **Open File**.

4. Add the following line:

   ```
   TRACELEVEL MAX
   ```

5. Save the file.

# NetBackup Reports

The administrator has access to operational progress reports through administrator interfaces. Reports may be generated for Backup Status, Client Backups, Problems, All Log Entries, Media Lists, Media Contents, Images on Media, Media Logs, Media Summary, and Media Written. These reports may be generated for a specific time frame, client, or master server. Refer to the *NetBackup System Administrator's Guide, Volume I* for details.

# Minimizing Timeout Failures on Large Database Restores

A large SQL Server restore may fail with a Client Read Timeout error before any data has been read from the NetBackup media. This occurs because the SQL Server may need to pre-write the database files before the restore operation begins. The time required for this process is a function of the size of the database files and the speed at which your host machine can write to disk. For example, if your system can perform disk writes at the rate of 60 megabytes per second and you have a 2.4 terabyte database, then it will take at least 12 hours for SQL Server to prep the disk before the actual restore can begin. In reality, the delay may be even longer than what you calculate by as much as 20% to 40%.

The timeout problem can be resolved by increasing the NetBackup Client Read Timeout setting. Use the NetBackup Administration Console on the server to change the properties of each client that contains a database you may need to restore. The default for the Client Read Timeout setting is 300 seconds (5 minutes). If you have clients which contain large SQL Server databases, you may need to set this value much higher.

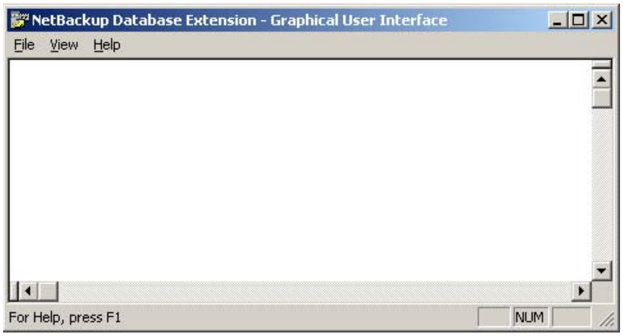
See "Instant Data File Initialization" on page 37 for information on how to eliminate file initialization during SQL Server restores, if you are using SQL Server 2005.

# Reference 11

This chapter provides reference information for the NetBackup for SQL Server, including:

- NetBackup for Microsoft SQL Server Graphical User Interface
- Menu Bar
- Dialogs

# NetBackup for Microsoft SQL Server Graphical User Interface

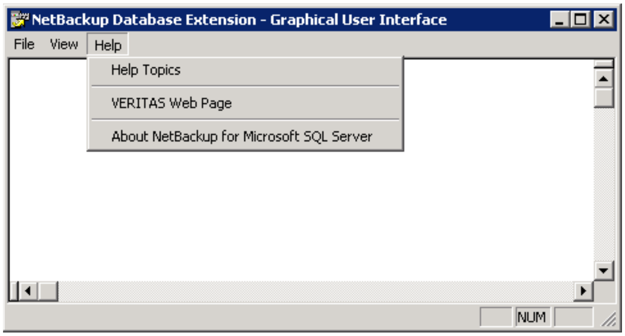


This window contains some familiar Windows elements: the Menu Bar, the Toolbar and the Status Bar. Refer to “Menu Bar” on page 193 and “View Menu” on page 195 for more information on these items.

# Menu Bar

The menus on the Menu Bar have been developed for NetBackup for SQL Server.

## File Menu


NetBackup Database Extension - Graphical User Interface
File
View
Help
Help Topics
VERITAS Web Page
About NetBackup for Microsoft SQL Server
NUM

Use this menu to start a backup, restore or batch file operation.

**Backup SQL Server Objects** This selection opens the Backup Microsoft SQL Server Objects dialog that allows you to perform a database backup, a database incremental backup, a database transaction log backup, a database filegroup backup and a database file backup. Refer to “Backup Microsoft SQL Server Objects” on page 200 for more details.

**Restore Backup SQL Server Objects** This selection opens the Restore Microsoft SQL Server Objects dialog that allows you to perform a database restore, a database incremental restore, a database transaction log restore, a database filegroup restore and a database file restore. Refer to “Restore Microsoft SQL Server Objects” on page 207 for more details.

**Manage script files** Opens the Manage Scripts dialog. Use this dialog to start batch files for NetBackup operations. Refer to “Manage Scripts” on page 217 for more details.

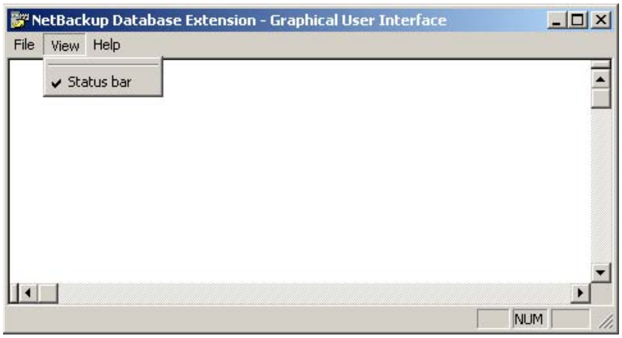
**Set SQL Server connection properties** Opens the SQL Server connection properties dialog. Use this dialog to select the SQL host and instance and to set the userid and password that you want to use for logging into SQL Server.

**Set NetBackup client properties** Opens the NetBackup client properties dialog. Use this dialog to set the current NetBackup for SQL Server trace level, and the block and certain tuning parameters.

**View status** Displays the status of NetBackup operations.

| | |
|---|---|
| **Exit** | Use this command to exit the NetBackup for Microsoft SQL Server GUI. |

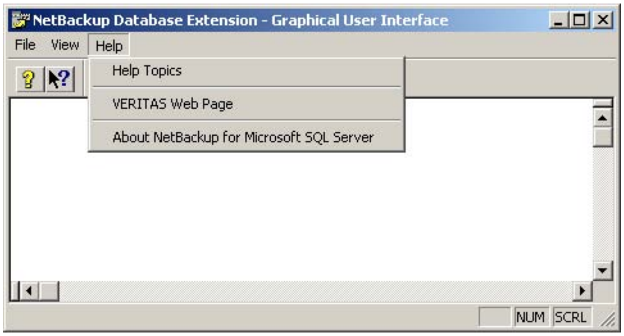
## View Menu



Use this menu to control the appearance of the Status Bar.

**Status bar** Turns the Status Bar on or off.

## Help Menu


NetBackup Database Extension - Graphical User Interface
File
View
Help
Help Topics
VERITAS Web Page
About NetBackup for Microsoft SQL Server
NUM
SCRL

Use this menu to access the following help pages.

**Help Topics** Opens the NetBackup Help topics. NetBackup Help contains the same information found in the user guide. You can also access NetBackup Help by using the context-sensitive button on the Toolbar. Refer to "NetBackup for Microsoft SQL Server Graphical User Interface" on page 156 for more details.

**VERITAS Web Page** Displays the VERITAS world wide web page on your computer's default browser.

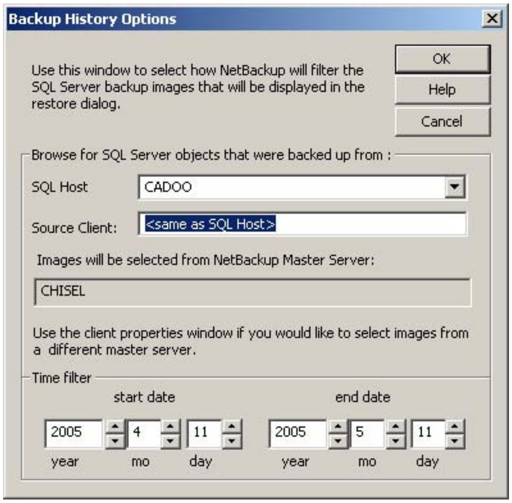
**About NetBackup for SQL Server** Displays the version number of NetBackup for SQL Server.

# Dialogs

The following dialog boxes appear in the NetBackup for SQL Server interface:

- "Backup History Options" on page 198
- "Backup Microsoft SQL Server Objects" on page 200
- "NetBackup client properties" on page 205
- "Restore Microsoft SQL Server Objects" on page 207
- "SQL Server connection properties" on page 215
- "Manage Scripts" on page 217

## Backup History Options



Backup History Options

Use this window to select how NetBackup will filter the SQL Server backup images that will be displayed in the restore dialog.

Browse for SQL Server objects that were backed up from :

SQL Host: CADOO

Source Client: <same as SQL Host>

Images will be selected from NetBackup Master Server: CHISEL

Use the client properties window if you would like to select images from a different master server.

Time filter

| | year | mo | day |
|---|---|---|---|
| start date | 2005 | 4 | 11 |
| end date | 2005 | 5 | 11 |

This dialog is displayed when you choose **File** > **Restore Backup SQL Server Objects**.

Use this dialog to:

- Set a different client on the NetBackup Server from which to browse for backup images
- Change the start and end times for which you would like to browse for images

**SQL Host** From this list, choose the SQL Server host that you would like to generate a backup history for.

**Source Client** Specifies the NetBackup client name for the selected client. This is the name used in the policy to identify the client.

**Note** If you use a specific network interface for backup, then the network interface name should be entered in the **Source Client** box. The network interface name is defined in the Host Properties for the server (open the properties for the server and click **Universal Settings**).

**Time Filter** Define the range of backups you want to search in this box.

**Start Date** Enter the earliest date of the backup you want to search.

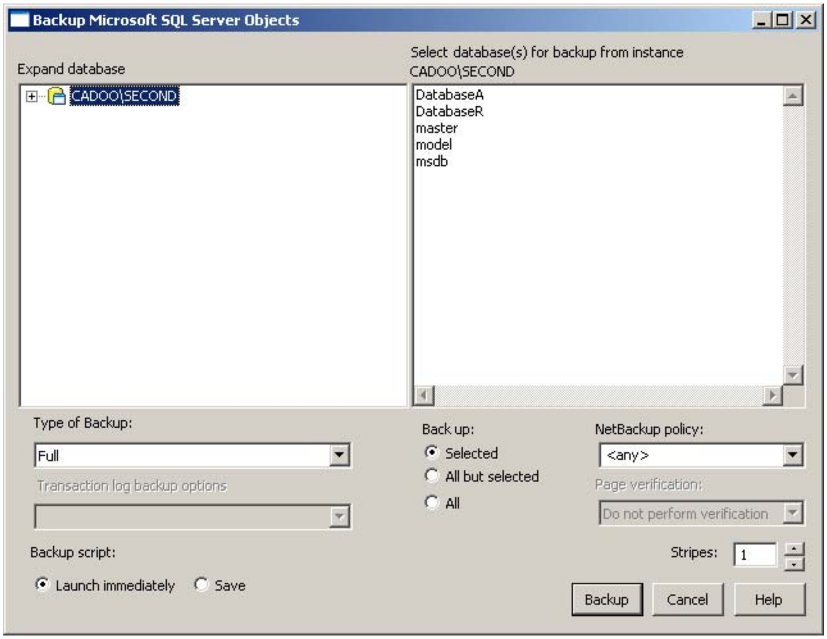
**End Date** Enter the latest date of the backup you want to search.

**OK** Click this button to change your backup history. After you click **OK** the Restore Microsoft SQL Server Objects dialog is displayed.

**Help** View the online help for this dialog.

**Cancel** Cancel changes and close the dialog.

## Backup Microsoft SQL Server Objects



Backup Microsoft SQL Server Objects

Expand database

CADOO\SECOND

Select database(s) for backup from instance CADOO\SECOND

DatabaseA
DatabaseR
master
model
msdb

Type of Backup: Full

Transaction log backup options:

Back up: ● Selected ○ All but selected ○ All

NetBackup policy: <any>

Page verification: Do not perform verification

Stripes: 1

Backup script: ● Launch immediately ○ Save

To access this dialog, by choose **File** > **Backup SQL Server Objects**. Use this dialog to initiate a backup operation.

## Dialog Options

Expand database — This pane allows you to traverse live databases. You can expand the SQL Server instance to view its databases. Expanding each database will allow you to view its filegroups and you can expand a filegroup to view its files. You can select any object in this pane in order to view its constituent objects in the right-hand pane.

Select database(s) for backup from *instance host\instance* — Select the objects that you want to back up from this pane. This pane displays the list of constituent database objects of the highlighted host\instance in the left-hand pane. You can select one or more objects (databases) in this pane. (Use Ctrl + click and Shift + click to select multiple objects.)

**Type of Backup**

**Full** — Create a full database backup.

**Full differential** — Create a differential backup.

**Transaction log** — Create a transaction log backup. This is only available for databases. When you select this type of backup, you then need to select a backup option from the **Transaction log backup options** list.

**Note** The following four backup types are only available for SQL Server 2005

**Read/write filegroups** — Create a backup of read/write filegroups in a database.

**Differential on read/write filegroups** — Create a differential backup of read/write filegroups in a database.

**Create a template for partial backup** — Create a backup of only the selected filegroups in a database.

**Create a template for partial differential backup** — Create a differential backup of only the selected filegroups in a database.

**Transaction log backup options** — These options are available when you have chosen a transaction log backup type.

**Back up and truncate transaction log** — Back up the transaction log and remove the inactive part of the transaction log.

**Back up transaction log but don’t truncate it** — Back up a transaction log without truncating it.

**Backup and restore tail log** Back up and recover the tail log from disk.

**Backup Script**

**Launch Immediately** Start the backup operation immediately.

**Note** **Launch immediately** is disabled if you are logged into a SQL Server instance that is not on the local host. If you generate a script for a non-local host, then it must be executed on that host.

**Save** Generate a script that can be started at a later time.

**Note** Launch immediately is disabled for remote hosts. Scripts must be executed from the remote host.

**Back up** Choose which objects to backup in the right-hand pane.

**Selected** Back up only the objects selected.

**All but selected** Back up all of the objects, except those selected.

**All** Back up all of the objects.

**NetBackup Policy** If this host is NetBackup master server, then this list will include all active policies of type MS-SQL-Server. You may select one of these policies, or type the name of a policy.

The default is <any>. If you choose the default, then NetBackup will select which MS-SQL-Server policy to use.

**Page verification** (SQL Server 2005 only) This list is disabled unless all of the objects in the right-hand pane have a page verification type that is either *torn page detection* or *checksum*.

**Note** There is a performance penalty when using page verification.

**Do not perform verification** Do perform page verification before running the backup.

**Stop on error** Perform page verification when running the backup and stop the backup if a verification error is encountered.

**Continue after error** Perform page verification when running the backup and continue the backup even if a verification error is encountered.

**Note** If this is selected and a snapshot method is chosen for backup, the SQL Server agent will ignore the request for page verification and generate a warning.

**Caution**Microsoft does not recommend using this option.

**Stripes** Set the number of backup stripes that you want SQL Server to create for your backup. Type a number from 1 to 32. Refer to “Configuring Multi-Stream Operations” on page 26.

**Backup** Start a database backup or generate a database backup script. This is enabled only when you have selected an object to back up.

**Cancel** Cancel changes and close the dialog.

**Help** View the online help for this dialog.

## Viewing Properties of SQL Server Objects Selected for Backup

You can view the properties of any object in the Backup Microsoft SQL Server Objects dialog by right-clicking on the object. The properties are described below.

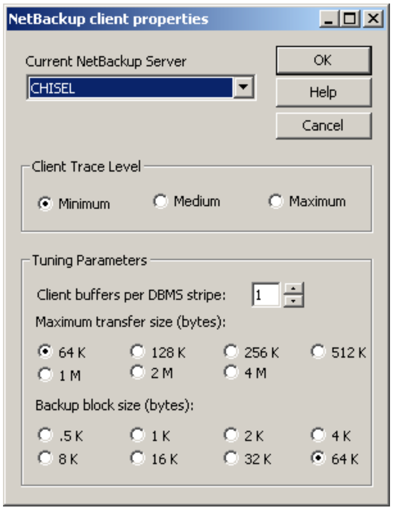
Properties of objects selected for backup

| **Property** | **Description** |
|---|---|
| Object type | Database, database filegroup, database file, or transaction log. |
| Object name | Name of the object. |
| Parent (database, instance, filegroup, etc.) | Name of the object’s parent. |
| SQL Server instance | SQL Server instance the object belongs to. |
| File size | (SQL 2005) The size of the component files. This size should closely match the size of a backup snapshot. |
| Data size | (SQL 2005) Size of the backup stream. Applies to databases only. |
| Page verification | (SQL 2005) The type of SQL Server page verification that is configured for selected databases, filegroups, and logical files. There are three possible values: *none*, *torn page detection*, or *checksum*. |

Properties of objects selected for backup

| Property | Description |
|---|---|
| Read-only/read-write | (SQL 2005) The attribute applied to the filegroup. |
| On-line/off-line | (SQL 2005) The status of the filegroup. |
| Path | (Database files only) The absolute path of the database file. |

## NetBackup client properties



NetBackup client properties

Current NetBackup Server: CHISEL

Client Trace Level

● Minimum ○ Medium ○ Maximum

Tuning Parameters

Client buffers per DBMS stripe: 1

Maximum transfer size (bytes):

● 64 K ○ 128 K ○ 256 K ○ 512 K
○ 1 M ○ 2 M ○ 4 M

Backup block size (bytes):

○ .5 K ○ 1 K ○ 2 K ○ 4 K
○ 8 K ○ 16 K ○ 32 K ● 64 K

To access this dialog, choose **File** > **Set NetBackup client properties**. Use this dialog to set the following options for your current session.

**Current NetBackup Server** Select the NetBackup Server that you wish to use for NetBackup For SQL Server Operations.

**Note** The NetBackup Server that you select will be associated with any NetBackup for SQL Server operations that are started under your current user account.

**Client Trace Level Group** Set the trace level written to the `dbclient` log. Normally, you should set the level to Minimum. Maximum generates a very large amount of debug data.

**Tuning Parameters** Set the number of client buffers per DBMS stripe, the maximum transfer size in bytes, and the backup block size in bytes.

**Client Buffers per DBMS Stripe** Set the number of buffers that SQL Server uses per database stripe.

**Maximum Transfer Size (Bytes)** Set the maximum transfer size that SQL Server uses for backing up and restoring database objects. It can be set to any of the indicated values between 64 kilobytes and 4 Megabytes.

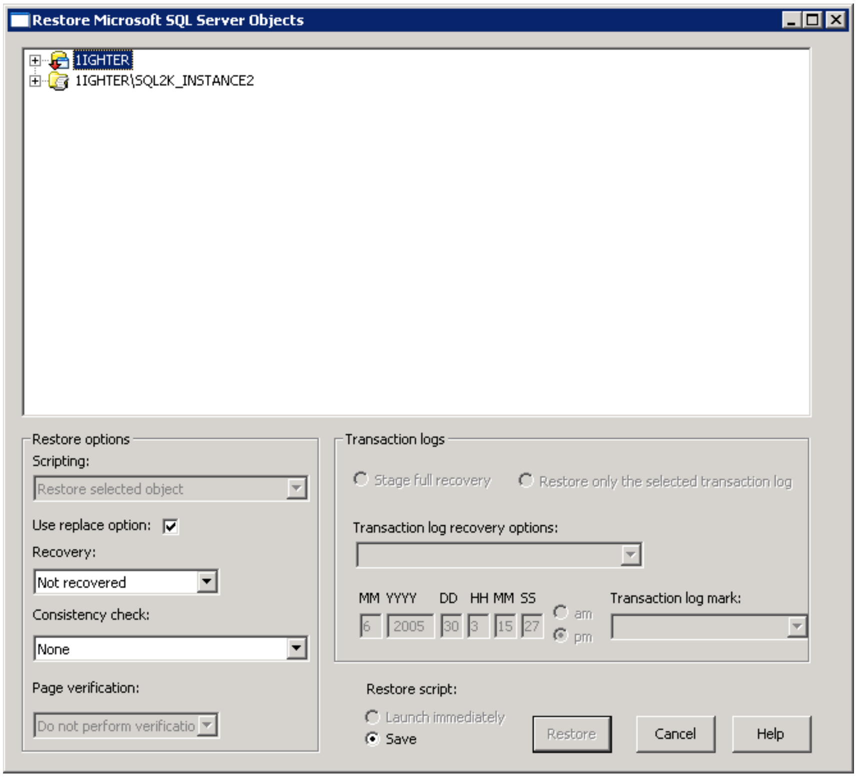
**Backup Block Size (Bytes)** Set the backup block size that SQL Server uses for backing up database objects. It can be set to any of the indicated values between 0.5 kilobytes and 64 kilobytes.

**OK** Click to accept any changes.

**Help** View the online help for this dialog.

**Cancel** Cancel changes and close the dialog.

# Restore Microsoft SQL Server Objects



Restore Microsoft SQL Server Objects

- 1IGHTER
- 1IGHTER\SQL2K_INSTANCE2

Restore options

Scripting: Restore selected object

Use replace option: ☑

Recovery: Not recovered

Consistency check: None

Page verification: Do not perform verificatio

Transaction logs

○ Stage full recovery ○ Restore only the selected transaction log

Transaction log recovery options:

MM YYYY DD HH MM SS: 6 2005 30 3 15 27 ○ am ● pm

Transaction log mark:

Restore script: ○ Launch immediately ● Save

To open this dialog, choose **File** > **Restore SQL Server Objects**, and after the Backup History Options dialog is displayed, click **OK**. Use this dialog to restore SQL Server databases.

## Dialog Options

Expand database

This is a tree display of the database backup history for a NetBackup client. The backup client is typically the local host on which you are running NetBackup for SQL Server.

However, it may be different if a different client has been specified through the Backup History Options dialog.

You can expand and collapse portions of the tree by clicking on the **+** and **-** buttons. The tree can be expanded as follows:

| | |
|---|---|
| Level 0: Database instance | Items at this level cannot be selected. |
| Level 1: Database icons | Items at this level cannot be selected. |
| Level 2: Database backup images | These are full database backup images. |
| Level 3: Database differential images | These are differential backup images. |
| Level 2: Read-write Filegroups icon | Expand to display backups of the read-write filegroups in the database. |
| Level 3: Read-write Filegroups backup images | These images contain all of the read-write filegroups in the database. |
| Level 4: Read-write Filegroups differential images | |
| Level 2: Partial database backup icon | Expand to display a list of partial database backup images. |
| Level 3: Partial database backup images | These images contain backups of some of the filegroups contained in the database. To view the contents of a partial database backup image, right-click on an image and choose **Properties**. |
| Level 4: Partial database differential backup images | These images contain differential backups of filegroups contained in the database. To view the contents of a partial database backup image, right-click on an image and choose **Properties**. |
| Level 2: Transaction log backup images | Each transaction log backup image is displayed with the date and time of the backup. |
| Level 2: Filegroup icons | Expand to display a list of filegroups backup images. |
| Level 3: Filegroup backup images | Click on the + next to a filegroup icon to view filegroup backups. The date and time of a backup is also displayed. |

Level 4: Filegroup differential images — Click on the + next to a filegroup backup image to view any filegroup differential backups. The date and time of a differential backup is also displayed.

Level 3: File icons — The individual file icons appear at level 3. These icons cannot be selected for restore.

Level 4: File images — Clicking on the + next to a file icon to view individual file backups. Each of these images is displayed with the date and time of the backup.

**Restore Options**

**Use replace option** — Restore with the SQL Server replace option.

**Scripting options** — These scripting options are available when restoring from a database image.

**Restore selected object** — Produce a script that performs a database restore. This is the default option.

**Create a move template** — Create a script template for moving the selected database.

**Create partial database restore template** — (SQL Server 2000 only) Create a template for a partial database restore.

**Restore read-only filegroups** — (SQL Server 2005 only) Restore the most recent backup of every read-only filegroup.

**Create a page restore template** — (SQL Server 2005 only) Create a template for restoring a database, filegroup, or file from the pages contained in the selected backup image.

**Verify backup image, but don't restore** — (SQL Server 2005 only) This option is only available if the image was backed up using the page verification option. NetBackup will process the image for errors, but will not perform a restore.

**Recovery** — Specify one of the SQL Server recovery options.

**Not recovered**
Use this option during a restore if additional backup images must be applied to the database following the current restore. When you use this option, the database is left in a loading state.

**Recovered**
Select this option when restoring the last image in a restore sequence. After the recovery operation, the database is ready for use. If recover database is not performed, the database is in an intermediate state and is not usable. If **Recovered** is selected when an intermediate backup is being applied, you cannot continue to restore backups; you must restart the restore operation from the beginning.

**Standby**
Create and maintain a standby during a transaction log and database restore. This option requires a standby undo log, which, by default, is placed in `install_path\NetBackup\logs\SQLStandBy\`. The database is placed in “standby” state following the restore.

**Consistency Check**
Select the consistency check to be performed after the restore. Output from the consistency check is written to the SQL Server client progress log. You cannot select consistency checking unless the database is being restored to the recovered state. If you choose consistency checking for a staged recovery, then the check occurs following the last restore.

**None**
Do not perform consistency checking.

**Full check, excluding indexes**
Exclude indexes from the consistency check. If indexes are not checked, the consistency check runs significantly faster but is not as thorough. Only the data pages and clustered index pages for each user table are included in the consistency check. The consistency of the non-clustered index pages are not checked.

**Full check, including indexes**
Include indexes in the consistency check. Any errors are logged.

**Check catalog**
Check for consistency in and between system tables in the specified database.

**Physical check only**
(SQL 2000 or later) Select this to perform a low overhead check of the physical consistency of the SQL Server 2000 or later database. This option only checks the integrity of the physical structure of the page and record headers, and the consistency between the pages’ object ID and index ID and the allocation structures.

**Page verification**

(SQL 2005 only) These options are available if the source object was backed up with torn page detection or checksum verification.

**Note** There is a performance penalty when using page verification.

**Do not perform verification**

Do not include page verification in the restore script.

**Stop on error**

Include page verification in the restore script and stop the restore if an error is encountered.

**Continue after error**

Include page verification in the restore script and continue the restore if an error is encountered.

**Caution**Microsoft does not recommend using this option.

Transaction logs

This group contains transaction log recovery options. It is enabled if you select a transaction log backup image.

**Stage full recovery**

Select this option if you wish to recover the database using the recovery set that NetBackup has found. If the transaction log that you have selected does not belong to a recovery set, then this option is disabled.

**Restore selected transaction log**

Select this option to restore only the highlighted transaction log. If the transaction log that you have selected does not belong to a recovery set, then this option is disabled.

**Transaction log recovery options**

This list contains the controls for you to restore a transaction log to a point in time that precedes the time at which the transaction log was dumped. The individual entries in this group are only enabled if you have highlighted a transaction log backup.

For SQL Server 7.0, the only available option from this group is **To point in time**. The remaining options, which allow you to restore to a transaction mark, or to a time before or after a transaction mark, are enabled if you have a later version of SQL Server.

**To point in time**

Select this radio item to have the transaction log recovered to a point in time.

**To transaction log mark**

Select this radio item to have the transaction log recovered to a transaction log mark. With this option, you must enter a transaction log mark name.

**To transaction log mark but after**

Select this radio item to choose to have the transaction log recovered to a transaction log mark but after a point in time. With this option, you must enter a transaction log mark name.

**Before transaction log mark** Select this radio item if you would like to have the transaction log recovered to a point before the occurrence of a transaction log mark. With this option, you must enter a transaction log mark name.

**Before transaction log mark but after** Select this radio item if you would like to have the transaction log recovered to a point before the occurrence of a transaction log mark but after a point in time. With this option, you must enter a transaction log mark name.

**MM, YYYY, DD, HH, MM, SS am, pm** Specify the time to which you want the transaction logs restored. These fields are only enabled if you have selected one of the following **Transaction log recovery options**:

- To point in time
- To transaction log mark but after
- Before transaction log mark but after

**Transaction log mark** This list is enabled if you have highlighted a database transaction log for restore, and the transaction log contains one or more transaction log marks, and you have selected one of the following **Transaction log recovery options**:

- To transaction log mark
- To transaction log mark but after
- Before transaction log mark
- Before transaction log mark but after

Restore script

**Launch immediately** Start the restore operation immediately.

**Note** **Launch immediately** is disabled if you are logged into a SQL Server instance that is not on the local host. If you generate a script for a non-local host, then it must be executed on that host.

**Save** Generate a script that can be started at a later time.

**Restore** Start the restore or generate a restore script. This is disabled if you have not selected any objects to restore.

**Cancel** Cancel changes and close the dialog.

**Help** View the online help for this dialog.

## Viewing Properties of Objects Selected for Restore

You can view the properties of any object in the Restore Microsoft SQL Server Objects dialog by right-clicking on the object. The properties are described below.

### ▼ To view the properties of an object selected for restore

1. Choose **File** > **Restore SQL Server Objects**.

   The Backup History Options dialog is displayed.

2. In the right pane, right-click on an object and choose **Properties**.

3. When you are finished, click **OK**.

**Backup properties tab**

The following properties are displayed on the Backup properties tab.

Properties of objects selected for restore

| Property | Description |
|---|---|
| Object type | Database, database filegroup, database file, or transaction log. |
| Object name | Name of the object. |
| Parent filegroup | Name of the parent filegroup |
| Parent database | Name of the parent database. |
| SQL Server instance | SQL Server instance the object belongs to. |
| Time backed up | The date and time the backup was performed. |
| Backup policy | Name of the policy used to perform the backup. |
| Page verification | (SQL 2005) The type of SQL Server page verification that is configured for selected databases, filegroups, and logical files. There are three possible values: *none*, *torn page detection*, or *checksum*. If multiple objects are selected, then a value is displayed only if all of these objects have the same verification property |
| Method | The method used to perform the backup, *standard backup* or *snapshot*. |
| Type of snapshot method | Snapshot method that was used for the backup. |

Properties of objects selected for restore

| Property | Description |
|---|---|
| Image | The name of the backup image containing the selected object(s). |
| Full text catalogs directories | Full text catalog names and directories. |

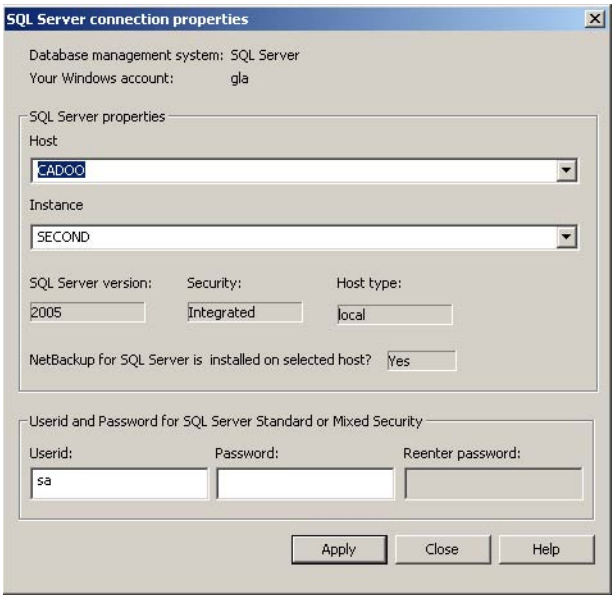
### Recovery set tab

This tab is provided for transaction log and differential backups that are members of a full recovery set. When the recovery set tab is selected, the set of images composing the recovery set is listed.

### Read-only backup set tab

This tab is provided at the database level for SQL Server 2005 images. When the tab is selected, a list of the images containing the most recent backups of each read-only filegroup is provided. If a complete set of read-only filegroups is not available or if the database does not contain any read-only filegroups, then an explanation is provided.

## SQL Server connection properties



SQL Server connection properties

Database management system: SQL Server
Your Windows account: gla

SQL Server properties

Host: CADOO

Instance: SECOND

SQL Server version: 2005
Security: Integrated
Host type: local

NetBackup for SQL Server is installed on selected host? Yes

Userid and Password for SQL Server Standard or Mixed Security

Userid: sa
Password:
Reenter password:

To access this dialog choose **File** > **Set SQL Server connection properties**. Use this dialog to set the login parameters that NetBackup for SQL Server uses when logging into SQL Server. Refer to "Selecting the SQL Host and Instance" on page 57 for information about how to use this dialog.

SQL Server properties

**Host** This list contains all of the SQL Server hosts in the network. Select a SQL Server host from the list or type a name.

**Instance** This list contains all of the SQL Server instances on the selected host. Select a SQL Server instance from the list or type a name.

**Note** The keyword <default> shown in the Instance box designates the default SQL Server.

SQL Server version The version of the SQL instance selected **Instance** box.

Security The type of security being used by the SQL instance selected in **Instance** box.

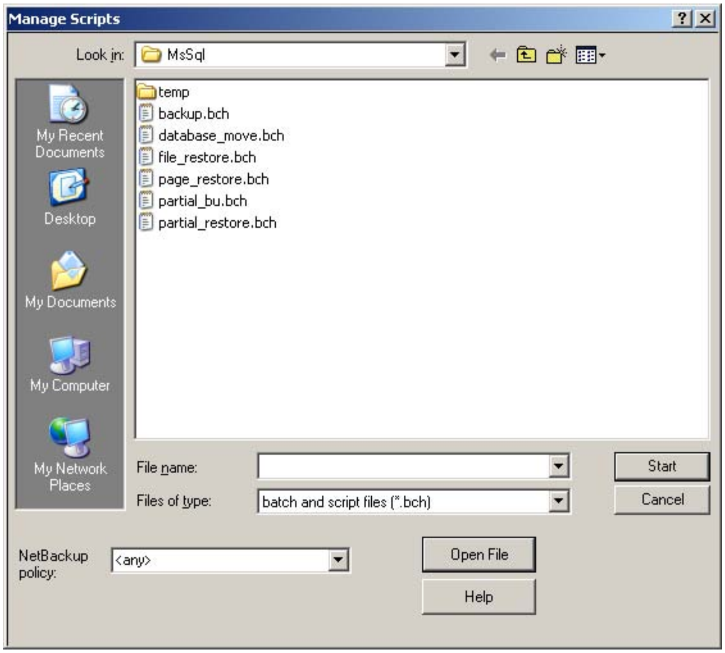
**Host type** The host type, local, remote, virtual or unknown.

**Note** The three previous items are updated when you click the **Apply** button.

Userid and Password for SQL Server Standard or Mixed Security

| | |
|---|---|
| **Userid** | The username, or Windows user id, that NetBackup for SQL Server uses to log into SQL Server. |
| **Password, Reenter password** | Indicate the SQL Server password associated with the Windows user id. |
| **Apply** | Apply changes made in the dialog. |
| **Close** | Close the dialog with accepting changes made in the dialog. |
| **Help** | View the online help for this dialog. |

## Manage Scripts



To access this dialog choose **File** > **Manage script files**. Use this dialog to start a NetBackup for SQL Server operation from a NetBackup for SQL Server batch file that you have written, or to open a batch file in NotePad.

This dialog displays the file icons for the contents of the `*install_path*\NetBackup\DbExt\MsSql` folder. VERITAS advises that you keep NetBackup for SQL Server batch files in this folder; however, batch files can be kept in another location.

**General** This dialog uses the common controls from the Windows File menu. You can get additional help on many of the items in this dialog by pressing F1 and clicking on that item.

**File name** Type the name of the file you want to use for starting a batch file operation or to open the file in NotePad. You can also select the file icon.

**Files of type** Select a file extension to control which files are displayed.

**NetBackup policy** When the policy type is MS-SQL-Server and the host is a NetBackup master server or a remote console, then a drop-down list is available from which you can choose from the list of the policies defined on the server.

Otherwise, type the name of the NetBackup policy on the master server that you want to back up to. If you use the default (<any>), NetBackup will back up to the first available policy. Note that the NetBackup policy is applicable only for backup operations, not for restores.

**Open File** Click this button to open the selected file in NotePad.

**Help** View the online help for this dialog.

**Start** Launch the NetBackup for SQL Server operation.

**Cancel** Cancel changes and close the dialog.

**Note** See "Overview of Batch Files" on page 100 for a description of how to create a batch file.

# Sample Batch Files A

## Sample 1 - Simple script to back up a database named BUSINESS

Note that the parameters for this operation are guided by certain default values. For example, there will be one backup stripe, minimum trace level, and the object type will be a database (as opposed to a transaction log).

```
OPERATION BACKUP
DATABASE "BUSINESS"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
ENDOPER TRUE
```

# Sample 2 - Simple script to restore a database named pubs

This sample restores database *pubs* based upon the backup:

```
NBIMAGE
"cadoo.MSSQL7.CADOO\SECOND.db.pubs.~.7.001of001.20050628123631..C"
```

You can find out which backups are available for restore by looking at the `dbclient` log file created when you did the backup or by using `bplist`. Refer to “Using bplist to Retrieve SQL Server Backups” on page 112.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE "pubs"
# The following image is type: Full
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.db.pubs.~.7.001of001.20050628123631..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

## Sample 3 - Perform a striped database backup. Allow multiple internal buffers per stripe

This example backs up the BUSINESS database using 4 data streams. Each data stream uses 2 buffers.

```
OPERATION BACKUP
DATABASE "BUSINESS"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
STRIPES 4
NUMBUFS 2
MAXTRANSFERSIZE 0
BLOCKSIZE 7
ENDOPER TRUE
```

## Sample 4 - Restore a database from multiple stripes

For a striped restore, you must specify the number of stripes and the name of the first backup image name. Notice that the backup image in this example is embedded with the string `.001of004`, which indicates that it is the first of four backups.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE “Northwind”
NBIMAGE cadoo.MSSQL7.CADOO.db.Northwind.~.0.001of004.20050216151937..C
STRIPES 004
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
BROWSECLIENT "CADOO"
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

# Sample 5 - Restore a database transaction log up to a point in time

This script is executed after the database is restored. The database will be restored to the specified point in time (Feb 16, 2005 at 2:03:00 PM), which precedes the date of the backup log (Feb 16, 2005 at 2:03:21 PM).

```
OPERATION RESTORE
OBJECTTYPE TRXLOG
STOPAT 20050216/14:03:00
DATABASE Northwind
NBIMAGE cadoo.MSSQL7.CADOO.trx.Northwind.~.0.001of001.20050216140321..C
MAXTRANSFERSIZE 0
BLOCKSIZE 0
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
BROWSECLIENT "CADOO"
ENDOPER TRUE
```

**Note** If `STOPAT` is not specified, then the database would be restored to the date of the backup log.

**Note** You can avoid manually staging the restoration of the database backup and the associated log files by creating a script using the Restore Microsoft SQL Server Objects dialog.

**Note** Since `RECOVEREDSTATE` was not specified, the database will be restored to a recovered state following successful execution of this script.

## Sample 6 - Perform an operation and specify the userid and password to use to SQL Server

Only specify a userid and password if you are using standard SQL Server security. See “SQL Server Privileges” on page 31 for more information.

```
OPERATION BACKUP
DATABASE "BUSINESS"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
USERID JSMITH
PASSWORD my.Pwd
ENDOPER TRUE
```

# Sample 7 - Perform multiple operations in sequence

In this sample batch file, five separate backups will be performed sequentially. Remember that each operation is required to be completely specified.

```
OPERATION BACKUP
DATABASE "BUSINESS"
OBJECTTYPE DATABASE
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
STRIPES 5
ENDOPER TRUE

OPERATION BACKUP
DATABASE "RECREATION"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
OBJECTTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
DATABASE "EDUCATION"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
STRIPES 2
ENDOPER TRUE

OPERATION BACKUP
DATABASE "GOVERNANCE"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
OBJECTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
```

```
DATABASE "SURVIVAL"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
OBJECTYPE TRXLOG
ENDOPER TRUE
```

# Sample 8 - Perform a set of operations in parallel

This sample is identical to the previous one except that the first operation contains `BATCHSIZE 3`. This tells NetBackup to start the first three operations in parallel, then after these are all completed begin the next set of 3. In this case, since there are just five operations, the second batch set will contain two operations.

```
BATCHSIZE 3
OPERATION BACKUP
DATABASE "BUSINESS"
OBJECTTYPE DATABASE
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
STRIPES 5
ENDOPER TRUE

OPERATION BACKUP
DATABASE "RECREATION"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
OBJECTTYPE TRXLOG
ENDOPER TRUE

OPERATION BACKUP
DATABASE "EDUCATION"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
STRIPES 2
ENDOPER TRUE

OPERATION BACKUP
DATABASE "GOVERNANCE"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
OBJECTYPE TRXLOG
```

```
ENDOPER TRUE

OPERATION BACKUP
DATABASE "SURVIVAL"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
OBJECTYPE TRXLOG
ENDOPER TRUE
```

# Sample 9 - Specify the maximum transfer size and block size for a backup

This sample batch file backs up database business using a maximum transfer size of 64 kilobytes bytes * $2^4$ (1M) and a block size of 512 bytes * $2^6$ (32 kilobytes).

```
OPERATION BACKUP
DATABASE "BUSINESS"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "CHISEL"
MAXTRANSFERSIZE 4
BLOCKSIZE 6
ENDOPER TRUE
```

# Sample 10 - Stage a database restore from a database backup, a differential backup, and a series of transaction backups

This is an example of a script generated by the Restore Microsoft SQL Server Objects dialog.

```
OPERATION RESTORE
OBJECTTYPE DATABASE
DATABASE "DatabaseA"
# The following image is type: Full
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.db.DatabaseA.~.7.001of001.20050701094227..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE DATABASE
DUMPOPTION INCREMENTAL
DATABASE "DatabaseA"
# The following image is type: Full database differential
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.inc.DatabaseA.~.7.001of001.20050701103323..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
DATABASE "DatabaseA"
# The following image is type: transaction log
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.trx.DatabaseA.~.7.001of001.20050701090005..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
```

```
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
DATABASE "DatabaseA"
# The following image is type: transaction log
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.trx.DatabaseA.~.7.001of001.20050701100030..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
DATABASE "DatabaseA"
# The following image is type: transaction log
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.trx.DatabaseA.~.7.001of001.20050701110015..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE
```

## Sample 11 - Stage a database restore from a filegroup backup, several file backups, and transaction log backups

This is an example of a full database restore script generated by the Restore Microsoft SQL Server Objects dialog.

```
OPERATION RESTORE
OBJECTTYPE FILEGROUP
DATABASE "DatabaseR"
OBJECTNAME "PRIMARY"
# The following image is type: Filegroup
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.fg.DatabaseR.PRIMARY.7.001of001.20050701095634..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILEGROUP
DATABASE "DatabaseR"
OBJECTNAME "DBR_FG2"
# The following image is type: Filegroup
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.fg.DatabaseR.DBR_FG2.7.001of001.20050701095425..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILE
DATABASE "DatabaseR"
OBJECTNAME "DBR_FG1_File1"
# The following image is type: File
NBIMAGE
"cadoo.MSSQL7.CADOO\SECOND.fil.DatabaseR.DBR_FG1_File1.7.001of001.20050701100824..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
```

```
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILE
DATABASE "DatabaseR"
OBJECTNAME "DBR_FG1_File2"
# The following image is type: File
NBIMAGE
"cadoo.MSSQL7.CADOO\SECOND.fil.DatabaseR.DBR_FG1_File2.7.001of001.20050701100908..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE FILE
DATABASE "DatabaseR"
OBJECTNAME "DBR_FG1_File3"
# The following image is type: File
NBIMAGE
"cadoo.MSSQL7.CADOO\SECOND.fil.DatabaseR.DBR_FG1_File3.7.001of001.20050701100953..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
DATABASE "DatabaseR"
# The following image is type: transaction log
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.trx.DatabaseR.~.7.001of001.20050701100030..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
```

```
RECOVEREDSTATE NOTRECOVERED
ENDOPER TRUE

OPERATION RESTORE
OBJECTTYPE TRXLOG
DATABASE "DatabaseR"
# The following image is type: transaction log
NBIMAGE "cadoo.MSSQL7.CADOO\SECOND.trx.DatabaseR.~.7.001of001.20050701110015..C"
SQLHOST "CADOO"
SQLINSTANCE "SECOND"
NBSERVER "BOW"
BROWSECLIENT "CADOO"
MAXTRANSFERSIZE 0
BLOCKSIZE 7
RESTOREOPTION REPLACE
RECOVEREDSTATE RECOVERED
ENDOPER TRUE
```

# Index